



Universidad Autónoma de Querétaro

Facultad de Ingeniería

Maestría en Ciencias en Inteligencia Artificial

Segmentación Automática de Vasos Sanguíneos en Imágenes Cerebrales

TESIS

Que como parte de los requisitos para obtener el grado de Maestro en Ciencias en
Inteligencia Artificial

Presenta:

Oliver Jonathan Quintana Quintana

Dirigido por:

Dr. Saúl Tovar Arriaga

Dirección General de Bibliotecas UAQ



Universidad Autónoma de Querétaro

Facultad de Ingeniería

Maestría en Ciencias en Inteligencia Artificial

Segmentación Automática de Vasos Sanguíneos en Imágenes Cerebrales

TESIS

Que como parte de los requisitos para obtener el grado de Maestro en Ciencias en

Inteligencia Artificial

Presenta:

Oliver Jonathan Quintana Quintana

Dirigido por:

Dr. Saúl Tovar Arriaga

SINODALES

Dr. Saúl Tovar Arriaga

Presidente

Dr. Iván Villanueva Naquid

Secretario

Dr. Arturo González Gutiérrez

Vocal

M.C. Alejandro de León Cuevas

Suplente

Dr. Marco Antonio Aceves Fernández

Suplente

Centro Universitario, Querétaro, QRO

Septiembre 2021

México.

© 2021 - Oliver Jonathan Quintana Quintana

All rights reserved.

Dirección General de Bibliotecas UAQ

Dirección General de Bibliotecas UAQ

Este trabajo está dedicado a mi familia.

Agradecimientos

Primeramente, al Consejo Nacional de Ciencia y Tecnología (CONACyT) por proveer los fondos para realizar este trabajo de investigación. Agradezco el apoyo de Luis Aguilar y Jair García del Laboratorio Nacional de Visualización Científica Avanzada de la Universidad Nacional Autónoma de México (UNAM) campus Juriquilla, así como por proveer material de apoyo para la realización de este trabajo. Al Dr. Erick Humberto Pasaye Alcaraz, adscrito al Laboratorio Nacional de Imagenología por Resonancia Magnética del Instituto de Neurobiología de la Universidad Nacional Autónoma de México (UNAM) campus Juriquilla por proveer acceso a las imágenes de resonancia magnética utilizadas en esta investigación. Así mismo, a todos mis profesores por darme las herramientas necesarias para poder finalizar los estudios de posgrado y, en especial, al Dr. Saúl Tovar Arriaga por apoyarme en este trabajo y a lo largo de todo el proceso de la maestría, dándome su confianza y orientándome con su experiencia en todo momento.

Dirección General de Bibliotecas UAQ

Abstract

Segmenting vessels in brain images is a critical step for many medical interventions and diagnoses of illnesses. Recent advances in artificial intelligence provide better models, achieving a human-like level of expertise in many tasks. In this work, a new approach to segment Time-of-Flight Magnetic Resonance Angiography (TOF- MRA) images is presented, relying on fewer training samples than state-of-the-art methods.

A conditional generative adversarial network is proposed with an adapted generator based on a concatenated U-Net with a residual U-Net architecture (UUr-cGAN) to carry out blood vessel segmentation in TOF-MRA images, relying on data augmentation to diminish the drawback of having few volumes at disposal for training the model, while preventing overfitting by using regularization techniques.

The proposed model achieves 89.52 precision and 87.23 in Dice score on average from the cross-validated experiment for brain blood vessel segmentation tasks, which is similar to other state-of-the-art methods while using considerably fewer training samples.

UUr-cGAN extracts important features from small datasets while preventing overfitting compared to other CNN-based methods and still achieve a relatively good performance in image segmentation tasks such as brain blood vessels from TOF-MRA.

Dirección General de Bibliotecas UAQ

Resumen

La segmentación de vasos sanguíneos en imágenes cerebrales es un paso fundamental para muchas intervenciones médicas y diagnósticos de enfermedades. Los avances recientes en inteligencia artificial proporcionan mejores modelos, logrando un nivel de experiencia similar al humano en muchas tareas. En este trabajo, se presenta un nuevo enfoque para segmentar imágenes de angiografía por resonancia magnética de tiempo de vuelo (TOF-MRA), que se basa en menos muestras para el entrenamiento que los métodos del estado del arte.

Se propone una red generativa adversaria condicional con un generador adaptado basado en una U-Net, concatenada con una arquitectura U-Net residual (UUr-cGAN), para llevar a cabo la segmentación de vasos sanguíneos en imágenes TOF-MRA, basándose en el aumento de datos para disminuir la desventaja de disponer de pocos volúmenes para entrenar el modelo, además de evitar el sobreajuste mediante técnicas de regularización.

El modelo propuesto alcanza una precisión de 89,52 y una puntuación de Dice de 87,23 en promedio a partir del experimento de validación cruzada para las tareas de segmentación de los vasos sanguíneos cerebrales, lo que es similar a otros métodos del estado del arte y utiliza considerablemente menos muestras para el entrenamiento.

UUr-cGAN extrae características relevantes de pequeños conjuntos de datos, al tiempo que evita el sobreajuste en comparación con otros métodos basados en CNN y es capaz de lograr un rendimiento relativamente bueno en las tareas de segmentación de imágenes, como en el caso de los vasos sanguíneos cerebrales en TOF-MRA.

Índice general

Agradecimientos	
Abstract	
Resumen	II
Contenido	III
Lista de Figuras	V
Lista de Tablas	X
1. Introducción	1
1.1. Inteligencia Artificial	1
1.2. Visión por Computadora	2
1.3. Imágenes de Resonancia Magnética	2
1.3.1. Angiografías de Resonancia Magnética	4
1.4. Aneurismas Intracraneales	6
1.4.1. Sacular	7
1.4.2. Fusiforme	8
1.5. Neurocirugía	9
1.6. Hipótesis	10
1.7. Descripción del Problema	11

1.8. Objetivos	12
1.8.1. Objetivos Específicos	12
1.9. Estructura de la Tesis	13
2. Antecedentes	15
3. Metodología	20
3.1. Conjunto de datos	20
3.2. Marco Teórico	20
3.2.1. Redes Neuronales Convolucionales	20
3.2.2. Funciones de Activación en Redes Neuronales	26
3.2.3. Redes Generativas Adversarias	30
3.2.4. Redes Generativas Adversarias Condicionales	31
3.2.5. Entrenamiento de modelos convolucionales	32
3.3. Metodología	36
3.4. Implementación	40
3.4.1. Arquitecturas de redes neuronales convolucionales	43
3.4.2. Modelo propuesto UUr-cGAN	48
4. Discusión y Resultados	53
4.1. Diseño del Experimento	53
4.2. Hardware Utilizado	54
4.3. Resultados	55
4.4. Discusión	57
5. Conclusión	63
5.1. Trabajo Futuro	64
Apéndice A. Anexos	73
A.1. Visualización de volúmenes de MRA.	73
A.2. Requerimientos para Obtención de Grado	78

Índice de figuras

1.1. Representación de una imagen en una matriz bidimensional. a) La flecha en la imagen es representada por valores binarios dentro de la matriz. b) Representación binaria de la punta de la flecha. [2]	3
1.2. Imagen de resonancia magnética de la vista axial de un cerebro humano [3]. .	3
1.3. Escáner de resonancia magnética de 3T [3].	4
1.4. Vista coronal de los vasos sanguíneos de las piernas de un voluntario saludable usando un escáner de CE-MRA, amplificando el contraste mediante una inyección intravenosa [3].	5
1.5. a) Imagen de TOF-MRA bidimensional de la vista coronal de las piernas. b) Representación del stack de imágenes que conforman el modelo tridimensional de un escáner TOF-MRA. [3].	6
1.6. Vistas a) axial b) coronal y c) sagittal adquiridas mediante un escáner 3D TOF-MRA [3].	7
1.7. Visualización de un aneurisma sacular cerebral no reventado. a) 3D TOF-MRA mostrando el aneurisma donde apunta la flecha. b) Reconstrucción 3D a partir del escáner de MRA mostrando el aneurisma detectado [9].	8
1.8. A) Imagen de resonancia magnética mostrando la vista axial, donde la flecha negra indica un bulto en la arteria cerebral anterior derecha. B) Angiografía de resonancia magnética, donde la flecha indica la dilatación del vaso sanguíneo de tipo fusiforme [12].	9

1.9. Mapa de riesgo calculado para una trayectoria con punto final en el ventrículo lateral [15].	10
3.1. Resonador Phillips Achieva 3T [?].	21
3.2. Vistas sagital, coronal y axial de un volumen del conjunto de datos junto al mapa de segmentación correspondiente.	21
3.3. Arquitectura general de una red neuronal convolucional.	23
3.4. Kernel simple para la detección de líneas horizontales.	24
3.5. Matriz de dos dimensiones (imagen) en convolución con kernel de 3x3.	24
3.6. Padding para agregar margen a la imagen de entrada y así evitar la pérdida de información al aplicar filtros convolucionales [57].	25
3.7. Convolución sobre todo el espacio de la imagen (izquierda). Convolución con stride de 2 (derecha).	26
3.8. Función max pooling con disminución de resolución.	26
3.9. Comportamiento de la función de activación ReLU.	28
3.10. Función de activación ReLU (izquierda) vs Leaky ReLU (Derecha).	28
3.11. Función de activación sigmoide [58].	29
3.12. Función de activación tanh [58].	30
3.13. Arquitectura básica de una red generativa adversaria [58].	31
3.14. Arquitectura de una red generativa adversaria condicional[59].	32
3.15. Algoritmo de backpropagation para entrenamiento de redes neuronales [57].	33
3.16. Propuesta metodológica del presente proyecto.	37
3.17. Algoritmo de ecualización de histograma adaptativo limitada por contraste (CLAHE) [48].	41
3.18. Ecualización por histograma implementada en imágenes de microscopía [37]	42
3.19. Visualización del preprocesamiento realizado a imágenes de TOF-MRA. Muestra del conjunto de datos (izquierda), máscara obtenida a partir de la operación de cerradura (centro), resultado de la multiplicación de la imagen de muestra y la máscara (derecha).	43

3.20. Comparación entre el volumen de TOF-MRA previo y después de la reducción de ruido. Volumen original (superior e inferior izquierda), resultado de la aplicación de la máscara obtenida para cada imagen del volumen (superior e inferior derecha).	44
3.21. Aumento de datos al aplicar acercamiento y transformaciones elásticas en regiones aleatorias de la imagen original	45
3.22. Aumento de datos al aplicar un filtro espejo a cada imagen	45
3.23. Arquitectura CNN de la Unet [31]	46
3.24. Arquitectura basada en Unet con bloques residuales [39].	46
3.25. Arquitectura con dos CNN concatenadas [34].	47
3.26. Modelo UUr-cGAN propuesto	48
3.27. Arquitectura del bloque residual dentro de la segunda etapa del generador del modelo cGAN propuesto. Una conexión saltada es usada para preservar información del mapa de características de la entrada del bloque a las capas más profundas, combinándolas con una operación de suma.	50
3.28. Modelos generador y discriminador basados en [31, 34, 50, 55] para la cGAN propuesta. El generador se inspira en la concatenación de dos modelos tipo U-Net, donde el segundo implementa bloques residuales, con el objetivo de propagar información a las capas más profundas del modelo, preservando detalles más finos entre los mapas de características.	51
4.1. Ejemplo de la segmentación de vasos sanguíneos cerebrales. a) Imágenes MRA, b) Segmentación objetivo, c) Predicción del modelo propuesto, and d) Error de segmentación (diferencia entre la segmentación objetivo y la predicción).	56
4.2. Comparación de los volúmenes de MRA segmentados con el método propuesto. Volúmenes MRA (izquierda). Segmentación objetivo (centro). Predicción (derecha).	60

4.3.	Mapas de segmentación obtenidos con cada modelo evaluado, descritos en tabla 4.3. La visualización 3D es generada con el software Aliza Medical Imaging ©, Bonn, Germany, and DICOM Viewer para la segmentación objetivo y predicciones. a) Segmentación objetivo, b) U-Net [31], c) Pix2Pix [50], d) U-Nets concatenadas [31, 55, 34], y e) cGAN propuesta.	61
4.4.	Mapas de error obtenidos de la diferencia entre el volumen de segmentación objetivo y la predicción de cada método evaluado, donde las regiones oscuras representan pixeles mal clasificados (entre menos mejor). a) U-Net, b) U-Net doble, c) Pix2Pix, d) Método propuesto.	62
A.1.	Visualización del volumen 1 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.	74
A.2.	Visualización del volumen 2 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.	75
A.3.	Visualización del volumen 3 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.	76
A.4.	Visualización del volumen 4 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.	77
A.5.	Constancia de la realización del examen de manejo de la lengua por la Facultad de Lenguas y Letras de la UAQ.	78

A.6. Carta de aceptación del artículo realizado para la revista "La Mecatrónica en México". 79

Dirección General de Bibliotecas UAQ

Índice de tablas

2.1. Métodos del estado del arte para segmentación de vasos sanguíneos en imágenes de MRA, así como la técnica en la que están basados, junto al modelo propuesto.	19
3.1. Entrenamiento de los modelos convolucionales sobre el conjunto de datos de microscopía.	47
4.1. Selección de hiper parámetros para los modelos evaluados, donde G representa el modelo generador y D el modelo discriminador.	54
4.2. Métricas evaluadas sobre los modelos entrenados en cada iteración del experimento k-Fold.	55
4.3. Resultado del experimento k-Fold para los modelos evaluados.	57
4.4. Índice Dice y tamaño del conjunto de datos para distintos métodos del estado del arte para segmentación de vasos sanguíneos cerebrales, así como del modelo propuesto.	58

Acrónimos

- ADAM – Estimación de Momento Adaptativo (del inglés Adaptive Moment Estimation)
- API – Interfaz de Programación de Aplicaciones (del inglés Application Programming Interface)
- CAE – Auto Encoder Convolutacional (del inglés Convolutional Auto Encoder)
- CDF – Función de Distribución Acumulada (del inglés Cumulative Distribution Function)
- CE-MRA – Angiografía de Resonancia Magnética con Contraste Mejorado (del inglés Contrast Enhanced Magnetic Resonance Angiography)
- cGAN - Red Generativa Adversaria Condicional (del inglés Conditional Generative Adversarial Network)
- CLAHE – Ecuación por Histograma Adaptativo por Contraste (del inglés Contrast Limited Adaptive Histogram Equalization)
- CNN – Red Neuronal Convolutacional (del inglés Convolutional Neural Network)
- CoW – Círculo de Willis (del inglés Circle of Willis)
- CPU – Unidad de Procesamiento Central (del inglés Central Processing Unit)
- DICOM – Comunicaciones e Imágenes Digitales en Medicina (del inglés Digital Imaging and Communications in Medicine)
- FCN – Red Totalmente Conectada (del inglés Fully Connected Network)
- GAN – Red Generativa Adversaria (del inglés Generative Adversarial Network)
- GPU – Unidad de Procesamiento Gráfico (del inglés Graphics Processing Unit)
- IA – Inteligencia Artificial
- MAE – Error Medio Absoluto (del inglés Mean Absolute Error)

MRA – Angiografía de Resonancia Magnética (del inglés Magnetic Resonance Angiography)

MRI – Imágenes de Resonancia Magnética (del inglés Magnetic Resonance Imaging)

ReLU – Unidad Lineal Rectificada (del inglés Rectified Linear Unit)

RMSProp – Propagación de la Raíz Cuadrática Media (del inglés Root Mean Square Propagation)

ROI – Región de Interés (del inglés Region of Interest)

SWI – Imágenes de Sensibilidad Ponderada (del inglés Susceptibility weighted imaging)

T – Teslas

Tanh – Tangente hiperbólica

TOF – Tiempo de Vuelo (del inglés Time-of-flight)

TOF-MRA – Angiografía de Resonancia Magnética al Tiempo de Vuelo (del inglés Time-of-flight Magnetic Resonance Angiography)

UIA – Aneurisma Intracraneal No Reventado (del inglés Unruptured Intracranial Aneurysm)

VRAM – Memoria de Video (del inglés Video Random Access Memory)

Dirección General de Bibliotecas UAQ

Introducción

1.1. Inteligencia Artificial

El campo de la inteligencia artificial (IA), iniciado formalmente en 1950, tiene como objetivo el entendimiento de las entidades inteligentes desde su construcción hasta la manera en que se comportan e interactúan con el ambiente, con el fin de comprender de mejor manera nuestra propia inteligencia. La IA es una herramienta con un gran impacto en nuestro día a día debido al sin fin de aplicaciones desarrolladas y en desarrollo con el objetivo de mejorar la calidad de vida de los seres vivos, así como la preservación del medio ambiente [1].

La IA intenta responder el como un pequeño cerebro o procesador, ya sea biológico o electrónico, pueda ser capaz de percibir, entender, predecir y manipular un mundo mucho más grande y complicado que sí mismo [1]. Partiendo de esta interrogante, los investigadores dentro del área han partido desde la inteligencia que mejor conocemos, la nuestra, intentando replicar estructuras neuronales similares a las de nuestros cerebros, así como nuestros sentidos, como la visión para la percepción de imágenes y el oído para el procesamiento de sonidos y voz. De igual manera, la forma en que aprendemos y estructuramos el conocimiento ha servido, en gran manera, de inspiración para algoritmos de aprendizaje automático, así como para la transferencia de información por medio del lenguaje y la interpretación de este, lo que ha dado lugar a tecnologías como lo son los asistentes de voz, análisis de sentimien-

tos, entre otras, para crear un vínculo entre máquinas y humanos de una forma más simple e intuitiva.

1.2. Visión por Computadora

Para comprender el funcionamiento y representación de imágenes en dispositivos electrónicos, es necesario hablar del sistema visual humano, el cual es el más sofisticado hasta donde sabemos. El sistema visual humano comienza con los ojos, órganos capaces de percibir ondas de luz dentro de un espectro determinado, conocido como el de luz visible. Los principales componentes del ojo humano son la córnea, pupila, lente, retina y el nervio óptico, los cuales en conjunto son capaces de representar imágenes a partir del espectro de luz de entrada, las cuales son enviadas al cerebro para su procesamiento [2].

Las cámaras digitales se han basado en este proceso, cuya representación de imágenes se reduce a un elemento básico, llamado pixel (del inglés: *picture element*), ordenados en un arreglo rectangular de $M \times N$ dimensiones, donde cada uno posee una coordenada única, como se representa en la figura 1.1

1.3. Imágenes de Resonancia Magnética

Las imágenes de resonancia magnética (MRI) ofrecen distintos beneficios en comparación con otras modalidades, usualmente utilizadas para la detección de tumores y anomalías en tejidos. Las MRI obtienen señales usando un campo magnético y energía de radiofrecuencia, sin el uso de materiales radiactivos, las cuales permiten diferenciar entre distintos tipos de tejidos suaves. Por otra parte, esta modalidad ofrece la ventaja de poder obtener imágenes tomográficas desde cualquier plano sin la necesidad de cambiar de postura al paciente o al equipo de adquisición de las imágenes. La figura 1.2 es una muestra de una imagen por resonancia magnética cerebral.

Los sistemas de MRI (figura 1.3) usan un fuerte campo magnético, con una intensidad

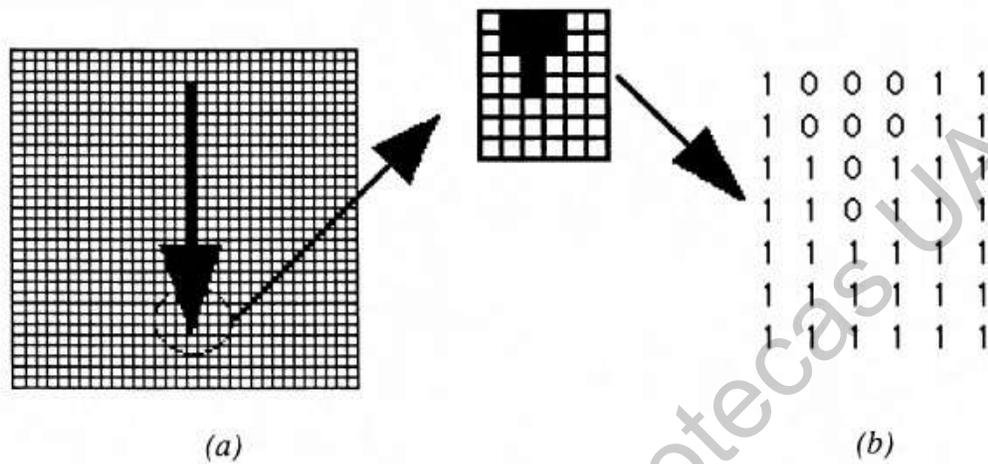


Figura 1.1: Representación de una imagen en una matriz bidimensional. a) La flecha en la imagen es representada por valores binarios dentro de la matriz. b) Representación binaria de la punta de la flecha. [2]

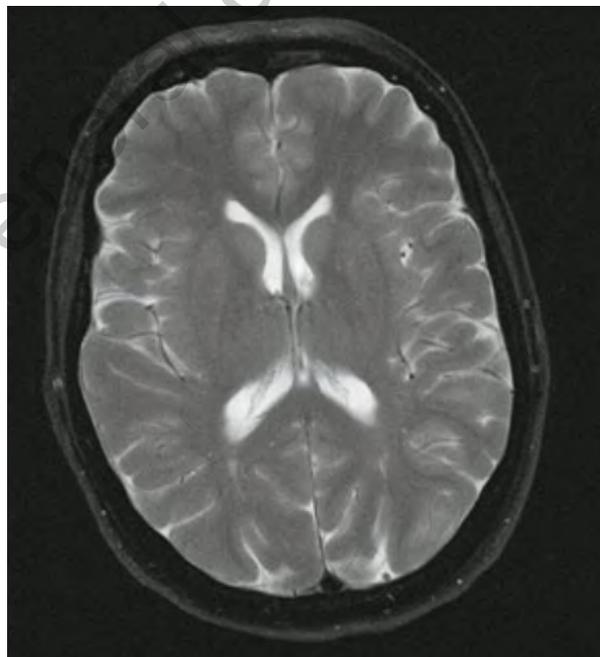


Figura 1.2: Imagen de resonancia magnética de la vista axial de un cerebro humano [3].

estable durante el tiempo y a lo largo de todo el volumen del dispositivo, el cual usualmente es de entre un diámetro de 60 y 70 cm, mientras que la intensidad del campo magnético, medido en Teslas (T), varía de entre 1.5T a 3T [3].



Figura 1.3: Escáner de resonancia magnética de 3T [3].

1.3.1. Angiografías de Resonancia Magnética

Las modalidades de angiografía de resonancia magnética (MRA), basadas en MRI, más utilizadas son CE (contrast-enhanced) y TOF (Time-of-Flight).

Contrast-Enhanced MRA

Los métodos de Contrast-enhanced MRA (CE-MRA) han sido adoptados ampliamente por su facilidad de uso, velocidad, confiabilidad y robustez en producir imágenes de gran calidad de regiones vasculares, reemplazando otras modalidades basadas en rayos X que se

usaban con anterioridad para la visualización de regiones como las arterias vertebrales, renales y de las extremidades inferiores del cuerpo humano. Las técnicas de CE-MRA localizan diferencias en las señales dadas entre la sangre y regiones estacionarias, como lo son los tejidos, al manipular la magnitud de la magnetización. Dicha magnetización se consigue al variar ciertos parámetros del resonador y al inyectar al paciente un material intravenoso, usualmente átomos de gadolinium, el cual provoca este incremento [3]. La figura 1.4 muestra una imagen obtenida mediante esta técnica de las piernas de un paciente.



Figura 1.4: Vista coronal de los vasos sanguíneos de las piernas de un voluntario saludable usando un escáner de CE-MRA, amplificando el contraste mediante una inyección intravenosa [3].

Time-of-Flight MRA

Así como las técnicas de CE-MRA, las Time-of-Flight MRA (TOF-MRA) derivan el contraste entre el flujo de la sangre y los tejidos estacionarios, asumiendo que la magnetización

del flujo sanguíneo es mayor. Como ventaja, esta modalidad no requiere de una inyección intravenosa de algún componente contrastante, únicamente se depende de la propiedad del flujo sanguíneo y el ajuste de los parámetros del resonador, donde las señales más débiles o desvanecidas se filtran, las cuales son correspondientes a los tejidos estacionarios. La adquisición de estas imágenes se puede realizar en dos y en tres dimensiones, donde la primera corresponde a un stack de imágenes para formar un modelo tridimensional del área escaneada [3], cuya representación se muestra en la figura 1.5 para un stack correspondiente a las piernas de un paciente, y en la figura 1.6 para un escáner cerebral 3D.

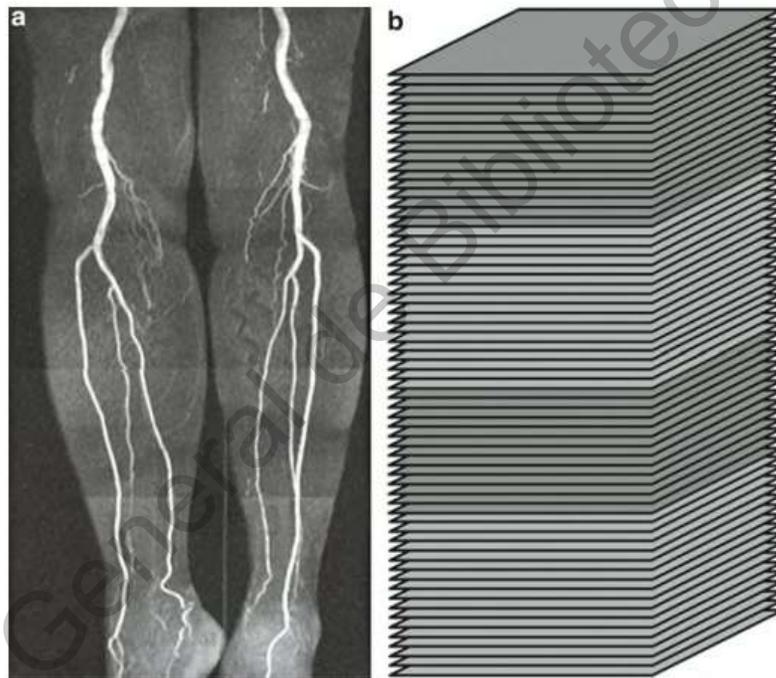


Figura 1.5: a) Imagen de TOF-MRA bidimensional de la vista coronal de las piernas. b) Representación del stack de imágenes que conforman el modelo tridimensional de un escáner TOF-MRA. [3].

1.4. Aneurismas Intracraneales

Un aneurisma intracraneal (o aneurisma cerebral) es una región débil o delgada en una arteria del cerebro que se infla y se llena de sangre, provocando presión en las regiones aledañas. La presión de la sangre sobre el aneurisma puede provocar una ruptura del mismo,

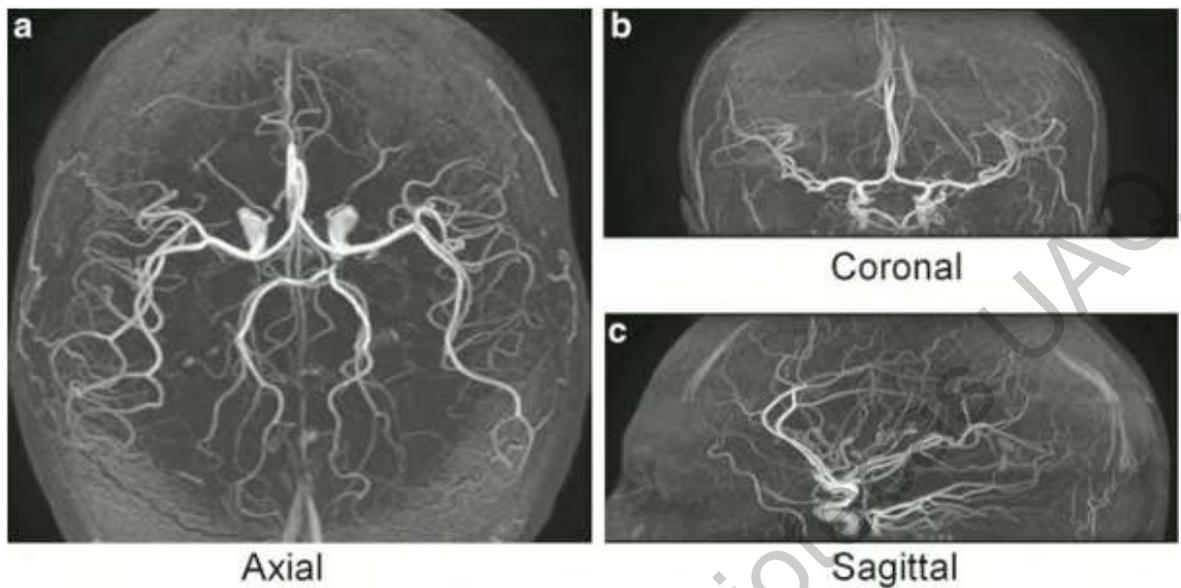


Figura 1.6: Vistas a) axial b) coronal y c) sagittal adquiridas mediante un escáner 3D TOF-MRA [3].

causando serios problemas de salud como derrames y daño cerebrales, coma e incluso la muerte [4].

Los aneurismas intracraneales no reventados (UIAs) son prevalentes en el 3% de la población adulta, siendo cada vez más comúnmente detectados gracias a que los escáneres de imágenes cerebrales son cada vez más populares [5, 6].

1.4.1. Sacular

Los aneurismas saculares son deformaciones en un vaso sanguíneo con forma de una bolsa pequeña, formados principalmente en los puntos de bifurcación de las arterias, que se desarrollan cuando las paredes del vaso sanguíneo se debilitan ante la resistencia de la presión sanguínea y se dilate [7, 8].

Este tipo de aneurisma representa entre un 66% y 98% del total de casos de aneurismas intracraneales y no son considerados un problema congénito sino desarrollado durante la vida, principalmente en adultos debido a la relación que tienen con la hipertensión, tabaquismo y alcoholismo [8].

En la figura 1.7 se muestra un aneurisma cerebral sacular, a partir de un escáner de TOF-MRA.

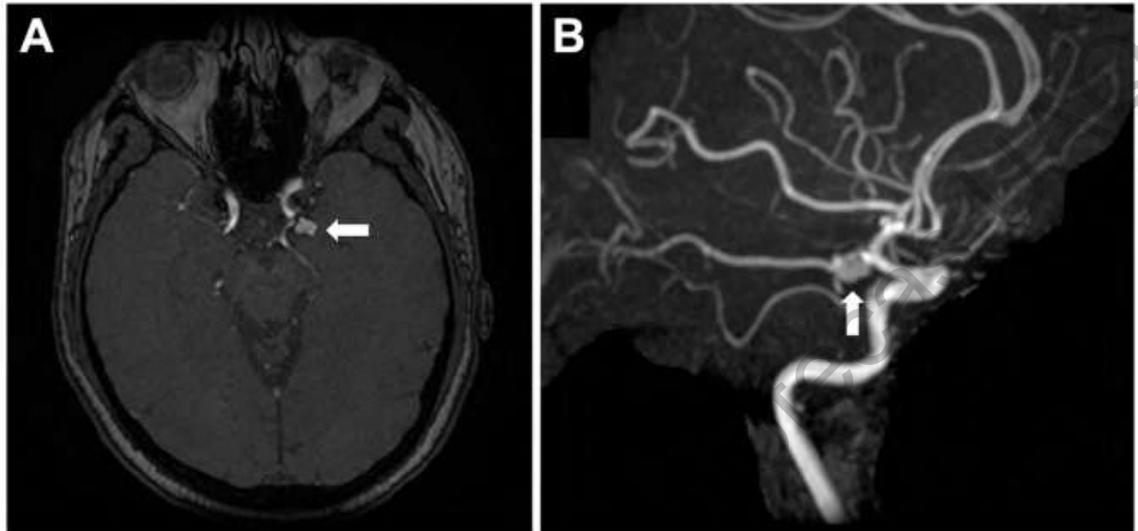


Figura 1.7: Visualización de un aneurisma sacular cerebral no reventado. a) 3D TOF-MRA mostrando el aneurisma donde apunta la flecha. b) Reconstrucción 3D a partir del escáner de MRA mostrando el aneurisma detectado [9].

1.4.2. Fusiforme

Los aneurismas fusiformes son segmentos dilatados de los vasos sanguíneos, caracterizados por no presentar un “cuello” bien definido en su formación, en contraste con los aneurismas saculares [8].

Este tipo de aneurisma puede ser provocado de manera congénita, adquirido o incluso iatrogénico, lo cual indica que puede ser la consecuencia de alguna otra intervención médica y pueden ocurrir en prácticamente cualquier región cerebral. También han sido asociados con la aterosclerosis, padecimiento en el cual placas de grasa, colesterol, calcio o alguna otra sustancia, se aloja en el vaso sanguíneo y provoca la obstrucción del mismo [8, 10, 11].

En la figura 1.8 se muestra un aneurisma cerebral fusiforme, a partir de un escáner de TOF-MRA.

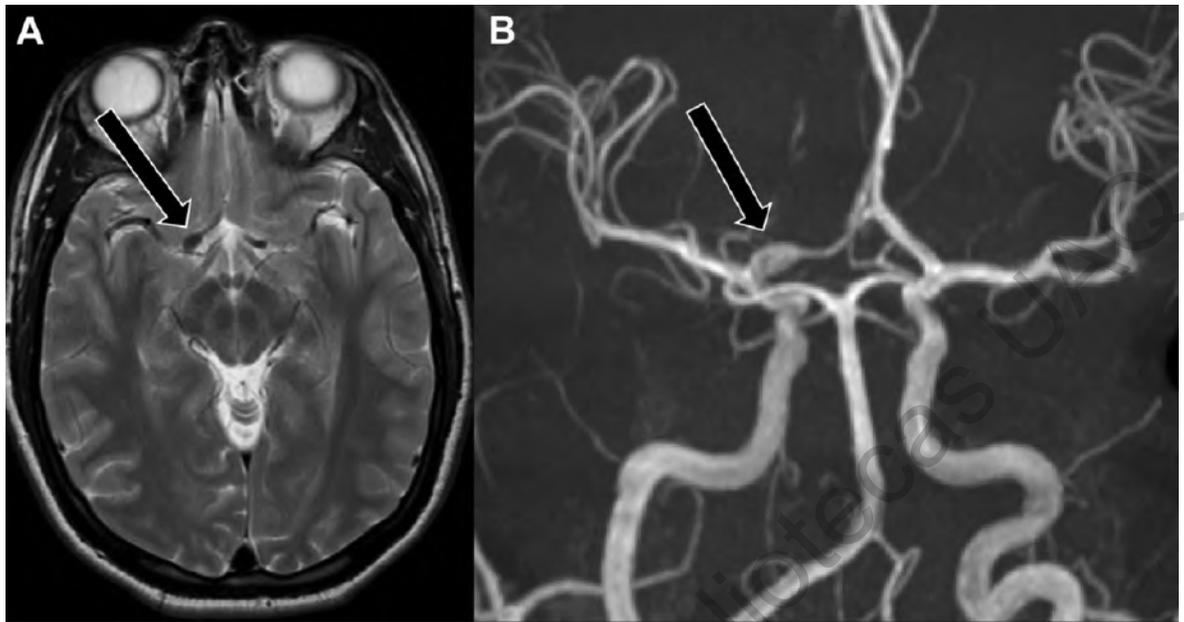


Figura 1.8: A) Imagen de resonancia magnética mostrando la vista axial, donde la flecha negra indica un bulto en la arteria cerebral anterior derecha. B) Angiografía de resonancia magnética, donde la flecha indica la dilatación del vaso sanguíneo de tipo fusiforme [12].

1.5. Neurocirugía

Las intervenciones quirúrgicas cerebrales mínimamente invasivas guiadas a través de imagenología médica han ganado relevancia en las rutinas clínicas en los tiempos recientes, realizando una correcta detección de tumores, lesiones y estructuras cerebrales con el fin de planificar una trayectoria apropiada, provocando el menor daño posible al paciente durante la intervención y la inserción de la herramienta por parte del médico especialista [13, 14, 15, 16, 17].

La planificación de la intervención es una tarea compleja incluso para cirujanos con experiencia por tres motivos principales [15, 18]: 1) la dificultad de hacer una reconstrucción tridimensional de las estructuras cerebrales a partir de tomografías o angiografías de dos dimensiones, 2) debido a la complejidad de la morfología cerebral y 3) por la dificultad que conlleva el predecir los posibles daños ocasionados al paciente después de la intervención quirúrgica.

De forma análoga a la neurocirugía, la estimulación profunda cerebral requiere de una planificación similar, donde se tiene un punto objetivo en la región del tejido cerebral a tratar y el punto de entrada del electrodo, al unirlos con el punto meta, produce la trayectoria a seguir, por lo que el punto de entrada es definitorio y complejo de seleccionar dada la cantidad de puntos de entrada posible [27, 25].

Para un posterior etiquetado de las zonas de riesgo, es de suma importancia contar con una clasificación de las estructuras que representan áreas de peligro, por las posibles complicaciones que pueden llevar consigo tanto al momento de la intervención como las secuelas que pueden provocarse en el paciente. Una de las regiones más importantes a evitar son los vasos sanguíneos, ya que el daño a uno de estos durante la intervención puede provocar hemorragias difíciles de controlar que pueden llevar a consecuencias fatales. En la figura 1.9, se muestra un modelo tridimensional de la región craneal de un paciente, con su respectivo mapa de riesgo asociado [15].

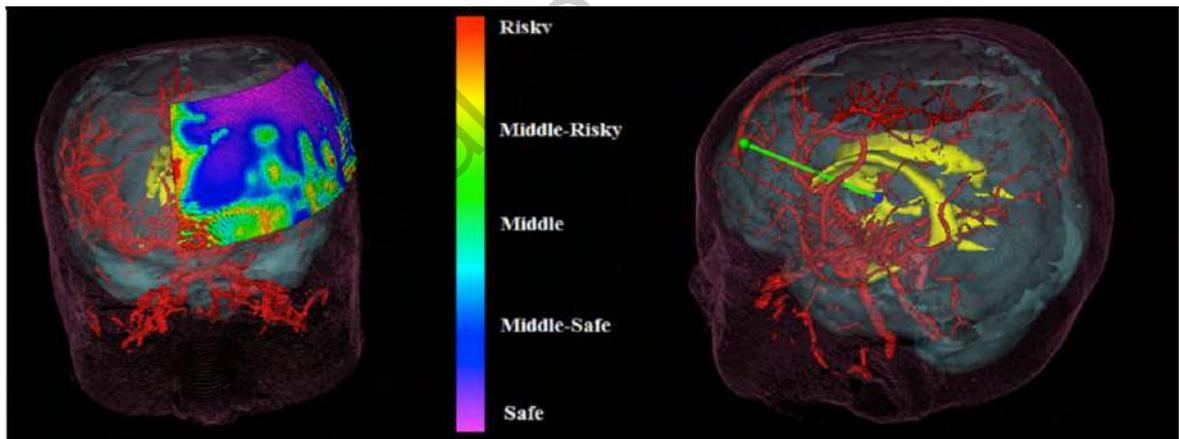


Figura 1.9: Mapa de riesgo calculado para una trayectoria con punto final en el ventrículo lateral [15].

1.6. Hipótesis

Los modelos convolucionales pueden ser entrenados con el fin de realizar la segmentación de vasos sanguíneos cerebrales en imágenes médicas, con el fin de proveer una herramienta

para especialistas y radiólogos en la detección de estas estructuras de interés para una variedad de aplicaciones.

Algunas de estas, como las intervenciones quirúrgicas, donde es de suma importancia la correcta evaluación de una trayectoria apropiada ya que, en caso contrario, se pueden presentar consecuencias fatales para el paciente, ya sea desde una hemorragia hasta la muerte, o incluso situaciones desconocidas tanto a corto como a largo plazo debido al daño de estructuras cerebrales. Así mismo, puede ser de ayuda en el diagnóstico de diversos padecimientos cerebrovasculares, como lo son los aneurismas intracraneales, con el fin de realizar una detección oportuna y facilitar al especialista la tarea de visualizar únicamente las regiones de interés, a partir de las angiografías obtenidas del paciente. Es viable la implementación de redes neuronales convolucionales para asistir en la segmentación de vasos sanguíneos en imágenes médicas cerebrales, para acelerar dicho proceso en comparación con la segmentación manual por parte de un especialista, obteniendo resultados similares.

1.7. Descripción del Problema

En los últimos años, la neurocirugía asistida por computadora ha ganado gran impacto dentro de la comunidad científica debido a lo prometedora que es, abordando las principales complicaciones que se pueden presentar dentro de una intervención quirúrgica.

La toma de la decisión a tomar en una neurocirugía es un proceso complicado para el cirujano, ya que hay una gran cantidad de factores que se deben de considerar, cada paciente es diferente y padecimiento que presenta lo es también, por lo que es complicado generalizar los casos para definir puntos de entrada, provocando el menor daño posible. Es difícil para un médico, aunque sea experto en el área, el visualizar un modelo en tres dimensiones a partir de imágenes en dos, definir la trayectoria óptima dentro de la gran cantidad de posibilidades y evaluar el riesgo, prediciendo que no se dañará a estructuras cerebrales que pueden ocasionar daños irreparables para el paciente.

De esta necesidad surge el objetivo de esta investigación, aportando con una herramienta

capaz de asistir al neurocirujano con la importante toma de decisión de la trayectoria a seguir y así lograr evitar consecuencias negativas al paciente y mejorar los tiempos de recuperación del mismo debido a una intervención mínima y precisa.

El diagnóstico de diversos padecimientos cerebrovasculares puede ser un procedimiento tardado para un especialista, desde el momento en que se propone la necesidad de un escaneo cerebral, hasta el punto en el que se obtienen las imágenes, además de requerir el filtrado y la utilización de diversas herramientas de visualización para las imágenes obtenidas, ya sea en dos o tres dimensiones, donde es difícil realizar una correcta selección del área de interés, los vasos sanguíneos, al tener ruido por la propia técnica de imagenología médica y otros factores, como la obstrucción de unas regiones cerebrales sobre otras, al estar sobrepuestas. Para facilitar la visualización del resultado de los escaneos cerebrales de los pacientes al especialista, la detección de los vasos sanguíneos en imágenes médicas, aislando regiones de no interés, es de suma importancia, para hacer un rápido y más certero diagnóstico.

1.8. Objetivos

Planificar e implementar un método de inteligencia artificial basado en redes neuronales convolucionales, el cual debe ser capaz de realizar una correcta segmentación de imágenes médicas cerebrales para la detección de vasos sanguíneos, con el fin de proveer una herramienta para especialistas en el diagnóstico y tratamiento de diversos padecimientos cerebrovasculares.

1.8.1. Objetivos Específicos

- Diseñar, implementar y evaluar una arquitectura basada en redes neuronales convolucionales para la segmentación de imágenes médicas cerebrales.
- Obtener imágenes segmentadas de vasos sanguíneos cerebrales a partir de imágenes DICOM.
- Comparar la arquitectura propuesta con otros métodos con el mismo objetivo.

- Proveer una herramienta de apoyo en el diagnóstico de diversos padecimientos cerebrales.

1.9. Estructura de la Tesis

La investigación mostrada en el documento presente es referente a la detección de estructuras cerebrales en imágenes médicas a partir de la segmentación de estas, evaluando diversas técnicas para realizar este procedimiento.

Las intervenciones quirúrgicas cerebrales son de alto riesgo de manera inherente. Los motivos por los que se llevan a cabo son muy variados, pero coinciden en que el procedimiento debe ser realizado con el mayor cuidado posible, para no poner en riesgo la integridad del paciente.

La neurocirugía de cerradura es un proceso en el cual la intervención quirúrgica es realizada llevando a cabo una craneotomía con la menor área posible, definiendo un punto de acceso que conlleve al punto meta de una manera directa.

La correcta selección de la trayectoria a realizar para el procedimiento quirúrgico es la tarea más compleja en la parte de planificación de la intervención, donde al no contar con imagenología médica correctamente etiquetada puede provocar daños irreversibles por lo que una correcta identificación de las estructuras cerebrales en la región de interés del neurocirujano es de vital importancia para llevar al mínimo el daño ocasionado al paciente y hacer de la cirugía de cerradura, que es mínimamente invasiva, aún más segura.

La segmentación automática de estructuras cerebrales es una herramienta que puede asistir al neurocirujano para hacer la planificación de la trayectoria de una manera más sencilla y segura.

Existen diversas técnicas de segmentación de imágenes médicas, en este trabajo se explorarán algunas y se propondrán puntos de mejora de las mismas.

La tesis se organiza de la siguiente manera:

- El capítulo 2 describe algunos de los trabajos del estado del arte, referente a la segmen-

tación de vasos sanguíneos y las técnicas empleadas, así como las posibles aplicaciones.

- El capítulo 3 introduce algunas de las técnicas implementadas en los métodos del estado del arte, además de proveer lo necesario para la comprensión de la técnica propuesta en este documento.
- El capítulo 4 muestra los detalles y lo que inspiró al desarrollo del método propuesto, junto con las pruebas realizadas y sus resultados.
- El capítulo 5 aborda las conclusiones del método propuesto y sus aplicaciones, así como el trabajo futuro y áreas de oportunidad relacionados al mismo.

Antecedentes

Múltiples trabajos se han presentado en relación a la segmentación de estructuras cerebrales, utilizando una variedad de técnicas de imagenología médica. Así mismo, existen diversas arquitecturas basadas de redes neuronales convolucionales con resultados aceptables. A continuación, se describe el estado del arte referente al tema de investigación del presente documento.

La cirugía asistida por computadora es un campo con un gran desarrollo en los últimos años, ya que tiene la premisa de revolucionar la forma en que las cirugías se llevan a cabo, apoyando al cirujano en las intervenciones quirúrgicas como herramienta para asegurar una mayor precisión en las trayectorias a realizar. Con el objetivo de minimizar el daño producido por estas intervenciones, es de vital importancia minimizar el área corporal que se verá afectada por la cirugía, término conocido cirugía de mínima invasión, teniendo como objetivos evitar complicaciones por dañar múltiples regiones que pueden llevar a consecuencias graves para el paciente y reducir los tiempos de recuperación. Las intervenciones cerebrales, siendo de las más complejas debido al riesgo que conllevan, sugieren un necesario uso de técnicas de cirugía de cerradura donde, a través de una pequeña incisión, es posible llevar a cabo la intervención sin la necesidad de una cirugía a cráneo abierto. Para estas metodologías, además de uno o varios especialistas, es necesario contar con algoritmos capaces de evaluar diversas

trayectorias por las cuales realizar la intervención con el riesgo mínimo para el paciente. Herramientas como esta se basan en el cálculo de riesgo a partir de ciertos parámetros, como la cercanía con distintas estructuras cerebrales y, así, seleccionar la ruta óptima para provocar el daño mínimo, aunque como ya se mencionó, es sólo una herramienta de asistencia al cirujano, ya que este cuenta con la decisión final de la selección de la trayectoria [21].

Los algoritmos de aprendizaje automático pertenecen a un área especializada dentro de la Inteligencia Artificial, los cuales dotan de “inteligencia” a las computadoras encontrando patrones entre los datos. Se ha propuesto una gran cantidad de algoritmos que intentan simular la inteligencia humana. Dentro del campo de la visión artificial, ha surgido una arquitectura de red neuronal especializada, las redes neuronales convolucionales (del inglés Convolutional Neural Network (CNN)), que intentan emular el sistema visual de los seres vivos. Las CNN son consideradas una de las mejores técnicas para el procesamiento de imágenes, ya sea para el reconocimiento y/o detección de objetos, segmentación, entre otras. La arquitectura de una CNN está dividida en distintas capas; convolución, función de activación y sub muestreo. Cada capa realiza múltiples transformaciones a partir de un filtro de convolución. La convolución es capaz de abstraer ciertas características de las imágenes para obtener patrones dentro de las mismas [22].

Se han propuesto diversos métodos para la extracción de información sobre regiones cerebrales de interés, como lo son los vasos sanguíneos, a partir de la segmentación de imágenes médicas, afrontando la problemática que conlleva la correcta segmentación de imágenes médicas.

Debido a la naturaleza de las técnicas de adquisición de imágenes de angiografía de resonancia magnética cerebrales, los volúmenes de estas son representados por voxeles para conformar estructuras 3D de la cabeza del paciente, donde imágenes adyacentes comparten ciertas propiedades debido a la cercanía de la misma, dependiendo también de la resolución del resonador para cada voxel. Esta propiedad es utilizada para extraer características relevantes del ROI del volumen por [30], siendo estas los vasos sanguíneos cerebrales, a partir de un autoencoder convolucional tridimensional cuya principal aportación es la reducción

del ruido de las imágenes de MRA mediante dicha arquitectura para una extracción de vasos sanguíneos más simple, cuyo resultado es un mapa de segmentación de estas estructuras. Este método reporta un índice Dice Score de 73.71, contando con un conjunto de datos de 49 volúmenes de MRA.

Las arquitecturas encoder-decoder han mostrado un incremento en el rendimiento de los modelos convolucionales. El modelo “U-Net” propuesto por Ronneberger et al. [31] ha establecido un punto de inflexión en la segmentación de imágenes para el área médica, demostrando que una red neuronal convolucional profunda puede ser entrenada en conjuntos de datos relativamente pequeños, logrando buenos resultados en comparación con otras arquitecturas convolucionales. Esto es de relevancia debido a la dificultad de acceder a conjuntos de datos etiquetados por especialistas, los cuales suelen contener un número reducido de elementos.

Considerando lo anterior, diversas arquitecturas convolucionales basadas en la “U-Net” se han propuesto para diversas aplicaciones, dentro y fuera del área médica. Partiendo de esto, Livne et al. [32] proponen su modelo.

Teikari et al. [37] desarrollaron un modelo híbrido 2D-3D de red neuronal convolucional para la segmentación de vasos sanguíneos en imágenes de microscopía de fotones, utilizando únicamente un CPU como acelerador del framework, mostrando ciertos modelos de redes convolucionales, optimizados de manera apropiada, pueden no requerir de hardware más especializado, como GPUs, para la etapa de entrenamiento.

En el trabajo presentado por Tetteh et al. [38], depende de transferencia de aprendizaje entre modelos y conjuntos de datos para realizar tres tareas; la localización de la línea central y puntos de bifurcación en vasos sanguíneos, así como la segmentación de los mismos. Para esto, se realiza un pre-entrenamiento con un conjunto de datos sintético para el modelo 3D convolucional propuesto, realizando un entrenamiento posterior y validación en angiografías de resonancia magnética de ratas y humanos.

Utilizando aumento de datos, Zhang et al. [39] aplican una transformación de reflejo a cada elemento del conjunto de datos para generar más ejemplos, esto dada la desventaja de contar con conjuntos de datos reducidos para el entrenamiento de modelos de aprendizaje pro-

fundo. De manera adicional, los autores del trabajo agregan ruido Gaussiano a los ejemplos para provocar una invarianza a intensidades absolutas del modelo durante el entrenamiento del mismo. Así mismo, proponen una función de pérdida híbrida para lidiar con el desbalance de clases inherente en los mapas de segmentación en imágenes médicas con una sola clase. Como resultado, logran un índice Dice de 75.6 en segmentación de vasos sanguíneos en imágenes de susceptibilidad ponderada (SWIs).

Considerando la variabilidad que conlleva el flujo sanguíneo al realizar la adquisición de imágenes cerebrales, Kandil et al. [40] dividen sus volúmenes de MRA en dos partes, por encima y debajo del Círculo de Willis (CoW), entrenando con ambas partes su modelo 3D CNN, obteniendo un índice Dice de 84.37 como resultado de la segmentación.

Zhao et al. [41] desarrollan una arquitectura que extrae estructuras candidatas a ser vaso sanguíneo de imágenes de MRA como parte de un pre-procesamiento. Una red neuronal totalmente conectada actúa como clasificador de las regiones extraídas previamente, dando como resultado la probabilidad de cada una de estas de ser vasos sanguíneos.

En la tabla 2.1 se muestra un resumen de las aportaciones antes descritas, incluyendo el tipo de imagen y base de datos utilizada, así como la técnica implementada para realizar la segmentación.

Tabla 2.1: Métodos del estado del arte para segmentación de vasos sanguíneos en imágenes de MRA, así como la técnica en la que están basados, junto al modelo propuesto.

Método	Tipo de Imagen	Técnica	Volúmenes en dataset
Chen et al. [30]	Angiografía de Resonancia Magnética	CAE	10
Hu et al. [28]	Microscopía de Fluorescencia	CNN + FCN	25
Phellan et al. [29]	Angiografía de Resonancia Magnética	CNN + FCN	5
Tetteh et al. [38]	Angiografía de Resonancia Magnética	CNN	40
Zhang et al. [39]	Imágenes de Susceptibilidad Ponderada	CNN	10
Kandil et al. [40]	Angiografía de Resonancia Magnética	CNN + FCN	30
Zhao et al. [41]	Angiografía de Resonancia Magnética	FCN	30
Teikari et al. [37]	Microscopía Multifotón	CNN	12
Livne et al. [32]	Angiografía de Resonancia Magnética	CNN	66

Metodología

3.1. Conjunto de datos

Las angiografías de resonancia magnética utilizadas en este trabajo han sido proveídas por el Instituto de Neurobiología de la Universidad Nacional Autónoma de México (UNAM), las cuales consisten en 4 volúmenes obtenidas de 4 pacientes por un resonador Phillips Achieva 3T (figura 3.1). Cada volumen contiene 200 imágenes con una resolución de 560×560 píxeles. Junto con el conjunto de datos, se incluye el mapa de segmentación de los vasos sanguíneos para cada una de las imágenes en todos los volúmenes, los cuales han sido anotados de manera manual por expertos del área. La figura 3.2 muestra un ejemplo del conjunto de datos, así como su correspondiente mapa de segmentación para el ejemplo seleccionado.

3.2. Marco Teórico

3.2.1. Redes Neuronales Convolucionales

Dentro del aprendizaje profundo, las redes neuronales convolucionales han jugado un papel importante, ya que son referente en cuanto a las aplicaciones que ha logrado el estudio de la anatomía humana aplicada a proyectos de aprendizaje automático. Las redes neuronales convolucionales son una clase de redes neuronales profundas, diseñadas para trabajar con



Figura 3.1: Resonador Phillips Achieva 3T [?].

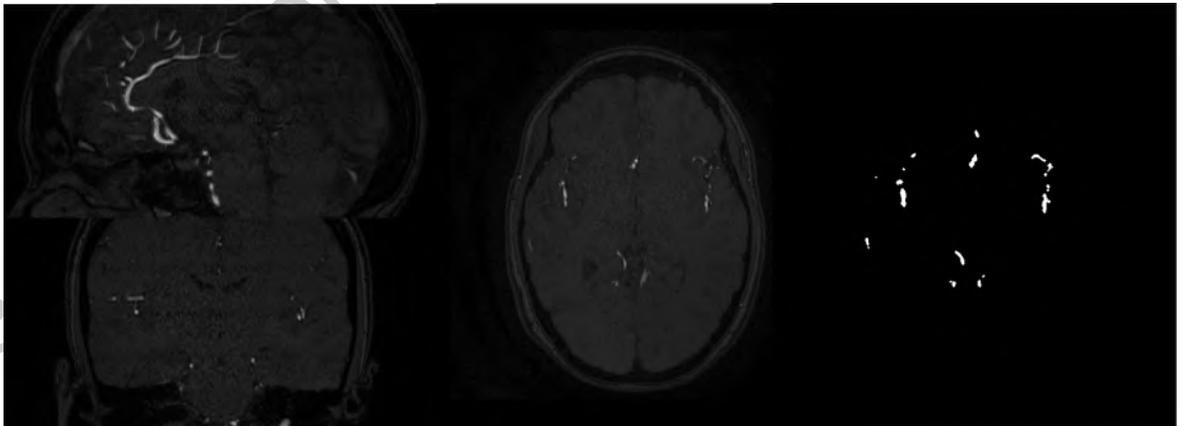


Figura 3.2: Vistas sagital, coronal y axial de un volumen del conjunto de datos junto al mapa de segmentación correspondiente.

datos de entrada en forma de cuadrícula en dos dimensiones. Comúnmente son utilizadas con imágenes, dentro del campo de la visión artificial, aunque son útiles también con otro tipo de datos, como texto y en general cualquier tipo de datos espaciales y/o temporales. Una red neuronal convolucional es simplemente una red neuronal común que en al menos una de sus capas realiza la operación matricial de convolución.

Arquitectura

Comúnmente, una red neuronal convolucional contiene tres tipos de capas; convolución, pooling y ReLU, como en la figura 3.3. Cada una de estas capas acepta datos (imágenes) de tipo volumétrico. Una red neuronal convolucional tiene las siguientes propiedades:

- Recibe a la entrada un volumen con dimensiones $W_1 * H_1 * D_1$. Donde:
 - W_q es el ancho (width) de la imagen.
 - H_q es la altura (height) de la imagen.
 - D_q es la profundidad (depth) de la imagen.
- Recibe 4 hiper parámetros:
 - Número de filtros K .
 - La longitud del filtro F .
 - El stride S .
 - El padding P .
- Produce una salida con dimensiones $W_2 * H_2 * D_2$. Donde:
 - $W_2 = (W_1 - F + 2P) / S + 1$.
 - $H_2 = (H_1 - F + 2P) / S + 1$.
 - $D_2 = K$.

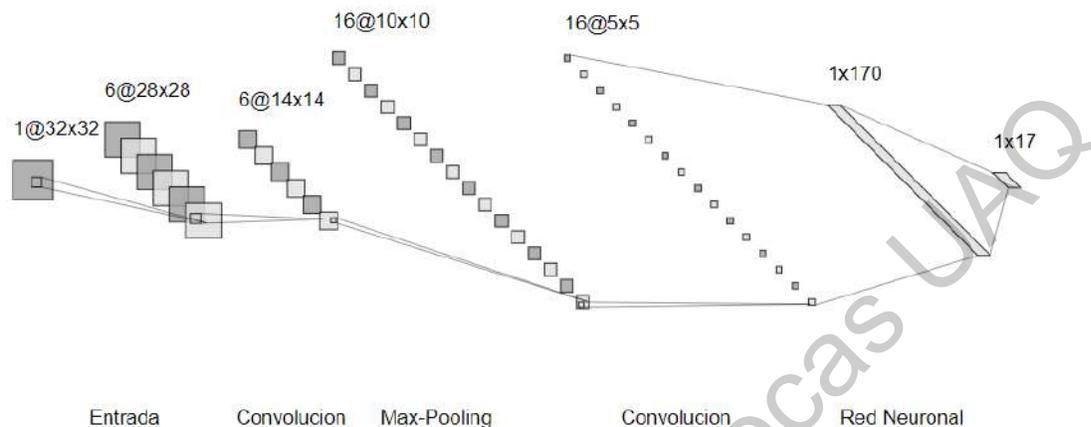


Figura 3.3: Arquitectura general de una red neuronal convolucional.

Convolución

La convolución de dos funciones está dada por la integral del producto de ambas funciones, después de desplazar una de ellas cierta distancia t .

$$s(t) = \int x(a)w(t - a)da \quad (3.1)$$

Siendo esta la operación de convolución, también denotada con un asterisco de la siguiente manera:

$$s(t) = (x * w)(t) \quad (3.2)$$

En términos de una red neuronal, el primer argumento es conocido como la entrada y el segundo como el kernel (figura 3.4). Comúnmente, el filtro de convolución (kernel) es un arreglo pequeño de números. Si la imagen a procesar la compone un solo canal (como lo es una imagen en escala de grises), basta con el filtro en dos dimensiones. Cuando la imagen es a color (3 canales si es RGB), es necesario definir un arreglo para cada canal, es decir, un filtro con 3 dimensiones.

Para realizar la convolución en una imagen, se realiza el producto punto entre el kernel y

$$\begin{vmatrix} 1.0 & 1.0 & 1.0 & 1.0 \\ 1.0 & 1.0 & 1.0 & 1.0 \\ -1.0 & -1.0 & -1.0 & -1.0 \\ -1.0 & -1.0 & -1.0 & -1.0 \end{vmatrix}$$

Figura 3.4: Kernel simple para la detección de líneas horizontales.

una sección de la imagen con dimensiones iguales y posteriormente se suman los resultados. De manera más formal, se considera el kernel como una función y se le conoce como función kernel (K), la cual es de dimensiones $F_q * F_q * D_q$. Como resultado a lo anterior y como se muestra en la figura 3.5 se obtiene un valor V en cierta posición con coordenadas (x,y) sobre una imagen (I), como lo muestra la siguiente operación:

$$V(x,y) = (I * K)(x,y) = \sum_m \sum_n I(x+m, y+n)K(m,n) \quad (3.3)$$

Una capa de una red neuronal convolucional, comúnmente consta de tres etapas. Primero se realizan las operaciones de convolución para producir un grupo de activaciones lineales. Segundo, cada activación lineal es transferida a una función de activación no lineal, como ReLU y, finalmente se utiliza la función de pooling para modificar el resultado de las capas aún más.

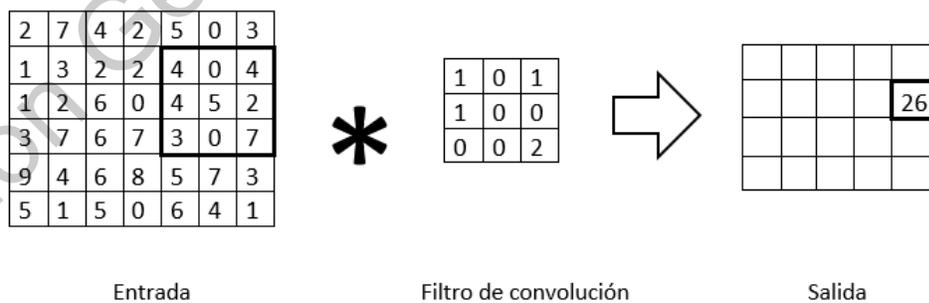


Figura 3.5: Matriz de dos dimensiones (imagen) en convolución con kernel de 3x3.

Preservación del margen en operaciones de convolución (del inglés *padding*)

Como se observa en la figura 3.6, la operación de convolución reduce el tamaño de la salida de la capa (n+1) en comparación con la capa n. Para evitar esta pérdida de información en el borde de la imagen, se utiliza el padding. Este método consiste en agregar $(F_q - 1)/2$ píxeles alrededor de los bordes de la imagen para mantener una correspondencia espacial. El valor de cada uno de estos píxeles es de 0. De esta manera, se agrega la cantidad de píxeles en la imagen que en la operación de convolución se perderían, evitando así que la imagen resultante después de varias capas de convolución sea demasiado pequeña. Cabe mencionar que el padding no afecta al producto punto final ya que, como ya se mencionó, estos píxeles extra contienen valores de 0.

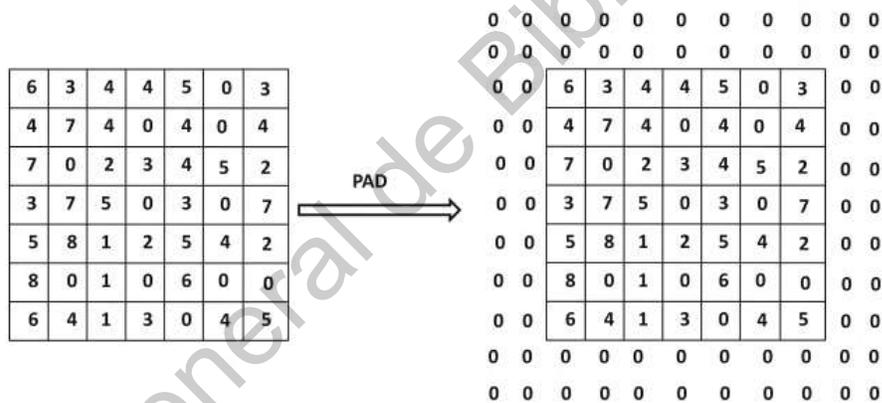


Figura 3.6: Padding para agregar margen a la imagen de entrada y así evitar la pérdida de información al aplicar filtros convolucionales [57].

Desplazamiento del filtro convolucional (del inglés *strides*)

En las funciones previas se muestra el cómo aplicar el kernel en la imagen para todo el espacio que representa. Sin embargo, para reducir la granularidad en una imagen al aplicar los distintos filtros, se introduce el concepto de strides. Esto es, simplemente aplicar el kernel para la convolución saltando una cantidad definida de píxeles. Como ejemplo, se muestra en la figura 3.7 el cómo sería el desplazamiento como es realizado comúnmente sin stride y en

la Ilustración 5, se puede visualizar un desplazamiento de 2 píxeles cada que se aplica el filtro de convolución, lo cual significa que al stride se le atribuye ese valor de mdesplazamiento.

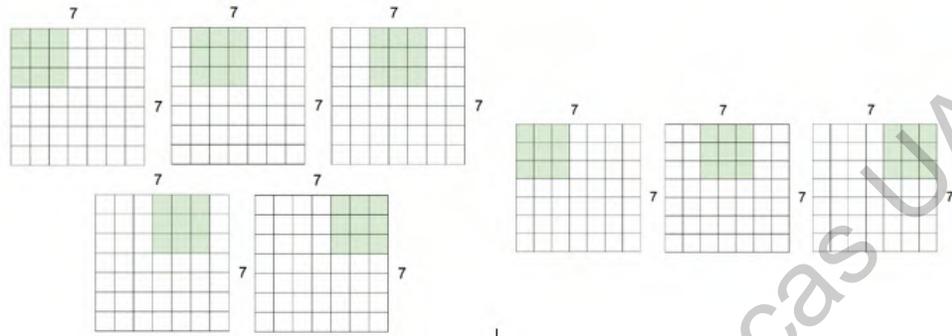


Figura 3.7: Convolución sobre todo el espacio de la imagen (izquierda). Convolución con stride de 2 (derecha).

Capa de reducción (del inglés *pooling*)

La función pooling reemplaza el valor de cierta posición dentro del arreglo que compone a la imagen con una suma estadística de los puntos cercanos. Por ejemplo, la función max pooling regresa el valor máximo entre un grupo de vecinos. El pooling (figura 3.8) ayuda a mantener casi invariante el resultado de la operación a pesar de pequeñas perturbaciones dentro de los datos de entrada a la función, mejorando considerablemente el rendimiento general de la red.

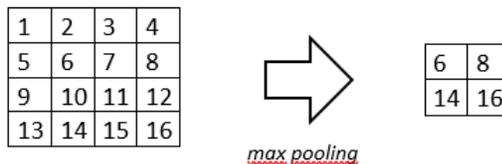


Figura 3.8: Función max pooling con disminución de resolución.

3.2.2. Funciones de Activación en Redes Neuronales

Uno de los elementos más importantes en la selección de hiper parámetros dentro de cualquier tipo de red neuronal es la función de activación que usará cada neurona, tanto para

entrenamiento como para predicción. Dependiendo de la tarea a resolver por la red neuronal, se suele preferir una función de activación por sobre las demás. Es común utilizar un tipo de función para la última capa del modelo mientras que el resto utiliza alguna distinta, esto con el fin de ajustar la salida del mismo al tipo de dato que se desea obtener. Por ejemplo, si a la salida del modelo se espera un valor real, se utiliza una función de activación identidad mientras que, en el caso de una probabilidad dentro de una clasificación binaria, se recomienda el uso de la función sigmoide. En la implementación de modelos profundos o multicapa, es importante el utilizar funciones de activación no lineales, como sigmoide o tangente hiperbólica [57].

Unidad Lineal Rectificada (ReLU)

La función de activación ReLU (Rectified Linear Unit) no es diferente de la que se aplica en una red neuronal convencional. Activa una neurona únicamente si el valor de entrada x se encuentra por encima de cierto umbral. Suponiendo un umbral de cero, cuando el valor de entrada es negativo, la función regresa cero y cuando es mayor a cero, simplemente regresa el mismo valor. Por lo tanto, no afecta a la dimensión de las capas previas, siendo únicamente una operación de activación uno a uno. A manera general, de la siguiente forma y en la figura 3.9, se resume que:

$$f(x) = \begin{cases} x & \text{si } x > 0 \\ 0 & \text{en caso contrario} \end{cases} \quad (3.4)$$

En general, ReLU ofrece las siguientes ventajas:

- Es fácil de calcular ya que no requiere de matemáticas complejas, por lo que al modelo le tomará menos tiempo el entrenamiento.
- No siempre se activa. Debido a que como resultado puede dar 0 con valores de entrada x menores a cero, situación que en muchas ocasiones es deseable.
- Converge de manera más rápida. No se satura con entradas de x grandes.

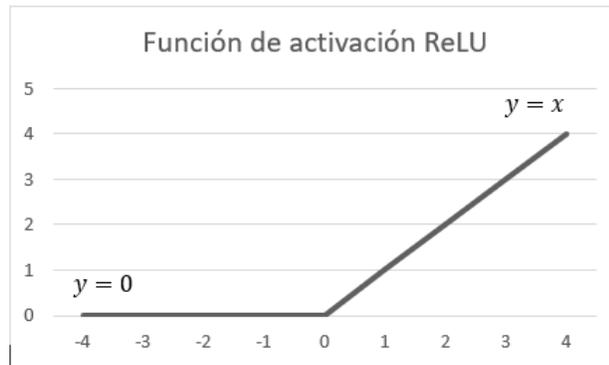


Figura 3.9: Comportamiento de la función de activación ReLU.

Unidad Lineal Rectificada con Fuga (Leaky ReLU)

Leaky ReLU es una pequeña variación a la función ReLU convencional, en vez de tener una salida de cero cuando $x \leq 0$, se obtendrá una pequeña pendiente, como se muestra en la figura 3.10. La salida está dada por:

$$f(x) = \begin{cases} x & \text{si } x > 0 \\ \alpha * x & \text{en caso contrario} \end{cases} \quad (3.5)$$

donde $\alpha \in \mathbb{R}$

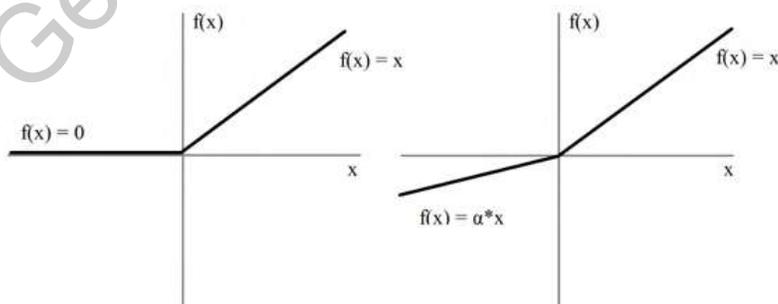


Figura 3.10: Función de activación ReLU (izquierda) vs Leaky ReLU (Derecha).

Función Sigmoid

Previo a la adopción de ReLU como función de activación no lineal, era popular el uso de la función sigmoide (figura 3.11). La desventaja de esta es el limitado rango de valores que puede dar como salida, siendo este de 0 a 1, mientras ReLU puede ir desde 0 hasta infinito. Esto es de relevancia en la implementación del algoritmo de propagación hacia atrás para el entrenamiento de las redes neuronales, ya que el gradiente puede convertirse en cero con relativa facilidad, provocando el efecto de desvanecimiento del gradiente [58]. La función sigmoide está definida por:

$$S(x) = \frac{e^{-x}}{1 + e^{-x}} \quad (3.6)$$

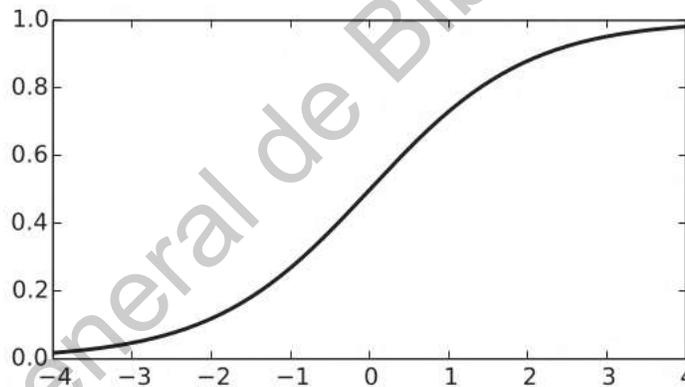


Figura 3.11: Función de activación sigmoide [58].

Función Tangente Hiperbólica (Tanh)

La función de activación tangente hiperbólica (tanh) es relativamente similar a la sigmoide, con la diferencia de que puede devolver como resultado un rango de valores más amplio, tanto positivos como negativos dentro del intervalo $[-1, 1]$, como se muestra en la figura 3.12. El hecho de estar centrada y más alargada que la sigmoide la hace más fácil de entrenar [57]. La función tanh está dada por:

$$g(x) = \tanh(x) \quad (3.7)$$

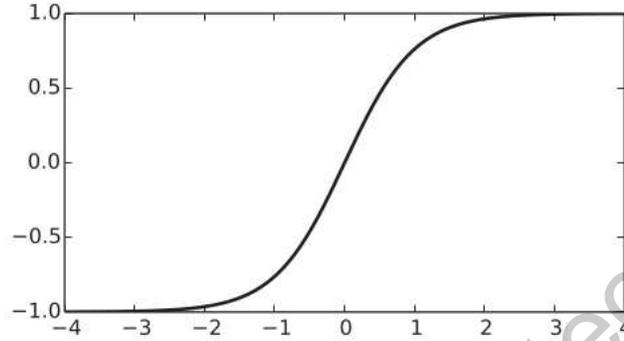


Figura 3.12: Función de activación tanh [58].

3.2.3. Redes Generativas Adversarias

Las redes generativas adversarias (GANs) son arquitecturas compuestas por dos modelos, un generador G y un discriminador D , con parámetros $\theta^{(G)}$ y $\theta^{(D)}$, respectivamente. G aprende un mapeo desde un vector de ruido aleatorio z a un mapa (imagen) de salida x , representado por $x = G(z; \theta^{(G)})$, mientras que la tarea de D es distinguir entre ejemplos del conjunto de datos y ejemplos generados por G , denotado por $D(x; \theta^{(D)})$ [43].

La función del modelo $V(D, G)$ es representado por un juego min-max, donde D es entrenado para maximizar la probabilidad de una etiqueta correcta entre ejemplos reales (del conjunto de entrenamiento) y falsos (generados por G), mientras que G intenta minimizarla generando ejemplos que son indistinguibles de los reales, como se muestra a continuación:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log (1 - D(G(z)))] \quad (3.8)$$

El propósito del modelo es que el discriminador aprenda las propiedades del conjunto de datos de entrenamiento, siendo estos los ejemplos considerados como reales, y los obtenidos por el generador, como falsos. De la clasificación del discriminador, el generador es forzado a generar mejores resultados cada vez más cercanos a ser indistintos del conjunto de

entrenamiento.

La estructura de una red generativa adversaria se muestra en la figura 3.13. Un mapa de ruido se introduce al modelo generador G , mediante el cual se obtiene a la salida un ejemplo bajo las características de la distribución de datos con la que es entrenado. Sin importar que tanto se asemeje a un ejemplo del conjunto de datos, se le asigna la etiqueta de *falso*, con el fin de entrenar al modelo discriminador D con estos ejemplos y, al complementar el entrenamiento con el conjunto de datos real, D sea capaz de extraer características pertenecientes y no pertenecientes a la distribución de datos del conjunto de entrenamiento.

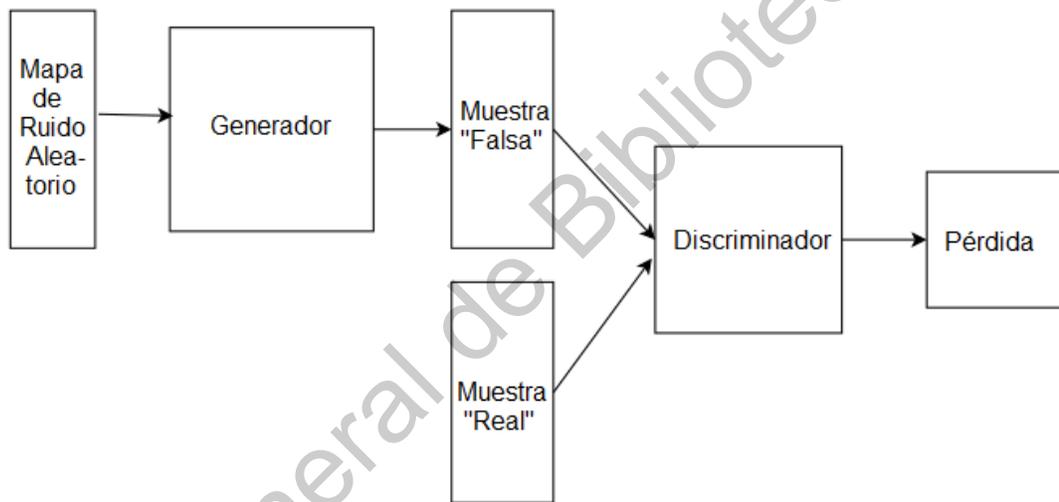


Figura 3.13: Arquitectura básica de una red generativa adversaria [58].

3.2.4. Redes Generativas Adversarias Condicionales

Las redes generativas adversarias condicionales (cGANs) agregan la propiedad sobre las GANs de tener un ejemplo de entrada junto al mapa z como información extra, con el fin de condicionar el modelo a generar ejemplos basados en este ejemplo. Esto es de gran utilidad para una variedad de aplicaciones, como es el caso de la segmentación de imágenes, donde el mapa de salida del modelo es relativamente similar al de entrada y, por lo tanto, una imagen totalmente distinta a la de entrada no es necesaria. La función objetivo para estos modelos es

la siguiente:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data(x)}} [\log D(x|y)] + \mathbb{E}_{z \sim p_z(z)} [\log (1 - D(G(z|y)))] \quad (3.9)$$

En la figura 3.14 se muestra la arquitectura de una red generativa adversaria condicional. Los mapas y, z descritos previamente se alimentan al modelo generador G , el cual realiza una predicción para obtener mapa de salida $G(z|y)$. Los mapas generados mediante la inferencia de G , dados los parámetros $\theta^{(G)}$, son utilizados para el entrenamiento de los parámetros $\theta^{(D)}$ del modelo discriminador, con una etiqueta de salida igual a 0 debido a que pertenecen al conjunto de ejemplos "falsos". Para el caso opuesto en el que la etiqueta de salida es igual a 1, el modelo D es entrenado directamente con el conjunto perteneciente a la distribución "real". La pérdida obtenida en D al realizar este proceso de entrenamiento con ambos conjuntos es de utilidad para actualizar los parámetros $\theta^{(G)}$ y $\theta^{(D)}$, así como se mostró previamente en la ecuación que describe al modelo condicional.

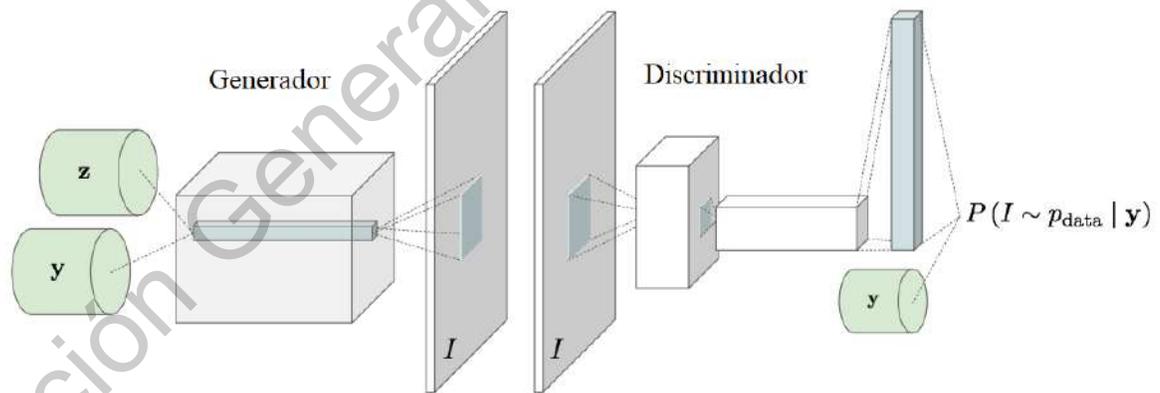


Figura 3.14: Arquitectura de una red generativa adversaria condicional[59].

3.2.5. Entrenamiento de modelos convolucionales

Para que un modelo logre el objetivo para el que es propuesto, es necesario alimentar al mismo con datos para el entrenamiento. Los modelos basados en redes neuronales imple-

mentan el método del descenso del gradiente (fig 3.15), basado en la propagación hacia atrás del error en función de la pérdida con el fin de minimizarla y así ajustarse a la distribución de datos de entrada y su correspondiente etiqueta de salida.

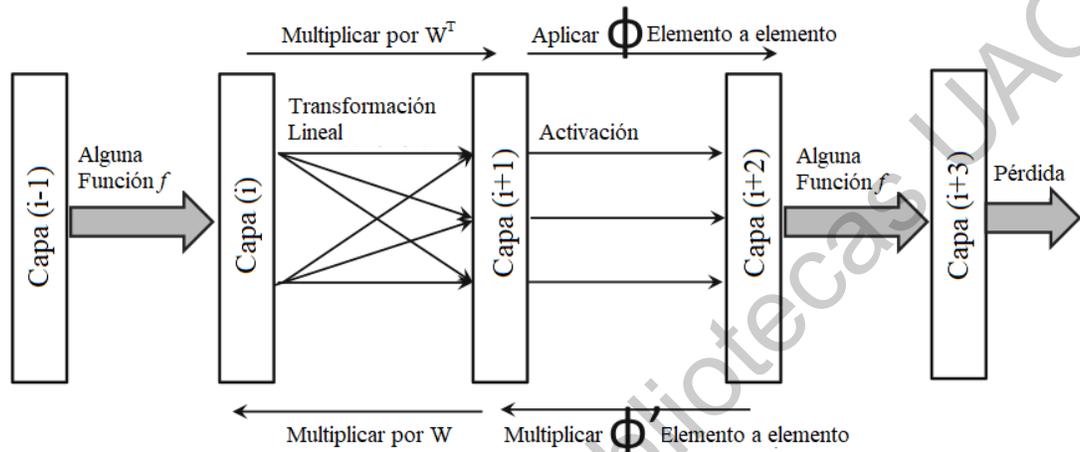


Figura 3.15: Algoritmo de backpropagation para entrenamiento de redes neuronales [57].

El algoritmo de backpropagation se compone de dos fases, la propagación hacia adelante, para el cálculo de la pérdida o error, y hacia atrás para actualizar los parámetros del modelo.

La propagación hacia adelante se asemeja al proceso de inferencia, donde un vector de ejemplos se propaga por cada una de las capas realizando las operaciones dadas por los parámetros del modelo en el momento dado, obteniendo la predicción del modelo bajo la evaluación de la función de pérdida a minimizar.

En la propagación hacia atrás, se calcula el gradiente de la función de pérdida con respecto a los parámetros en cada capa del modelo. Primeramente, es necesario calcular la derivada de la función de pérdida con respecto de la predicción $\frac{\partial \mathcal{L}}{\partial o}$

Para llevar a cabo el entrenamiento del modelo convolucional, se utiliza el algoritmo de Backpropagation, el cual no es muy distinto del que se aplica en redes neuronales convencionales, donde una operación de propagación hacia atrás se realiza multiplicando un vector de gradientes en la capa $(q + 1)$ con la transpuesta de la matriz de pesos entre las capas q y $(q + 1)$ para obtener el vector de gradientes en la capa q . En CNN, las derivadas de la propagación hacia atrás son también asociadas con posiciones espaciales dentro de las capas. Desde la capa $(i + 1)$ hasta la capa i , se multiplica la derivada del error con respecto a la capa

$(i + 1)$ con la transpuesta de la matriz de la propagación hacia adelante entre las capas i e $(i + 1)$, donde la propagación hacia atrás puede verse como una forma de convolución transpuesta. La derivada del error con respecto a una celda en la capa $(i + 1)$ es definida como la derivada parcial de la función de coste con respecto a la variable en esa celda.

- Transferencia de Aprendizaje.

Para el entrenamiento de una CNN, no es común inicializar los parámetros de manera aleatoria, ya que puede ser demasiado costoso debido a los tiempos de procesamiento y el consumo de recursos energéticos que conlleva. Debido a esto, existen redes pre entrenadas con datasets gigantescos, como ImageNet, de las cuales es posible extraer los parámetros obtenidos del entrenamiento y aplicarlos en una red neuronal diferente para alguna aplicación más específica. A esta técnica se le conoce como transfer learning. Por ejemplo, el entrenamiento de una CNN sobre ImageNet puede tardar entre 2 y 3 semanas utilizando varias unidades de procesamiento gráfico (GPUs), por lo cual es común el compartir estos modelos a otros investigadores para realizar el entrenamiento sobre aplicaciones más específicas. Existen dos disyuntivas en cuanto al uso de la técnica:

- Ajustar un modelo ya existente.

Consiste en emplear una CNN ya entrenada con una gran cantidad de datos y modificar la última capa, conocida como la capa de “clasificación”, con algo más específico del dataset que se desea emplear para la detección de características más específicas. En algunos casos, el algoritmo de entrenamiento de backpropagation sólo es empleado en las últimas capas con el fin de reducir el tiempo de cómputo, en comparación con el entrenamiento sobre toda la CNN. Esto se debe a que algunas características de la visión artificial son demasiado generales y no requiere un re entrenamiento para las mismas.

- Utilizar un modelo de CNN como extractor de características.

En esta variante, se remueve la última capa de la CNN pre entrenada y se considera el resto como el “extractor de características” para el dataset más pequeño. Esto es importante

ya que CNNs grandes pueden diferenciar entre miles de clases, lo cual es innecesario para aplicaciones más específicas.

El uso de transfer learning es común bajo las siguientes situaciones: El dataset que se posee es:

- Demasiado pequeño.
- Comparte algunas características visuales con el dataset base.

Estándar para imágenes médicas DICOM

DICOM (por sus siglas en inglés Digital Imaging and Communications in Medicine) es un estándar internacional para transmitir, guardar, procesar y publicar información sobre imágenes médicas. Esto hace que la información médica sea interoperable, contenga una integración con una variedad de dispositivos de distintos proveedores y se mantenga actualizado constantemente. Es de carácter libre, por lo que no tiene costo alguno la utilización y descarga de este formato. DICOM agrupa la información en datasets. Un mismo archivo contiene, por ejemplo, imágenes de rayos X y la información del paciente, por lo que la información de este jamás puede ser separada de las imágenes por error.

APIs

Para la implementación de la red neuronal convolucional, actualmente existen herramientas (o APIs) que pueden convertir de este proceso algo mucho más intuitivo que realizando la programación de manera tradicional. Además, ofrecen múltiples mejoras en rendimiento y aceleración tanto por hardware como por software para simplificar aún más el proceso de implementación de algún sistema de aprendizaje profundo. A continuación, se describen algunas APIs a utilizar en el entorno de este proyecto.

- Tensorflow

TensorFlow es una plataforma de código abierto orientada al aprendizaje automático. Contiene un ecosistema de múltiples herramientas, librerías y una comunidad de investigadores que

de manera constante apoyan al crecimiento de la plataforma, conllevándola hacia un entorno totalmente dedicado al aprendizaje automático y las redes neuronales profundas. El sistema es lo suficientemente general como para ser aplicado en una gran cantidad de campos dentro de la investigación. TensorFlow provee versiones estables para Python y C++.

- Keras

Keras es un API de alto nivel para el desarrollo e implementación de redes neuronales. Está escrito en Python y es capaz de correr a la par con algunos otros APIs de aprendizaje automático, como TensorFlow, CNTK o Theano. Su objetivo principal es el desarrollo con enfoque a una rápida experimentación de los modelos a implementar. Sus principales características son:

- Provee un entorno de fácil prototipado con una usabilidad amigable con el usuario.
- Soporta redes neuronales convolucionales, recurrentes y una combinación de ambas.
- Se puede ejecutar tanto en CPU como en GPU.
- Software para visualización de imágenes DICOM.

Es de suma importancia el contar con diferentes herramientas para la visualización de las imágenes médicas a procesar. Es necesario saber exactamente con qué tipo de datos se trabaja y comprender cómo están compuestos y lo que significan, por lo que para la visualización de imágenes DICOM, los modelos generados en 3 dimensiones, será necesario utilizar una combinación de las siguientes herramientas. Además, para realizar un etiquetado de manera manual para los datos de entrenamiento, es factible la implementación de la segmentación en alguna de estas plataformas, ya que es común que se utilicen para ello y están optimizadas para lo mismo.

3.3. Metodología

En la figura 3.16 se muestra la metodología propuesta a seguir. En esta se destacan las principales fases a implementar dentro del desarrollo del proyecto.

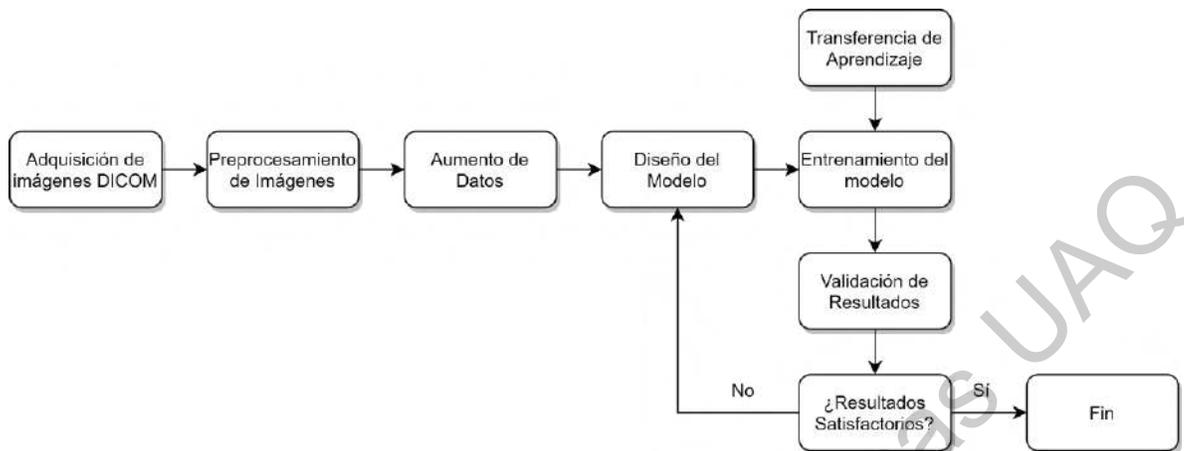


Figura 3.16: Propuesta metodológica del presente proyecto.

- Adquisición de Imágenes DICOM (del inglés Digital Imaging and Communication On Medicine).

Las imágenes serán obtenidas con el fin de obtener los datos necesarios para el entrenamiento de la CNN, las cuales serán imágenes de resonancia magnética del área cerebral. Estas imágenes serán adquiridas en el formato estándar para la visualización de imágenes médicas DICOM.

- Selección del ROI (del inglés Region Of Interest).

Una vez adquiridas las imágenes, debido al gran tamaño que las caracteriza, es necesario reducir sus dimensiones. Estas imágenes en 2 dimensiones al ser agrupadas representan un modelo tridimensional, siendo necesarias cientos de las mismas para modelar el cráneo y su interior de manera apropiada, lo cual es la principal característica por la cual es tan demandante la capacidad computacional para el entrenamiento. Al seleccionar una ROI, se reduce considerablemente el número de datos a procesar, considerando todo el volumen cerebral y reduciendo el área sobre la cual se trabajará

- Reducción de dimensiones.

Como ya se ha mencionado anteriormente, para reducir el tiempo de entrenamiento de la CNN es necesaria la reducción de los datos de entrada a la red. Las imágenes deberán ser

re escaladas con el fin de reducir el número de neuronas de entradas. Este proceso se realiza mediante funciones específicas para extraer características sin perder información importante de las imágenes, como pueden ser las operaciones de convolución

- Etapa de codificación.

Este proceso consiste en reorganizar los datos que se tienen previo al procesamiento, siendo estos las imágenes médicas sin haber aplicado ningún filtro, para posteriormente ajustarlas para que satisfagan a la arquitectura de la red neuronal en la entrada. Para este caso, el reacomodo de las imágenes es una transformación de su forma matricial a una forma vectorial, donde cada pixel corresponderá a una neurona de entrada de la red neuronal convolucional. Aunque este paso es sencillo e incluso trivial, es importante realizar el encoding apropiado a los datos para evitar pérdida de información y, por ende, un resultado no correcto.

- Entrenamiento de la CNN.

El proceso de entrenamiento es el más demandante de todo el proyecto. Esto es debido al tamaño de los datos ya que, siendo tridimensionales, requieren de una mayor capacidad de computación del equipo sobre el que se implementa para reducir los tiempos de ejecución y desarrollar prototipos de manera más rápida y cómoda. Para el entrenamiento, como ya ha sido mencionado, se implementará el algoritmo de backpropagation para la CNN. Para esto no se partirá de los parámetros de la red asignados de manera aleatoria. Se usará la técnica de transfer learning partiendo de una red semi o completamente entrenada para alguna tarea afín o alguna más general, y a partir de estos parámetros que serán implementados en la CNN propuesta, al algoritmo le tomará un menor tiempo completar el entrenamiento y por lo tanto los tiempos de evaluación y ajuste entre cada prototipo será considerablemente menor que partiendo de unos parámetros aleatorios.

- Validación.

Una vez se obtenga un modelo de la CNN funcional, entrenado a partir del conjunto de datos de entrenamiento, este será sometido a pruebas de validación con el restante del dataset

adquirido, siendo esto conocido como el conjunto de datos de validación. A partir de ello, se obtendrán ciertos resultados que al compararse con datos etiquetados por un humano se mostrará que tan eficiente es el algoritmo propuesto. En caso de que el resultado no sea el esperado, será necesario realizar los ajustes necesarios al prototipo, regresando una vez más al ajuste en las dimensiones de los datos. Este ciclo de retroalimentación asegurará que en cada iteración el resultado obtenido se acerque cada vez más a los resultados esperados, donde el objetivo es realizar la segmentación de las imágenes médicas al nivel que lo haría un humano o incluso llegando a ser mejor, tomándole al algoritmo un menor tiempo.

Una vez implementado el prototipo, se llevará a cabo una evaluación para obtener métricas del desempeño del algoritmo de clasificación. Los posibles parámetros a medir son los siguientes:

- Precisión. Es el número de elementos identificados correctamente como positivos del total de elementos identificados como positivos:

$$\frac{TP}{TP + FP} \quad (3.10)$$

- Índice de similitud entre imágenes de Dice:

$$\frac{2TP}{2TP + FP + FN} \quad (3.11)$$

- Sensitividad.

$$\frac{TP}{TP + FN} \quad (3.12)$$

- Sensibilidad. Es el número de elementos identificados correctamente como positivos del total de positivos verdaderos:

$$\frac{TP}{TP + FN} \quad (3.13)$$

Donde:

TP = Verdaderos positivos

FP = Falsos positivos

FN = Falsos negativos

TN = Verdaderos negativos

3.4. Implementación

Preprocesamiento del conjunto de datos

La ecualización de histograma adaptativo limitada por contraste (CLAHE) ha producido buenos resultados en el procesamiento de imágenes para el resaltado de características de interés [34, 46, 47]. Este método se formula en la división de imágenes en varias regiones no solapadas entre sí de el mismo o casi el mismo tamaño.

El objetivo de esto es resaltar áreas de interés en la imagen que se encuentran opacadas o en un rango de intensidad de píxeles muy bajo para hacerlas más notorias y que sean más fáciles de identificar por el algoritmo de segmentación.

El cálculo del histograma para cada región es relativamente simple. Para cada píxel en escala de grises en la imagen, se contabiliza su intensidad. El conjunto de esto es conocido como el histograma de esa región. Esta función es una aproximación a la función de densidad en escala de grises. La ecualización por histograma es obtenida usando un estimado de la función de distribución acumulativa (CDF).

Si el número de píxeles en escala de grises en cada región son respectivamente M y N , y si $h_{i,j}(n)$ para $n = 0, 1, \dots, N - 1$, es el histograma de la región (i, j) , entonces una estimación de la CDF en una imagen en escala de grises es:

$$f_{i,j} = \frac{(N-1)}{M} \sum_{k=0}^n h_{i,j}(k); \quad n = 1, 2, 3, \dots, N-1 \quad (3.14)$$

El pseudocódigo para el método de CLAHE se muestra en la figura 3.17 y los resultados de la implementación del mismo en la figura 3.18

Para el caso del conjunto de datos de MRA de la UNAM, las imágenes muestran ruido

```

Excess = 0
For n = 0,1,2,...,N-1
  If h(n) > β, Then
    Excess ← Excess + h(n) - β
    h(n) ← β
  End If
End For
m = Excess/N
For n = 0,1,2,...,N-1
  If h(n) < β - m, Then
    h(n) ← h(n) + m
    Excess ← Excess - m
  Else If h(n) < β, Then
    Excess ← Excess - β + h(n)
    h(n) ← β
  End If
End For
While Excess > 0
  For n = 0,1,2,...,N-1
    If Excess > 0, Then
      If h(n) < β, Then
        h(n) ← h(n) + 1
        Excess ← Excess - 1
      End If
    End If
  End For
End While

```

Figura 3.17: Algoritmo de ecualización de histograma adaptativo limitada por contraste (CLAHE) [48].

en forma de aura alrededor del cráneo de los pacientes, por lo que un preprocesamiento simple puede reducir este ruido. El procedimiento implementado consiste en la operación morfológica de cerradura, la cual consiste en una operación de dilatación en la imagen, con un tamaño de kernel = 9, y posteriormente se erosiona con el mismo tamaño de kernel, para obtener una máscara binaria, correspondiente al área de interés y evitando así el ruido externo. La máscara es multiplicada por la imagen original para obtener de vuelta la imagen de resonancia magnética, sin el ruido en los bordes. Esto se aplica para cada una de las

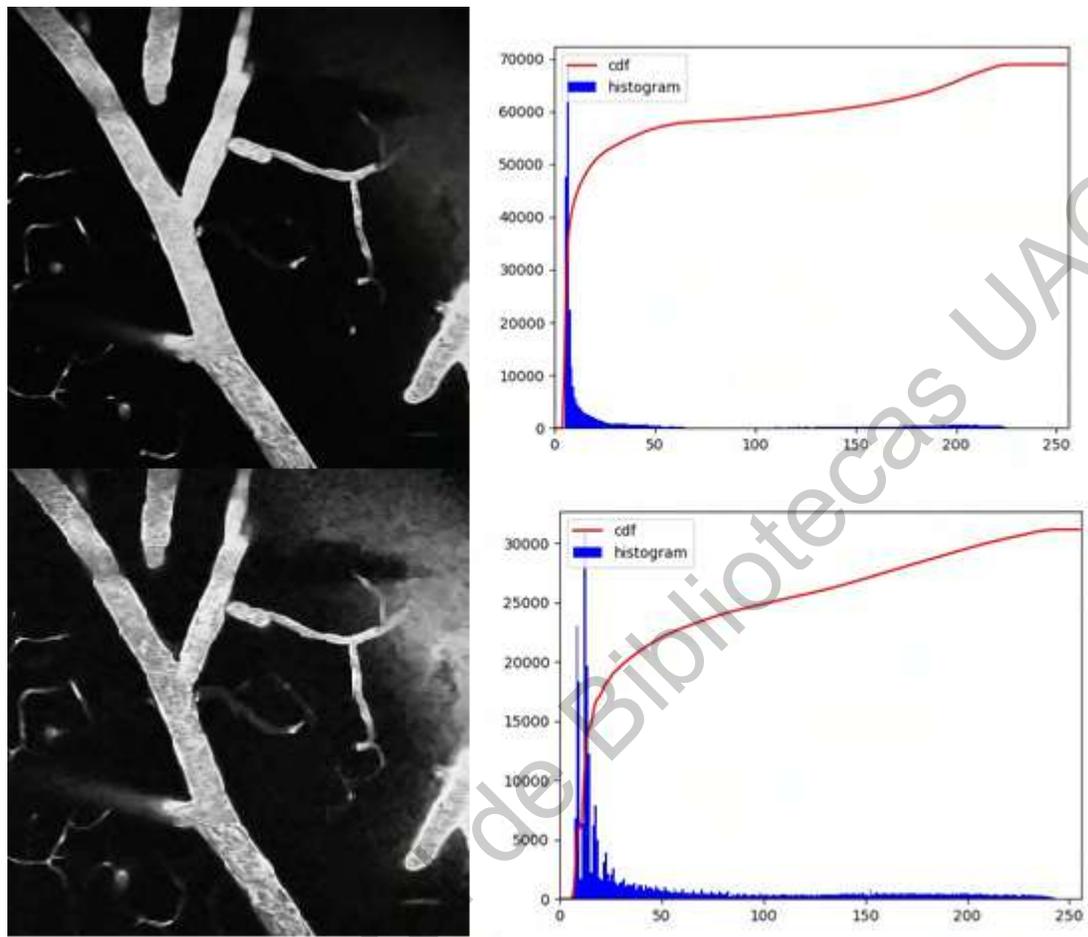


Figura 3.18: Ecualización por histograma implementada en imágenes de microscopía [37]

imágenes contenidas en cada volumen del conjunto de datos y la vista tridimensional en ambos casos, previo y después al procesamiento, se muestran en las figuras 3.19 y 3.20.

Aumento de Datos

Debido a la limitante en cuanto a cantidad de datos en imágenes médicas, se propone el realizar aumento de datos (data augmentation) mediante la librería Keras, la cual permite realizar ciertas transformaciones a las imágenes y así conseguir un mayor volumen de las mismas, sin recurrir a generación de imágenes sintéticas y asegurando que son válidas y pertenecen a la misma distribución de los datos. Las transformaciones aplicadas al conjunto de datos son de acercamiento y reflejo. En la figura figura 3.21 se muestra la transformación

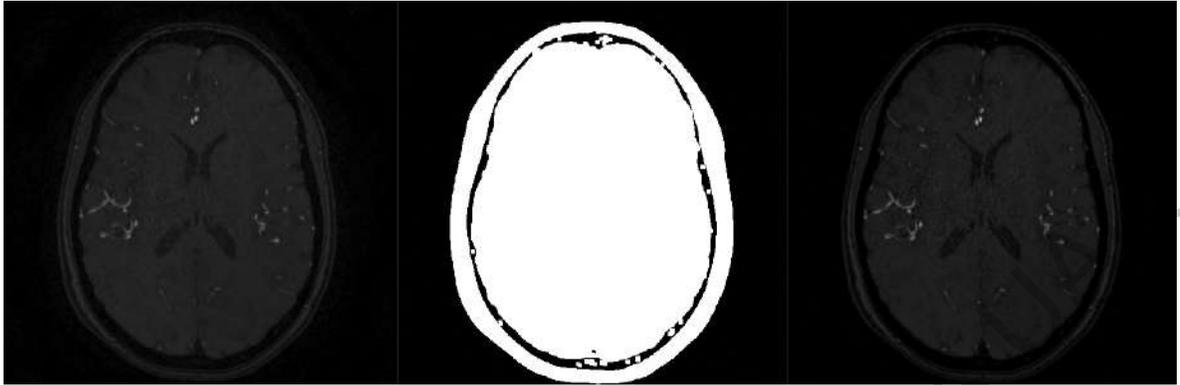


Figura 3.19: Visualización del preprocesamiento realizado a imágenes de TOF-MRA. Muestra del conjunto de datos (izquierda), máscara obtenida a partir de la operación de cerradura (centro), resultado de la multiplicación de la imagen de muestra y la máscara (derecha).

por acercamiento, técnica mediante la cual se puede obtener cualquier número de imágenes, en este caso se generan 3 a partir de una.

En la figura figura 3.22 se aplica la transformación de espejo a las imágenes de la izquierda, obteniendo imágenes totalmente distintas para el entrenamiento de una CNN junto con su etiqueta. De esta manera se puede duplicar el tamaño de cualquier conjunto de datos.

3.4.1. Arquitecturas de redes neuronales convolucionales

Para realizar una comparativa entre los resultados y las condiciones de entrenamiento de distintas arquitecturas de redes neuronales convolucionales, se proponen las siguientes.

Unet

Esta arquitectura (figura 3.23), consiste en una sección de encoding, que captura el contexto y extrae las características de la imagen de entrada. Posteriormente, una sección de decoding reconstruye la imagen a partir de las características extraídas, obteniendo así la imagen segmentada. Esta arquitectura ha demostrado obtener un mejor desempeño al ser combinada con el aumento de datos, por lo que es posible obtener buenos resultados con pocas imágenes para entrenamiento.

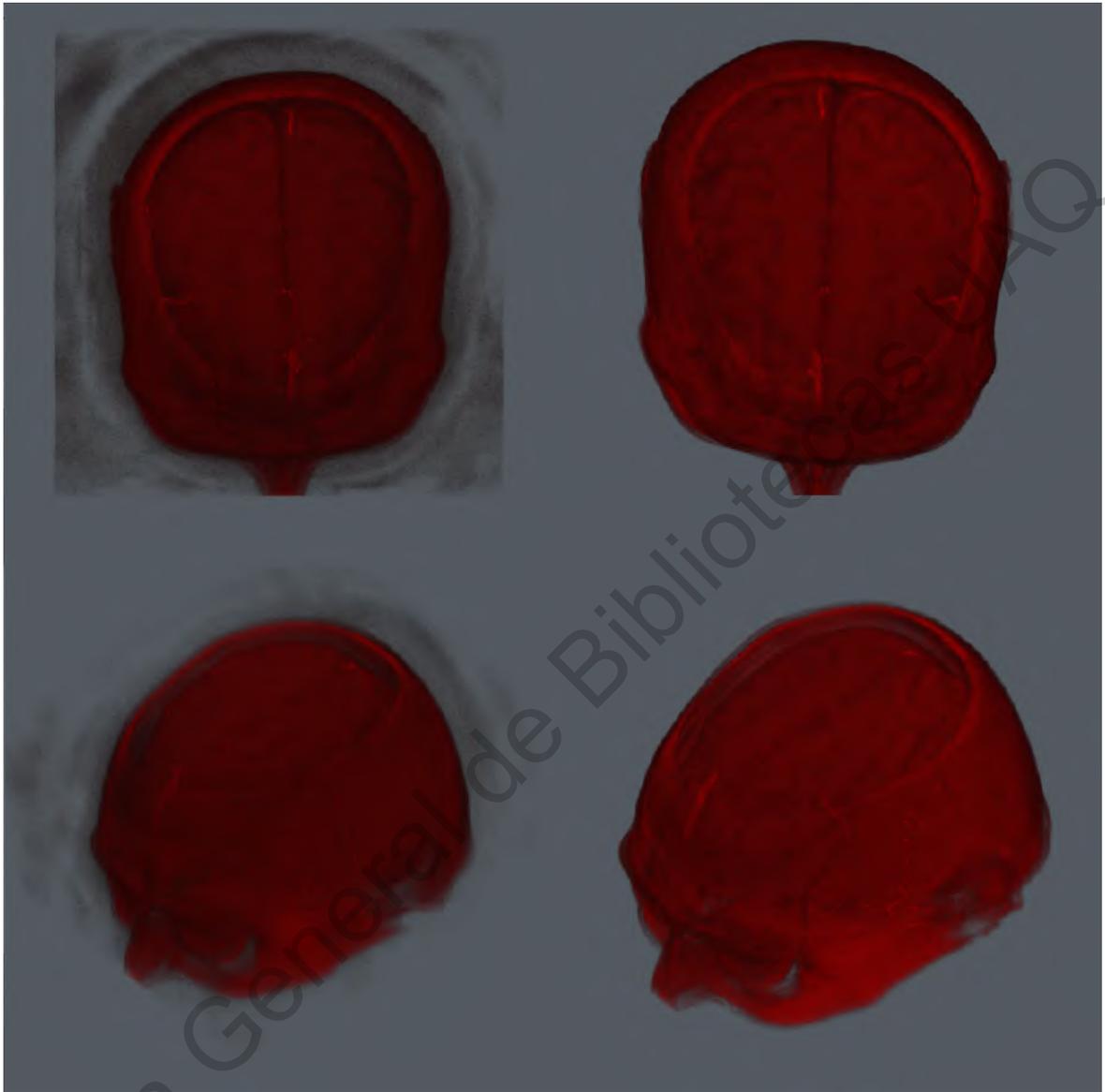


Figura 3.20: Comparación entre el volúmen de TOF-MRA previo y después de la reducción de ruido. Volúmen original (superior e inferior izquierda), resultado de la aplicación de la máscara obtenida para cada imagen del volumen (superior e inferior derecha).

ResUnet

A partir de la arquitectura de la Unet, una alternativa es propuesta mediante la implementación de bloques residuales, combinando las ventajas de estos con el potencial de la Unet, donde los bloques residuales facilitan el entrenamiento y las conexiones saltadas en ellos provocan una mejor propagación de la información semántica de la imagen a capas más pro-

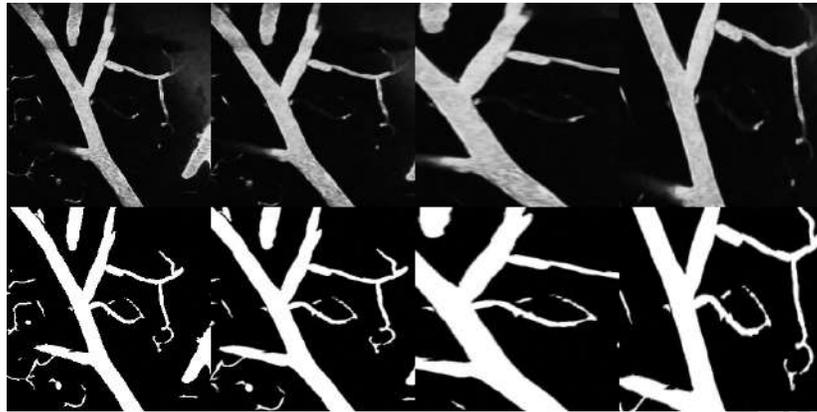


Figura 3.21: Aumento de datos al aplicar acercamiento y transformaciones elásticas en regiones aleatorias de la imagen original



Figura 3.22: Aumento de datos al aplicar un filtro espejo a cada imagen

fundas. La arquitectura ResUnet se muestra en la figura 3.24.

Unet residual doble

La concatenación de una Unet y una Unet residual (figura 3.25) conlleva a mejores resultados en la segmentación de imágenes, principalmente en cuanto precisión como se muestra en [34], combinando los beneficios de estas arquitecturas y, al obtener un modelo más profundo, una mayor cantidad de parámetros dentro del modelo son capaces de extraer carac-

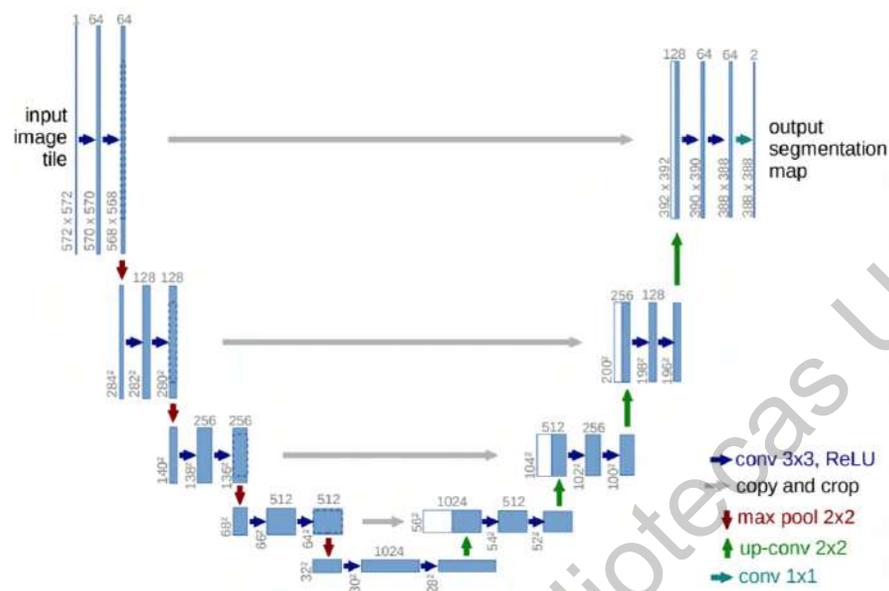


Figura 3.23: Arquitectura CNN de la Unet [31]

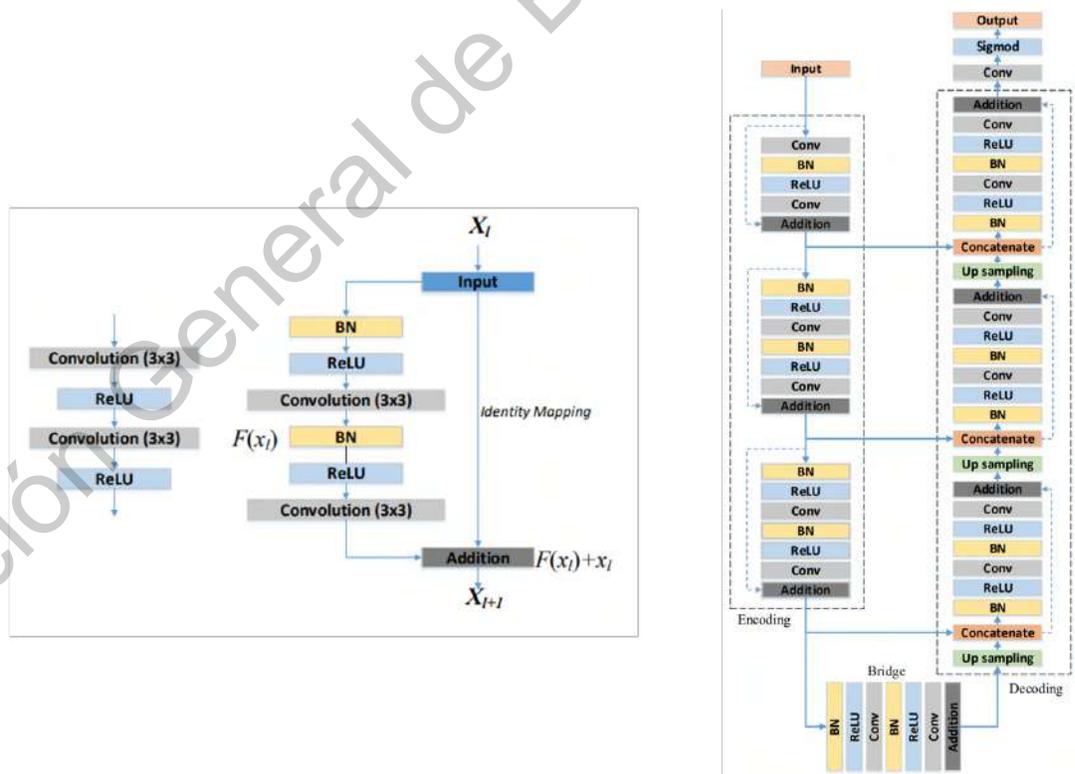


Figura 3.24: Arquitectura basada en Unet con bloques residuales [39].

terísticas relevantes del conjunto de datos, incluso cuando este cuenta con pocas muestras, en conjunto con las técnicas de regularización y aumento de datos.

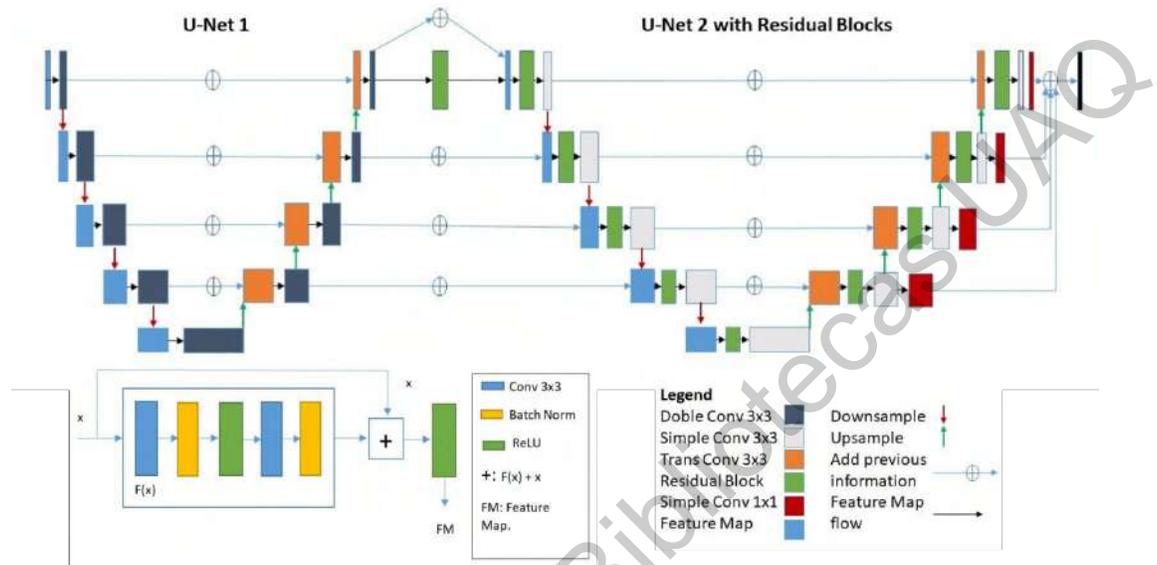


Figura 3.25: Arquitectura con dos CNN concatenadas [34].

El propósito de trabajar con las redes convolucionales antes mencionadas, es realizar una selección de la mejor en cuanto a los experimentos realizados, para realizar la función de generador en el modelo propuesto a continuación. En el caso del conjunto de datos de microscopía [37], los resultados obtenidos se muestran en la tabla 3.1, con los cuales se define la arquitectura de redes concatenadas como el más sobresaliente en este caso.

Tabla 3.1: Entrenamiento de los modelos convolucionales sobre el conjunto de datos de microscopía.

Modelo	Precisión Entrenamiento	Precisión Validación	Pérdida Entrenamiento	Pérdida Validación
U-Net	95.53	93.08	0.0639	0.1421
ResU-Net	95.27	93.72	0.0736	0.1417
U-Net - ResU-Net	95.63	95.54	0.0634	0.0784

3.4.2. Modelo propuesto UUr-cGAN

El modelo propuesto es basado en la arquitectura Pix2Pix [50], la cual es basada, a su vez en dos modelos. Primeramente, un modelo generador G con dos etapas: un modelo encoder-decoder convolucional basados en la U-Net [31], concatenado con otro similar pero con bloques residuales, basado en [55]. Posteriormente, un modelo discriminador D basado en la sección de encoder de la primera parte del generador. El modelo UUr-cGAN se resume a continuación en la figura 3.26.

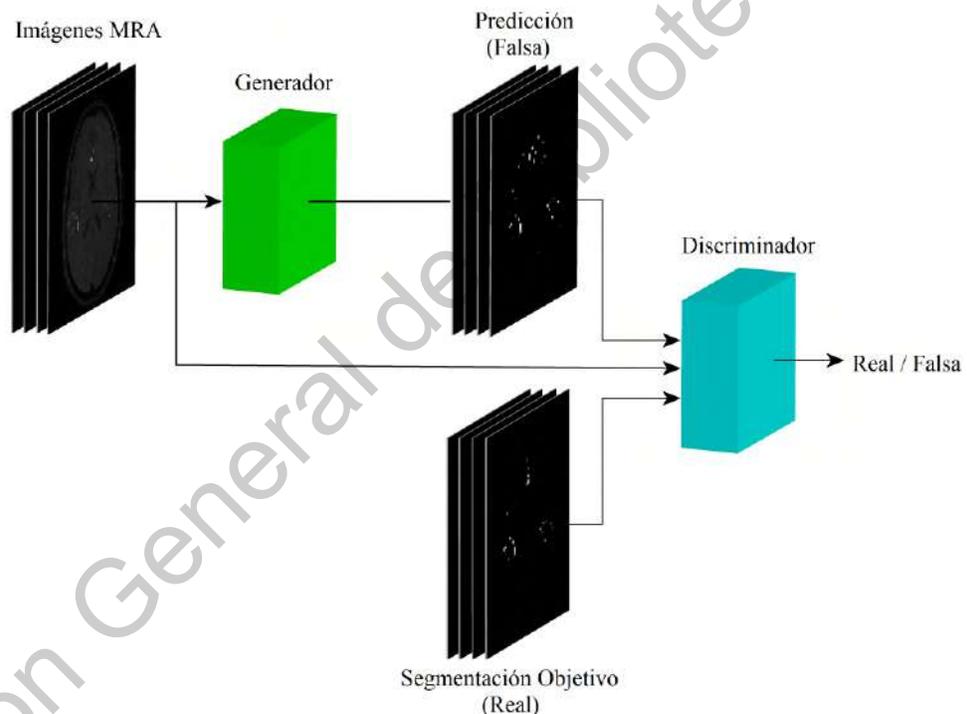


Figura 3.26: Modelo UUr-cGAN propuesto

UUr-cGAN toma como entrada una imagen perteneciente al volumen 3D MRA, con una resolución de $256 \times 256 \times 1$ píxeles, con una profundidad de color de 8 bits, junto con el mapa de ruido aleatorio z en forma de dropout [56], lo cual entra a G . Por otra parte, a la salida de D se obtiene un mapa de características de 16×16 , como se sugiere en [50].

Para el encoder de la primera sección de G , cuatro filtros convolucionales de tamaño 3×3

son implementados con *same padding*, con el primero de estos con un stride de 2, seguido por normalización de instancia y la función de activación LeakyReLU. El decoder realiza cuatro convoluciones transpuestas para incrementar el tamaño del mapa de características, seguido por otro filtro de convolución de tamaño 3×3 , normalización por instancia y LeakyReLU como activación.

La segunda sección del generador es muy similar a la anterior, con la diferencia de reemplazar las convoluciones siguientes a la reducción de dimensiones del mapa de características por un bloque residual. Esto afecta en ambas partes, tanto en el encoder como en el decoder de esta sección. Al final de esta, se implementa *tanh* como función de activación, siendo esta la salida del modelo generador G .

Los bloques residuales están basados en [55, 34], y se representan en la ecuación siguiente y en la figura 3.27, donde cada uno de estos toma como entrada un mapa de características n -dimensional, denotado como FM . Primero una convolución de tamaño 3×3 con *same padding*, seguido por normalización de instancia, LeakyReLU como función de activación, otra convolución de 3×3 , y finalmente normalización de instancia. La salida de estas operaciones se suman con el mapa de características inicial en forma de conexión saltada y se implementa una última función de activación LeakyReLU.

$$FM(x) = F(x) + x \quad (3.15)$$

El modelo del discriminador D está basado en el encoder de la primer etapa del generador G , con filtros de tamaño 4×4 , manteniendo las mismas operaciones y agregando al final una convolución de un solo filtro, con *same padding* para generar un mapa de salida de 16×16 . Se usa como función de activación LeakyReLU con excepción de la última capa, donde se implementa una función Sigmoide.

La arquitectura final del modelo se muestra en la figura 3.28, para ambos modelos G y D , dentro de la cGAN. G es seguido por D para obtener un arreglo binario a la salida, el cual contiene la clasificación de si un ejemplo podría pertenecer a la distribución del conjunto de datos real o de la generada por G . A partir de esta clasificación, G es entrenado para

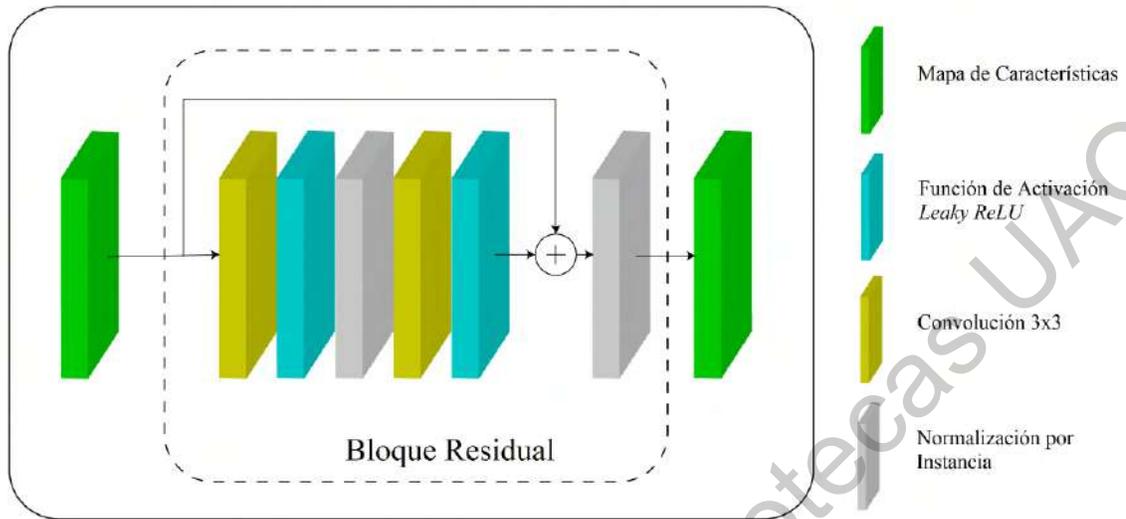


Figura 3.27: Arquitectura del bloque residual dentro de la segunda etapa del generador del modelo cGAN propuesto. Una conexión saltada es usada para preservar información del mapa de características de la entrada del bloque a las capas más profundas, combinándolas con una operación de suma.

minimizar la función de pérdida asociada a este, mientras que los pesos de D permanecen como no entrenables al momento.

El proceso de entrenamiento para D depende de dos conjuntos de ejemplos, reales y falsos, sin que este afecte en lo absoluto al modelo G .

Dos funciones de pérdida son utilizadas, una para cada modelo dentro de la arquitectura. La función de pérdida L1 es utilizada para G junto a la pérdida de la cGAN, mientras que D hace uso de la función de entropía cruzada binaria como pérdida.

El procedimiento general de entrenamiento del modelo cGAN está basado en [43], donde un paso del descenso del gradiente es ejecutado para el modelo discriminador, seguido de uno para el modelo generador. La función objetivo se muestra en la ecuación 3.16. Una combinación de dos funciones de pérdida es implementada como en [50] para G , donde la pérdida L1 como se describe en la ecuación siguiente.

$$g = \arg \min_G \max_D \mathcal{L}_{cGAN}(D, G) + \lambda \mathcal{L}_{L1}(G) \quad (3.16)$$

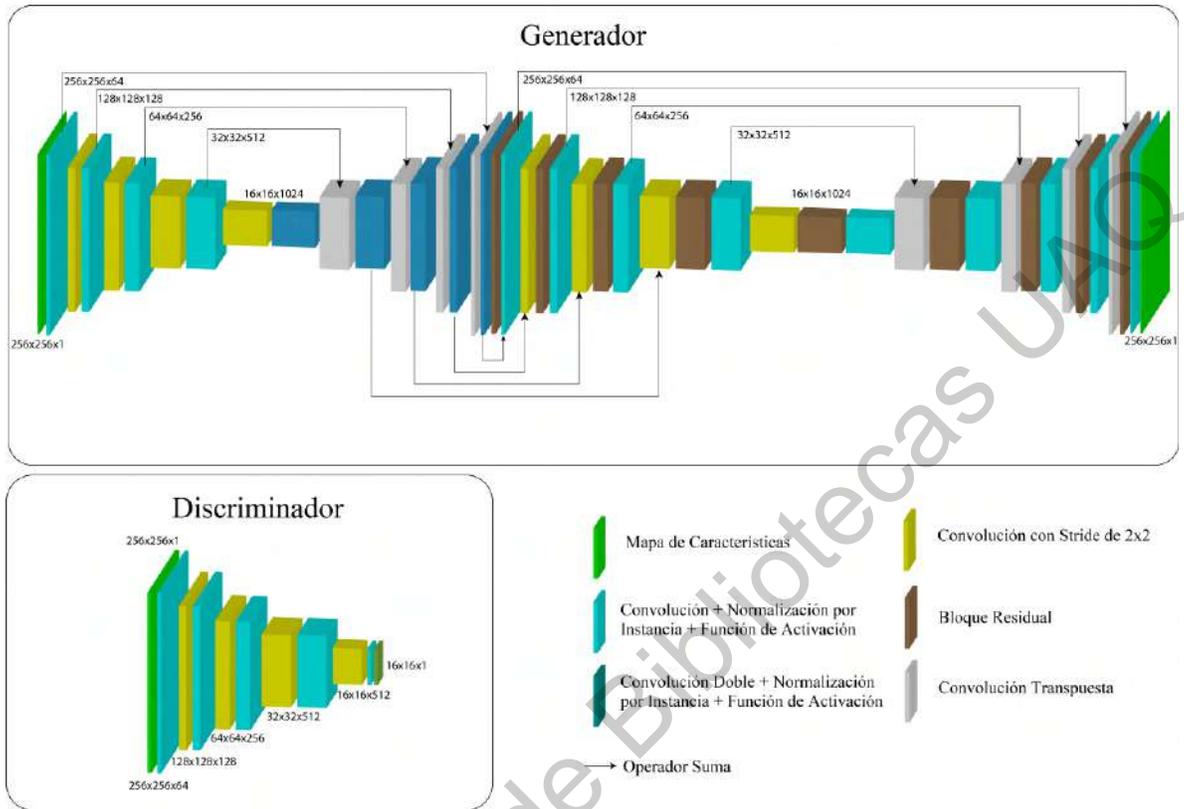


Figura 3.28: Modelos generador y discriminador basados en [31, 34, 50, 55] para la cGAN propuesta. El generador se inspira en la concatenación de dos modelos tipo U-Net, donde el segundo implementa bloques residuales, con el objetivo de propagar información a las capas más profundas del modelo, preservando detalles más finos entre los mapas de características.

Donde:

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z} \|y - G(x,z)\| \quad (3.17)$$

Usualmente, se agrega un mapa de ruido aleatorio z junto con el ejemplo de entrada del modelo. Como reemplazo a esto, se implementa dropout [56] en la sección del cuello de botella de G . De manera similar, D usa dropout en la capa final. En ambos casos la probabilidad asignada es de 0.1.

El modelo generador es entrenado con el optimizador Adam [53], tasa de aprendizaje inicial $\alpha = 0,0002$ con un decaimiento exponencial de 0,9, $\beta_1 = 0,5$ y $\beta_2 = 0,9$, mientras que el discriminador usa RMSProp [54], bajo las mismas condiciones para la tasa de aprendizaje y

con un momentum = 0,9. Adicionalmente, el paso del entrenamiento para D es dividido entre 2 con respecto a G . Normalización de instancia se implementa por las mejoras en estabilidad y desempeño general en el proceso de entrenamiento, como se sugiere en [52] para modelos generativos.

Dirección General de Bibliotecas UAQ

Discusión y Resultados

4.1. Diseño del Experimento

El conjunto de datos es dividido en subconjuntos, cada uno utilizado en distintas iteraciones como conjunto de validación para la evaluación del experimento por el método de k-Fold. Por lo tanto, dado el conjunto de datos a disposición para el experimento, se implementa un 4-fold, por lo que en cada iteración, 3 volúmenes son utilizados para el entrenamiento del modelo, mientras que uno es utilizado como validación.

La selección de hiper parámetros para el modelo propuesto han sido seleccionados a partir del modelo Pix2Pix [50], realizando ajustes en función de las necesidades y de los resultados de los experimentos, entrenando la arquitectura propuesta con variaciones de los hiper parámetros hasta acercarse al mejor desempeño posible en base a las métricas de evaluación.

Como se muestra en la ecuación 3.16, el parámetro α debe ser establecido, el cual representa el balance entre la pérdida de la cGAN y la pérdida L1, por lo que distintos valores fueron probados para el modelo UUr-cGAN, partiendo de lo seleccionado en [50]. Después de varias iteraciones, se eligió una proporción de 75 a 1 en favor de la pérdida L1, ya que bajo esta configuración, los resultados de la segmentación fueron más cercanos a la segmentación objetivo.

El valor de probabilidad para dropout [56] es definido a partir del entrenamiento de múlti-

ples modelos, variando el mismo. Un valor menor al seleccionado ($probabilidad = 0,8$) provoca un bajo ajuste del modelo al conjunto de datos de entrenamiento, mientras que un valor menor conlleva a que el modelo sea más propenso al sobre ajuste, cuyo comportamiento fue observado en los experimentos.

Para el caso del tamaño del batch, se establece el valor máximo posible en función de las capacidades de memoria de la GPU, variando este dependiendo del modelo a entrenar.

En la tabla 4.1, se muestra la selección de hiper parámetros para los cuatro modelos implementados, la U-Net [31], U-Net doble [34], Pix2Pix [50] y el método propuesto UUr-cGAN. Cabe destacar que en todos los casos, el optimizador Adam tiene como parámetros $\beta_1 = 0,9$ y $\beta_2 = 0,999$, y RMSProp con momentum = 0,9.

Tabla 4.1: Selección de hiper parámetros para los modelos evaluados, donde G representa el modelo generador y D el modelo discriminador.

Híper parámetro	U-Net	U-Net Doble	Pix2Pix	UUr-cGAN
Optimizador	Adam	Adam	G: Adam D: Adam	G: Adam D: RMSProp
Función de Pérdida	Logarítmica	Logarítmica	G:MAE+cGAN D:Logarítmica	G:MAE+cGAN D:Logarítmica
Dropout	0.7	0.7	0.8	0.8
Tasa de Aprendizaje Inicial	0.001	0.001	0.0002	0.0002
Tamaño del Batch	50	50	30	10

4.2. Hardware Utilizado

El entrenamiento para los modelos evaluados fue realizado en el servicio de computación en la nube de Microsoft Azure. La instancia de máquina virtual utilizada cuenta con un procesador Intel Xeon E5-2690, una GPU Nvidia Tesla K80 con 16 GB de VRAM, 56 GB de

memoria RAM y Ubuntu 18.04. Las librerías utilizadas son Keras en Tensorflow 2.3.0 para definir y entrenar los modelos convolucionales, OpenCV y Numpy para el manejo de imágenes y el preprocesamiento de las mismas, Nibabel para la lectura y escritura de archivos sobre el estándar DICOM, y Medpy para obtener las funciones de las métricas de evaluación, todo sobre Python 3.6.

4.3. Resultados

La tabla 4.2 muestra los resultados para el experimento mencionado, así como en la figura 4.1 se visualizan algunas secciones de un volumen de MRA (ver también Anexo A), así como su comparación de segmentación objetivo y predicción realizada por el modelo entrenado, mientras que en la figura 4.2 se visualizan los resultados obtenidos con el método propuesto con cada uno de los volúmenes a disposición, donde para cada uno de estos, la predicción mostrada es el volumen de validación dentro del experimento de k-Fold correspondiente.

Tabla 4.2: Métricas evaluadas sobre los modelos entrenados en cada iteración del experimento k-Fold.

k	Índice Dice	Precisión	Sensitividad	Especificidad
1	88.25	87.89	88.64	99.96
2	85.74	90.64	80.22	99.97
3	87.42	91.41	83.87	99.97
4	87.52	88.17	87.46	99.96
Promedio	87.23	89.52	85.04	99.96

En la tabla 4.3, se muestran las distintas métricas evaluadas para los modelos implementados. La U-Net, las U-Net concatenadas (U-Net y U-Net residual), el modelo Pix2Pix y el propuesto, siguen la misma metodología del experimento 4-Fold. Los modelos CNN fueron entrenados con una selección similar de hiperparámetros, basados en [55], usando dropout

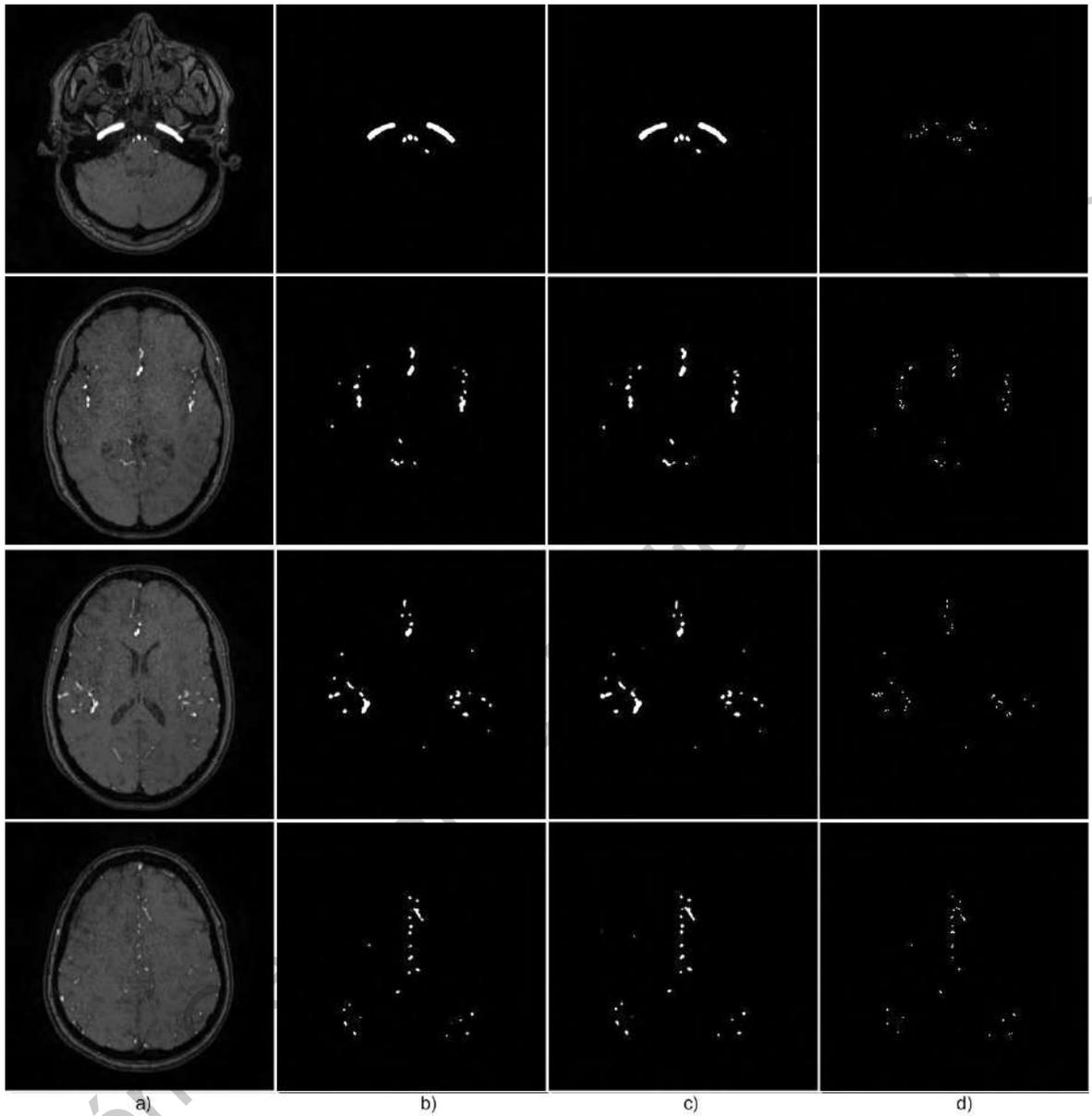


Figura 4.1: Ejemplo de la segmentación de vasos sanguíneos cerebrales. a) Imágenes MRA, b) Segmentación objetivo, c) Predicción del modelo propuesto, and d) Error de segmentación (diferencia entre la segmentación objetivo y la predicción).

para disminuir el sobre ajuste, dado el tamaño del conjunto de entrenamiento, con una probabilidad de 0.2 entre cada capa, así como una detención del entrenamiento (*early stopping*) de 10 épocas. De manera similar, el modelo Pix2Pix hace uso de los hiperparámetros propuestos por los autores en [50].

Para todos los métodos, se implementa una rutina de decaimiento exponencial de la tasa de aprendizaje, con un ratio de decaimiento de 0.96, así como un tamaño de batch de 10. El modelado 3D de un volumen de validación se muestra en la figura 4.3, así como la predicción de segmentación obtenida con cada modelo entrenado. Por otra parte en la figura 4.4, se muestra la diferencia entre la segmentación objetivo y la predicción para cada uno de los modelos evaluados, sobre del mismo volumen de validación, con el fin de visualizar, a lo largo de todo el volumen, las regiones con un mayor error en las predicciones.

Tabla 4.3: Resultado del experimento k-Fold para los modelos evaluados.

Método	Índice Dice	Precisión	Sensitividad	Especificidad
U-Net [31]	83.71	79.78	93.84	99.94
U-Nets concatenadas [31, 55, 34]	86.13	87.95	86.33	99.97
Pix2Pix [50]	80.92	83.62	78.38	99.96
UUr-cGAN (Modelo Propuesto)	87.23	89.52	85.04	99.96

Es importante mencionar que una comparación directa no puede ser realizada entre los métodos debido a la varianza entre los conjuntos de datos usados para cada uno. De cualquier forma, una referencia es útil, como se muestra en la tabla 4.4.

4.4. Discusión

En nuestros experimentos, la U-Net [31] tiende a sobre ajustarse relativamente fácil, dado el reducido número de ejemplos en el conjunto de entrenamiento. Aún con la implementación de técnicas de regularización, se obtiene cierto nivel de ruido alrededor de las estructuras cerebrales a destacar, mientras que otras áreas con intensidad de pixel similar a los vasos sanguíneos son clasificadas de manera errónea como tales. Las U-Net concatenadas generan volúmenes segmentados con menos ruido de manera general, pero con un nivel más alto de falsos positivos, lo cual se confirma con una sensibilidad inferior a la U-Net. De cualquier

Tabla 4.4: Índice Dice y tamaño del conjunto de datos para distintos métodos del estado del arte para segmentación de vasos sanguíneos cerebrales, así como del modelo propuesto.

Método	Índice Dice	Volúmenes de MRA utilizados
Chen et al. [30]	73.71	10
Phellan et al. [29]	77.40	5
Tetteh et al. [38]	86.68	40
Kandil et al. [40]	84.37	30
Zhao et al. [41]	85.03	30
Livne et al. [32]	92.10	66
Modelo propuesto	87.23	4

forma, se obtiene una precisión más alta y una similitud mayor con la segmentación objetivo, lo que se traduce en un índice Dice más alto.

El modelo Pix2Pix [50] atiende el problema del sobre ajuste, obteniendo una mejor generalización con el mismo conjunto de datos disponible para entrenamiento. Por lo tanto, se implementa este modelo, que conlleva a una tasa más alta de clasificación correcta en pequeños vasos sanguíneos, dentro de la arquitectura cGAN como modelo generador, en lugar del basado en la U-net como en [50]. Se demuestra entonces que al utilizar un modelo basado en la concatenación de dos U-Nets como generador en una cGAN, es posible preservar las ventajas de ambas arquitecturas, reduciendo el impacto del posible sobre ajuste y mejorando el índice Dice para un problema con clases desbalanceadas. Adicionalmente, si se compara la segmentación obtenida en la figura 4.3, es fácil visualizar que el modelo propuesto logra una mejor definición, así como un grado menor de ruido con respecto al resto de modelos evaluados.

En la tabla 4.4, se muestra que el modelo UUr-cGAN propuesto alcanza un índice Dice más alto que los métodos del estado del arte, con excepción del propuesto por Livne et al.

[32], el cual es basado en una única U-Net. Es relevante mencionar de igual forma que ellos, en su experimento, utilizan 66 volúmenes, mientras que en el presente se utilizan únicamente 4.

La tabla 4.2 muestra los resultados de las métricas evaluadas, donde es notorio que todos los modelos, independientemente del resto de métricas, obtienen una alta especificidad. Esto se debe a que el conjunto de datos cuenta con clases desbalanceadas, es decir, de las dos posibles clases, "fondo" y "vaso sanguíneo", la mayoría de los píxeles corresponden al fondo, por lo que los modelos encuentran con mayor facilidad características correspondientes a esta clase, el fondo, dado a que representa una mayor cantidad de información en todas las imágenes. Dado esto, la especificidad nos muestra el qué tan bien el modelo clasifica los píxeles para la clase "fondo", mientras que la sensibilidad hace lo mismo con la clase opuesta, la clase "vaso sanguíneo". Por esto, el clasificar píxeles correspondientes al fondo no es tan complejo para cualquiera de los modelos evaluados como el caso de la clase de interés, aunque de igual manera es importante monitorear la especificidad, con el fin de conocer si hay una mala clasificación de píxeles correspondientes al fondo, con predicción de vaso sanguíneo.

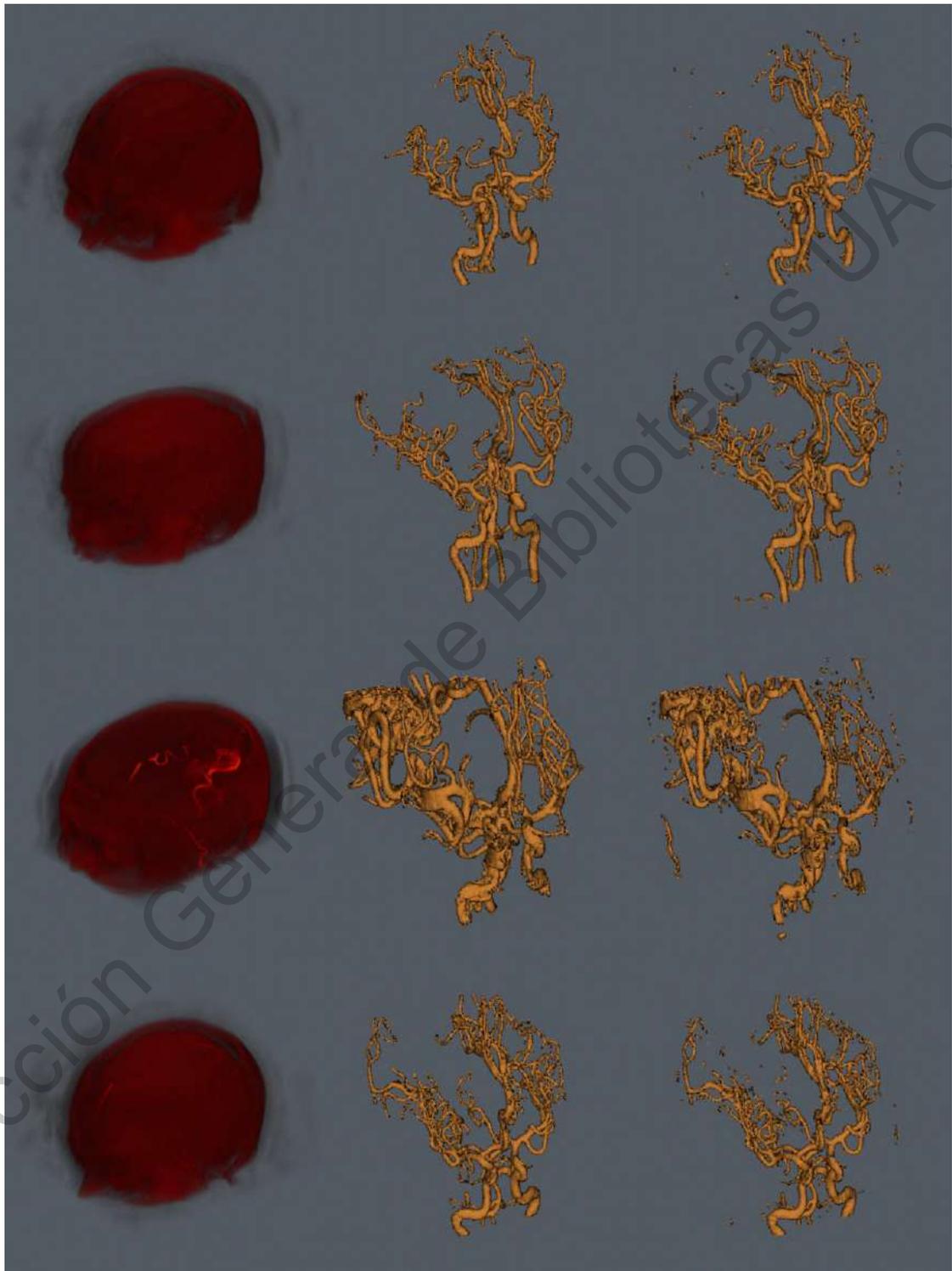


Figura 4.2: Comparación de los volúmenes de MRA segmentados con el método propuesto. Volúmenes MRA (izquierda). Segmentación objetivo (centro). Predicción (derecha).



Figura 4.3: Mapas de segmentación obtenidos con cada modelo evaluado, descritos en tabla 4.3. La visualización 3D es generada con el software Aliza Medical Imaging ©, Bonn, Germany, and DICOM Viewer para la segmentación objetivo y predicciones. a) Segmentación objetivo, b) U-Net [31], c) Pix2Pix [50], d) U-Nets concatenadas [31, 55, 34], y e) cGAN propuesta.



Figura 4.4: Mapas de error obtenidos de la diferencia entre el volumen de segmentación objetivo y la predicción de cada método evaluado, donde las regiones oscuras representan píxeles mal clasificados (entre menos mejor). a) U-Net, b) U-Net doble, c) Pix2Pix, d) Método propuesto.

Conclusión

Los conjuntos de datos para la segmentación de vasos sanguíneos en imágenes cerebrales anotados manualmente son difíciles de crear, dada la tarea de realizar dicho proceso para cada imagen individual para cada volumen a etiquetar, lo que se traduce en un gran consumo de tiempo para el experto que la realiza. Así mismo, el proceso debe ser repetido para cada paciente que deba ser atendido, siendo esta una tarea repetitiva que retrasa el proceso de diagnóstico por parte del especialista, donde dependiendo del caso, puede resultar en complicaciones para el paciente que puede afectar su estado de salud, debido al tiempo de espera para poder realizar dicho procedimiento. Por esto, es de suma importancia contar con modelos capaces de acelerar estas tareas que consumen demasiado tiempo del especialista, para poder realizar una visualización más simple y rápida de los vasos sanguíneos en una muestra de un paciente dado a partir de las angiografías de resonancia magnética.

En este trabajo, se presentó una propuesta de arquitectura basada en redes generativas adversarias, con la propiedad de utilizar la concatenación de dos modelos convolucionales convencionales, como lo son la red U-Net y la U-Net Residual como un único modelo generador. El propósito de esta es llevar a cabo la segmentación de vasos sanguíneos cerebrales en angiografías de resonancia magnética.

Como se mostró en los experimentos realizados, el modelo propuesto logra un índice Dice Score de 87.23 como métrica de similitud de entre imágenes, las segmentadas por expertos

y las predicciones realizadas por el modelo, siendo esto superior a algunos otros trabajos del estado del arte, con la ventaja de requerir de significativamente menos volúmenes de MRA. Aunque no es posible realizar una comparación directa de la propuesta con otros métodos debido a la diferencia entre los conjuntos de datos utilizados, es una buena aproximación dado el hecho de que la mayoría de propuestas recientes han sido basadas en la arquitectura U-Net.

El contar con un número reducido de datos disponibles para entrenamiento de modelos convolucionales ha sido una limitante para muchas otras aplicaciones dentro del área médica, principalmente donde es requerido realizar tareas de segmentación. El trabajo presentado es una propuesta para atender esta problemática, dada la dificultad de acceder a conjuntos de datos con un mayor volumen.

5.1. Trabajo Futuro

Para los métodos de aprendizaje automático supervisado, lo más importante es contar con un buen conjunto de datos, donde lo bueno se define por el tamaño del conjunto y la limpieza o preparación del mismo para la etapa de entrenamiento y validación. Dado esto, la ampliación del conjunto de datos de angiografías de resonancia magnética utilizado en este trabajo puede ser de gran ayuda para mejorar los resultados obtenidos, donde más información ayuda a encontrar y extraer más características relevantes del conjunto de datos para aproximar el modelo de una mejor manera, sin sobre ajustar el modelo ni recurrir (o hacerlo en un menor grado) al aumento artificial de datos.

Así mismo, ante la posible dificultad de aumentar el tamaño del conjunto de datos con el que se cuenta, dado a que debe contar con las mismas características de adquisición, es posible utilizar algún otro de características similares que cuente con su respectiva etiqueta o mapa de segmentación correspondiente. Este conjunto de datos adicional puede ser utilizado como pre entrenamiento para el modelo de aprendizaje automático seleccionado, donde se puede recurrir a datos generados artificialmente como en algunos casos es implementado en

otros trabajos del estado del arte, debido a la facilidad de generar un número arbitrario de muestras, para finalmente hacer un *fine-tuning* del modelo con el conjunto de datos definitivo.

Por otra parte, así como se implementaron métodos para reducir el ruido presente en las imágenes de resonancia magnética, se pueden evaluar algunos otros con el fin de destacar características de interés en una etapa previa al entrenamiento del modelo convolucional, así como eliminar algunas otras como el cráneo y otras regiones de no interés, como ha sido implementado en el estado del arte.

El presente trabajo es de ayuda para sentar las bases de una arquitectura más completa, capaz de realizar la predicción de trayectorias viables a seguir en intervenciones quirúrgicas, dado el punto meta, para obtener el punto de entrada de la herramienta en el área cerebral, buscando realizar el menor daño posible al paciente y evitar complicaciones al especialista al momento de la intervención al dañar estructuras de vasos sanguíneos cerebrales. Así mismo, es viable proponer una arquitectura capaz de realizar la clasificación de diversos padecimientos cerebrovasculares, en caso de presentarse en el paciente, como ya se ha discutido previamente, en el caso de los aneurismas intracraneales, para su detección en función de las imágenes médicas cerebrales obtenidas y la clasificación del tipo al que pertenece y así el especialista pueda actuar en consecuencia en beneficio de la salud del paciente.

Bibliografía

- [1] S. J. Russell and P. Norvig, *Artificial Intelligence: A modern approach*. New Jersey: Prentice Hall, 1995.
- [2] H. R. Myler and A. R. Weeks, *Computer Imaging Recipes in C*. New Jersey: Prentice Hall, 1993.
- [3] J. C. Carr and T. J. Carroll, Eds., *Magnetic resonance angiography. Principles and Applications*, 1st ed. New York: Springer-Verlag, 2012.
- [4] "Cerebral Aneurysms Fact Sheet". National Institute of Neurological Disorders and Stroke. Último acceso: Junio 22, 2021. Disponible en: <https://www.ninds.nih.gov/Disorders/Patient-Caregiver-Education/Fact-Sheets/Cerebral-Aneurysms-Fact-Sheet>
- [5] N. Etminan et al., "The unruptured intracranial aneurysm treatment score," *Neurology*, vol. 85, no. 10, pp. 881–889, 2015.
- [6] D. Backes, G. J. E. Rinkel, K. G. Laban, A. Algra, and M. D. I. Vergouwen, "Patient-and aneurysm-specific risk factors for intracranial aneurysm growth: A systematic review and meta-analysis," *Stroke*, vol. 47, no. 4, pp. 951–957, 2016.
- [7] J. Frösen et al., "Saccular intracranial aneurysm: Pathology and mechanisms," *Acta Neuropathol.*, vol. 123, no. 6, pp. 773–786, 2012.

- [8] M. Forsting and I. Wanke, Eds., *Intracranial Vascular Malformations and Aneurysms*, 2nd ed. Springer-Verlag Berlin Heidelberg, 2008.
- [9] N. K. Yoon, S. McNally, P. Taussky, and M. S. Park, “Imaging of cerebral aneurysms: a clinical perspective,” *Neurovascular Imaging*, vol. 2, no. 1, pp. 1–7, 2016.
- [10] H. C. Echiverri, F. A. Rubino, S. R. Gupta, and M. Gujrati, “Fusiform aneurysm of the vertebrobasilar arterial system,” *Stroke*, vol. 20, no. 12, pp. 1741–1747, 1989.
- [11] “Atherosclerosis”. National Heart, Lung, and Blood Institute. Último acceso: Junio 22, 2021. Disponible en: <https://www.nhlbi.nih.gov/health-topics/atherosclerosis>
- [12] A. Griffin, A. Cutler, and L. F. Gonzalez, “Treatment of a Fusiform Anterior Cerebral Artery Aneurysm by Remodeling the Circle of Willis with Flow Diversion: A Novel Technical Note,” *World Neurosurg.*, vol. 129, pp. 164–169, 2019.
- [13] A. Seitel et al., “Computer-assisted trajectory planning for percutaneous needle insertions,” *Med. Phys.*, vol. 38, no. 6, pp. 3246–3259, 2011.
- [14] M. Rincon-Nigro, N. V. Navkar, N. V. Tsekos, and Z. Deng, “GPU-accelerated interactive visualization and planning of neurosurgical interventions,” *IEEE Comput. Graph. Appl.*, vol. 34, no. 1, pp. 22–31, 2014.
- [15] A. De León-Cuevas, S. Tovar-Arriaga, A. González-Gutiérrez, and M. A. Aceves-Fernández, “Risk map generation for keyhole neurosurgery using fuzzy logic for trajectory evaluation,” *Neurocomputing*, vol. 233, no. December 2015, pp. 81–89, 2017.
- [16] J. O. Neumann et al., “Evaluation of three automatic brain vessel segmentation methods for stereotactical trajectory planning,” *Comput. Methods Programs Biomed.*, vol. 182, p. 105037, 2019.

- [17] I. Villanueva-Naquid et al., “Risk assessment methodology for trajectory planning in keyhole neurosurgery using genetic algorithms,” *Int. J. Med. Robot. Comput. Assist. Surg.*, vol. 16, no. 2, 2020.
- [18] M. Vaillant, C. Davatzikos, R. H. Taylor, R. N. Bryan, A Path-planning Algorithm for Image-guided Neurosurgery, in: *CVRMed-MRCAS’97*, Springer Berlin Heidelberg, 1997, pp. 467–476
- [19] E. A. Rashed, J. Gomez-Tames, and A. Hirata, “End-to-end semantic segmentation of personalized deep brain structures for non-invasive brain stimulation,” *Neural Networks*, vol. 125, pp. 233–244, 2020.
- [20] C. Essert et al., “Statistical study of parameters for deep brain stimulation automatic preoperative planning of electrodes trajectories,” *Int. J. Comput. Assist. Radiol. Surg.*, vol. 10, no. 12, pp. 1973–1983, 2015.
- [21] A. B. Valadka and C. S. Robertson, “Surgery of cerebral trauma and associated critical care,” *Neurosurgery*, vol. 61, no. 1 SUPPL., pp. 203–221, 2007, doi: 10.1227/01.NEU.0000255497.26440.01.
- [22] A. Khan, A. Sohail, U. Zahoor, and A. S. Qureshi, “A Survey of the Recent Architectures of Deep Convolutional Neural Networks,” pp. 1–67, 2019.
- [23] M. Hernández-Pérez et al., “Dynamic Magnetic Resonance Angiography Provides Collateral Circulation and Hemodynamic Information in Acute Ischemic Stroke,” *Stroke*, vol. 47, no. 2, pp. 531–534, 2016.
- [24] H. Arimura, T. Magome, Y. Yamashita, and D. Yamamoto, “Computer-aided diagnosis systems for brain diseases in magnetic resonance images,” *Algorithms*, vol. 2, no. 3, pp. 925–952, 2009.
- [25] C. Essert et al., “Statistical study of parameters for deep brain stimulation automatic preoperative planning of electrodes trajectories,” *Int. J. Comput. Assist. Radiol. Surg.*, vol. 10, no. 12, pp. 1973–1983, 2015.

- [26] M. W. K. Law and A. C. S. Chung, "Segmentation of intracranial vessels and aneurysms in phase contrast magnetic resonance angiography using multirange filters and local variances," *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 845–859, 2013.
- [27] E. A. Rashed, J. Gomez-Tames, and A. Hirata, "End-to-end semantic segmentation of personalized deep brain structures for non-invasive brain stimulation," *Neural Networks*, vol. 125, pp. 233–244, 2020.
- [28] C. Hu et al., "Cerebral vessels segmentation for light-sheet microscopy image using convolutional neural networks," *Med. Imaging 2017 Biomed. Appl. Mol. Struct. Funct. Imaging*, vol. 10137, p. 101370K, 2017.
- [29] R. Phellan, A. Peixinho, A. Falcão, and N. D. Forkert, "Vascular Segmentation in TOF MRA Images of the Brain Using a Deep Convolutional Neural Network," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10552 LNCS, pp. 39–46, 2017.
- [30] L. Chen et al., "3D intracranial artery segmentation using a convolutional autoencoder," *Proc. - 2017 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2017*, vol. 2017-Janua, pp. 714–717, 2017.
- [31] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9351, pp. 234–241, 2015.
- [32] M. Livne et al., "A U-net deep learning framework for high performance vessel segmentation in patients with cerebrovascular disease," *Front. Neurosci.*, vol. 13, no. FEB, pp. 1–13, 2019.
- [33] Z. Zhang, Q. Liu, and Y. Wang, "Road Extraction by Deep Residual U-Net," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 749–753, 2018.

- [34] G. Alfonso Francia, C. Pedraza, M. Aceves, and S. Tovar-Arriaga, "Chaining a U-Net With a Residual U-Net for Retinal Blood Vessels Segmentation," *IEEE Access*, vol. 8, pp. 38493–38500, 2020.
- [35] X. Li, H. Chen, X. Qi, Q. Dou, C. W. Fu, and P. A. Heng, "H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation from CT Volumes," *IEEE Trans. Med. Imaging*, vol. 37, no. 12, pp. 2663–2674, 2018.
- [36] Venkatesh, G. M., et al. A deep residual architecture for skin lesion segmentation. *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*. Springer, Cham, 2018.
- [37] P. Teikari, M. Santos, C. Poon, and K. Hynynen, "Deep Learning Convolutional Networks for Multiphoton Microscopy Vasculature Segmentation," *arXiv preprint arXiv:1606.02382*, 2016.
- [38] G. Tetteh et al., "DeepVesselNet: Vessel Segmentation, Centerline Prediction, and Bifurcation Detection in 3-D Angiographic Volumes," *Frontiers in Neuroscience* 14, 2020.
- [39] X. Zhang, Y. Zhang, and Q. Hu, "Deep learning based vein segmentation from susceptibility-weighted images," *Computing*, vol. 101, no. 6, pp. 637–652, 2019.
- [40] H. Kandil, A. Soliman, F. Taher, A. Mahmoud, A. Elmaghraby, and A. El-Baz, "Using 3-D CNNs and Local Blood Flow Information to Segment Cerebral Vasculature," *2018 IEEE Int. Symp. Signal Process. Inf. Technol. ISSPIT 2018*, pp. 701–705, 2019.
- [41] S. Zhao, Y. U. N. Tian, X. Wang, L. Xie, and L. Sun, "Intracranial Vascular Structure Extraction : A Machine Learning Approach," *IEEE Access*, vol. PP, p. 1, 2019.
- [42] Archivo URM INB UNAM, 2019
- [43] I. J. Goodfellow et al., "Generative adversarial nets," *Adv. Neural Inf. Process. Syst.*, vol. 3, no. January, pp. 2672–2680, 2014.

- [44] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Vol. 1. Massachusetts, USA: MIT Press, 2016.
- [45] M. Mirza and S. Osindero, “Conditional Generative Adversarial Nets,” arXiv preprint arXiv:1411.1784, 2014.
- [46] S. M. Pizer, R. E. Johnston, J. P. Ericksen, B. C. Yankaskas, and K. E. Muller, “Contrast-limited adaptive histogram equalization: Speed and effectiveness,” *Proc. First Conf. Vis. Biomed. Comput.*, pp. 337–345, 1990.
- [47] G. Yadav, S. Maheshwari, and A. Agarwal, “Contrast limited adaptive histogram equalization based enhancement for real time video system,” *Proc. 2014 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2014*, pp. 2392–2397, 2014.
- [48] A. M. Reza, “Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement,” *J. VLSI Signal Process. Syst. Signal Image. Video Technol.*, vol. 38, no. 1, pp. 35–44, 2004.
- [49] M. D. Cirillo, D. Abramian, and A. Eklund, “Vox2Vox: 3D-GAN for Brain Tumour Segmentation,” arXiv preprint arXiv:2003.13653 pp. 1–10, 2020.
- [50] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 5967–5976, 2017.
- [51] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, Jan. 2014.
- [52] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Instance Normalization: The Missing Ingredient for Fast Stylization,” arXiv preprint arXiv:1607.08022, 2016.
- [53] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

- [54] G. Hinton, N. Srivastava and K. Swersky, “Neural networks for machine learning lecture 6a overview of mini-batch gradient descent”. Cited on, 14(8).
- [55] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.
- [56] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” J. Mach. Learn. Res., vol. 15, pp. 1929–1958, Jan. 2014, doi: 10.1016/0370-2693(93)90272-J.
- [57] Charu C. Aggarwal, Neural Networks and Deep Learning. Springer, 2018.
- [58] E. Charniak, Introduction to Deep Learning, vol. 53, no. 9. 2019.
- [59] J. Gauthier, “Conditional generative adversarial nets for convolutional face generation,” Tech. Rep., 2014.

Anexos

A.1. Visualización de volúmenes de MRA.

Las figuras siguientes muestran los volúmenes de MRA del conjunto de datos, segmentación objetivo, predicción y mapas de error correspondientes, obtenidos por el método propuesto, de los vasos sanguíneos cerebrales en las imágenes médicas.

Dichos mapas de error son obtenidos al convertir los archivos DICOM, tanto de la segmentación objetivo como de la predicción, al formato *Numpy*, por medio de la librería *NIBABEL* en Python, con el fin de poder realizar operaciones aritméticas sobre estos. El error en la segmentación es calculado mediante la diferencia absoluta entre la predicción y el objetivo, donde los píxeles desiguales corresponden a un píxel segmentado erróneamente. El resultado de esta operación es almacenado como un arreglo para ser exportado al formato DICOM y visualizarlo al igual que los mapas de segmentación.

Esto es de ayuda para realizar una visualización más sencilla de las regiones con mayor error al realizar la predicción de la segmentación sobre los volúmenes de MRA, notando principalmente el error de segmentación en base al ancho de cada estructura de interés, donde regiones más oscuras del mapa, corresponden a un mayor número de píxeles mal clasificados en el proceso de inferencia del método propuesto.

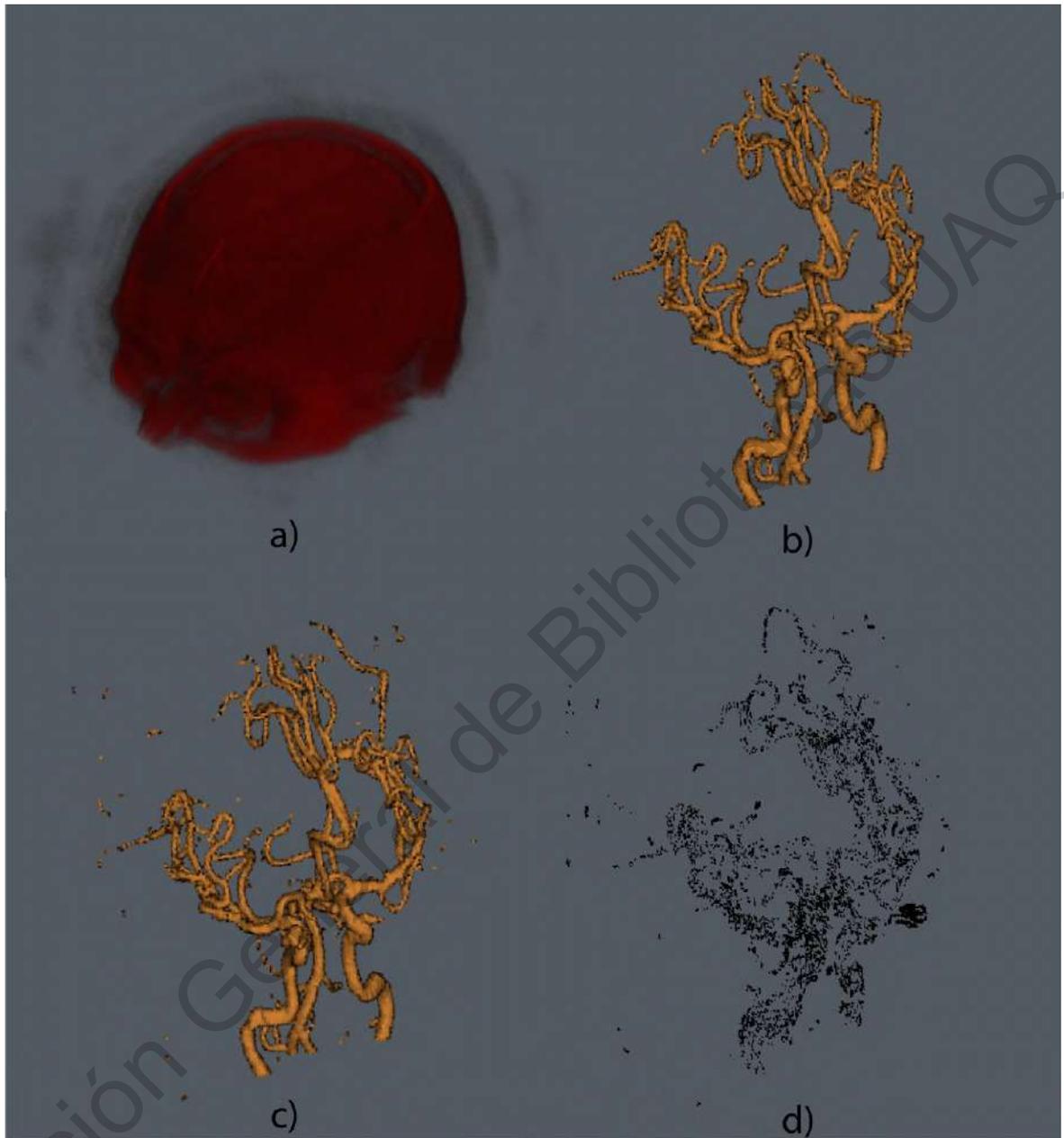


Figura A.1: Visualización del volumen 1 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.

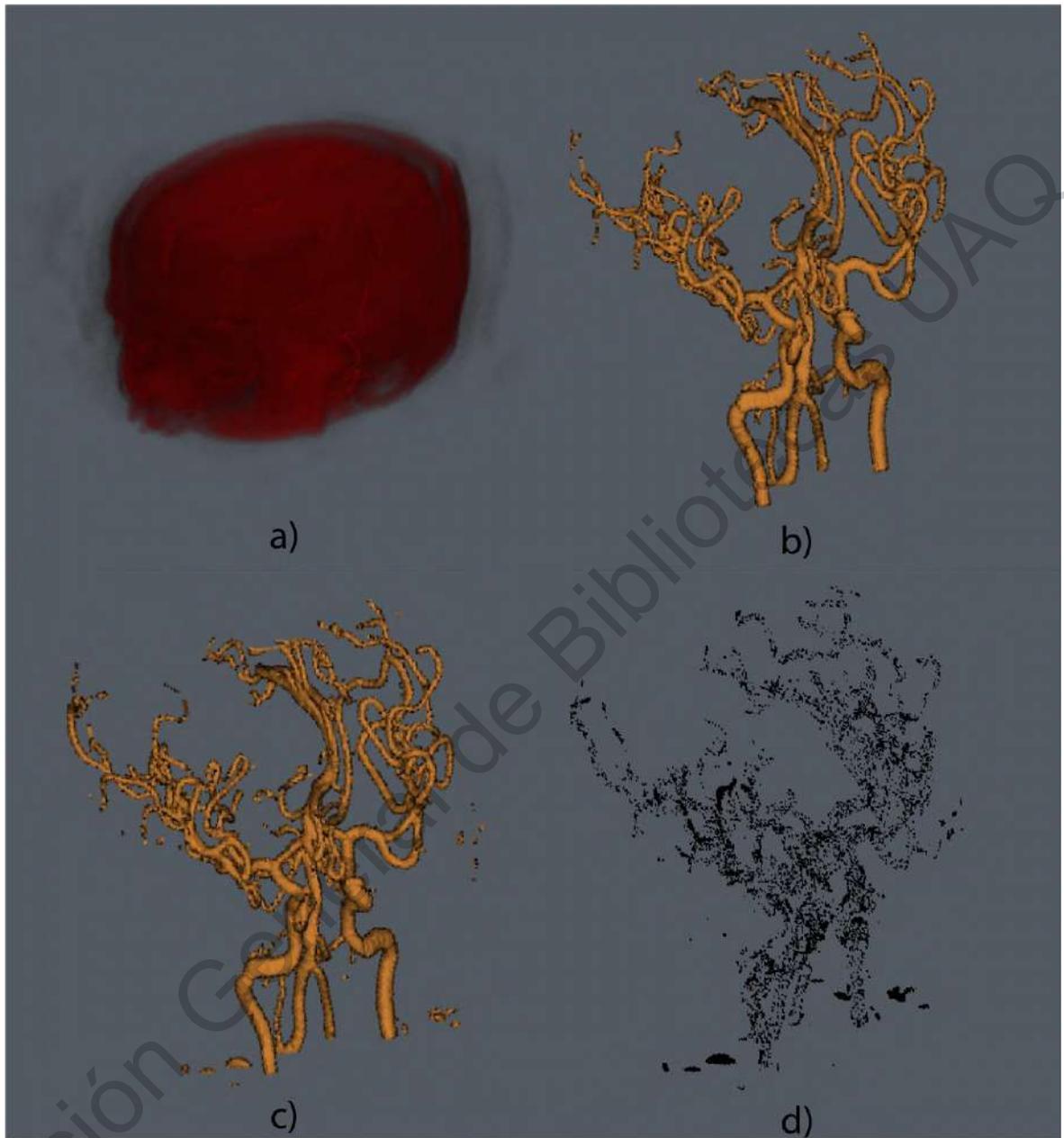


Figura A.2: Visualización del volumen 2 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.

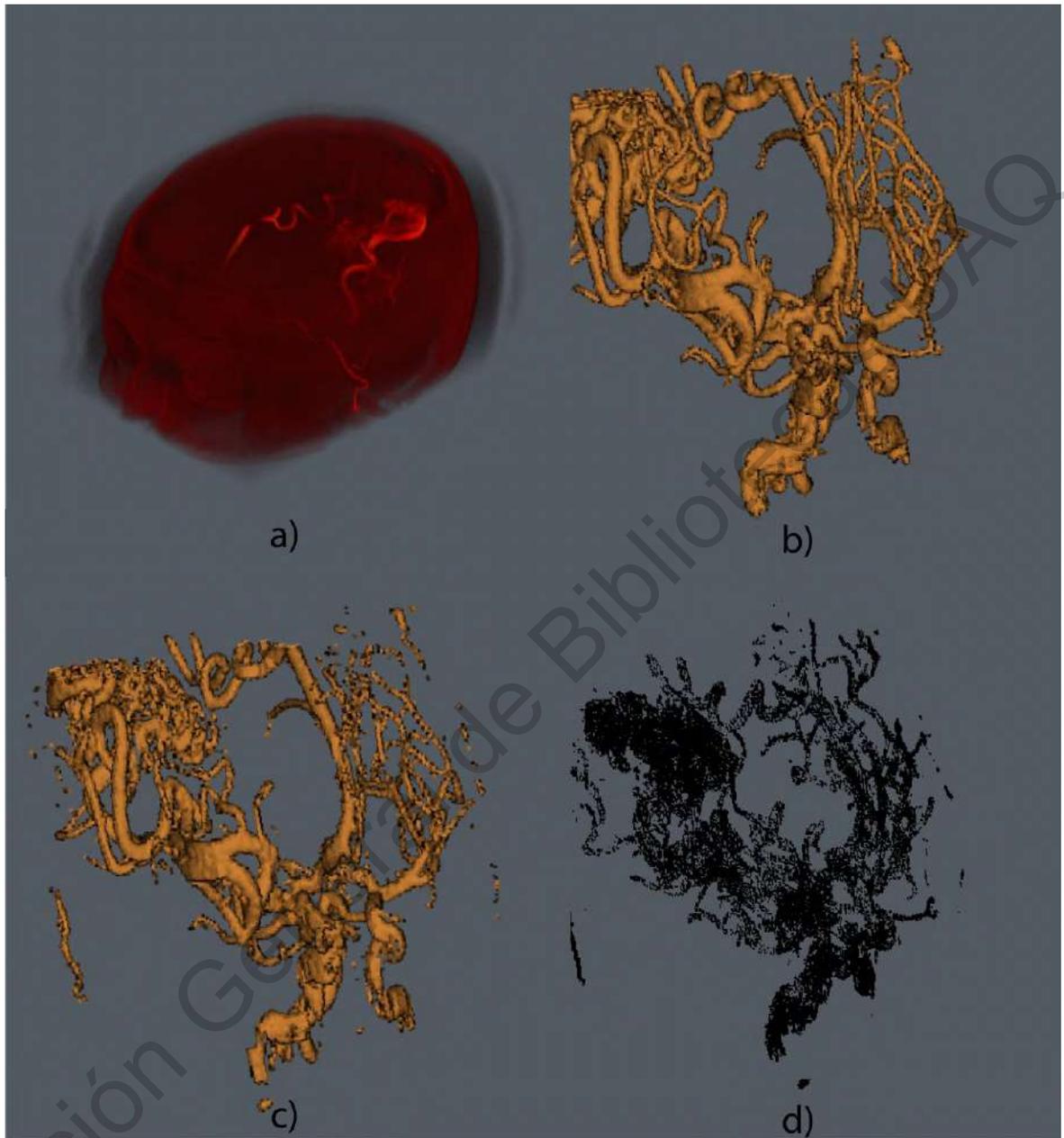


Figura A.3: Visualización del volumen 3 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.

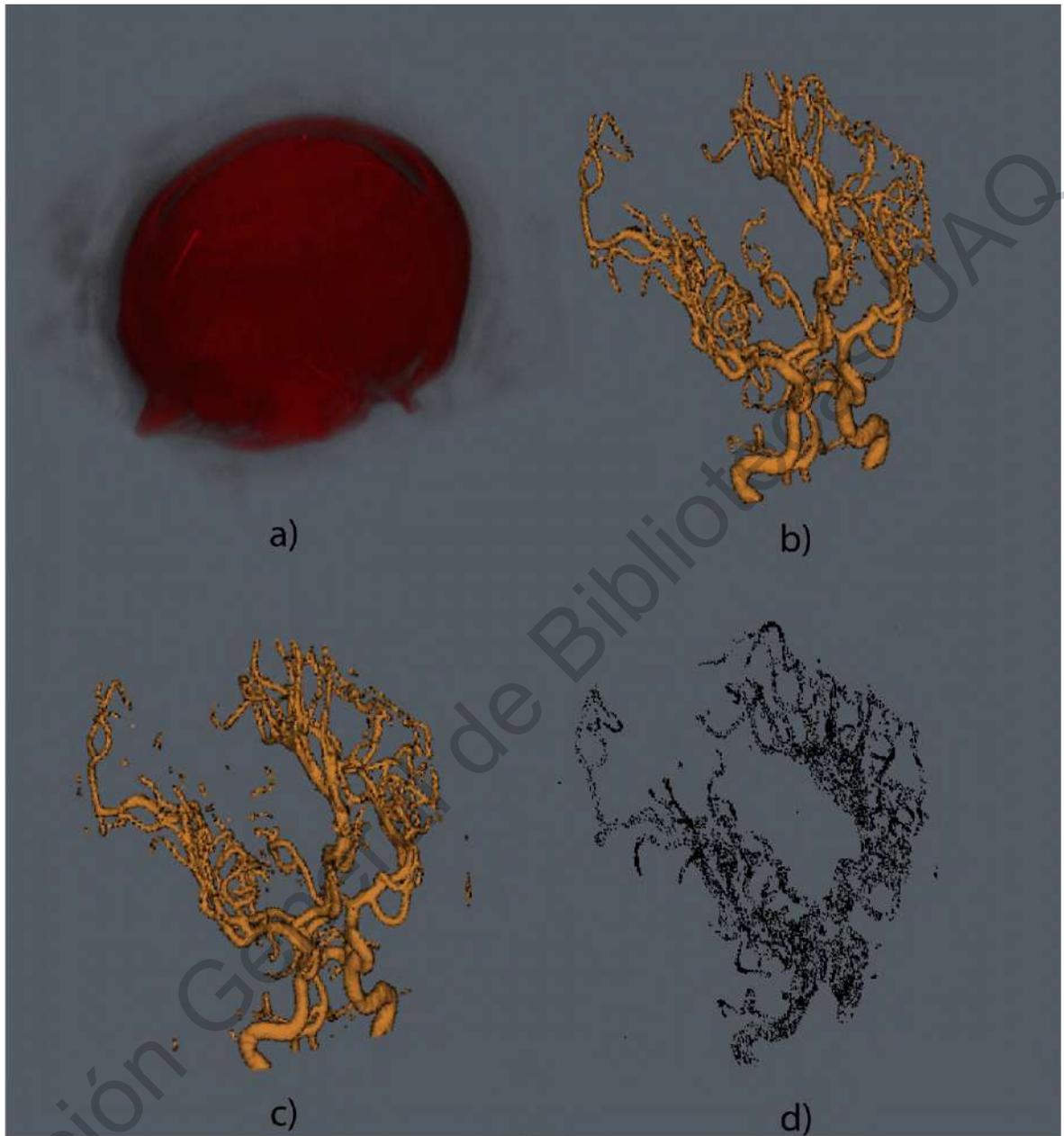


Figura A.4: Visualización del volumen 4 de MRA del conjunto de datos y resultados de la segmentación de vasos sanguíneos cerebrales mediante el software Aliza. a) Imágenes de MRA. b) Segmentación objetivo. c) Predicción por el método propuesto. d) Mapa de error entre la segmentación objetivo y la predicción.

A.2. Requerimientos para Obtención de Grado

UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE LENGUAS Y LETRAS

A QUIEN CORRESPONDA:

La que suscribe, Directora de la Facultad de Lenguas y Letras, hace **C O N S T A R** que

QUINTANA QUINTANA OLIVER JONATHAN

Presentó el **Examen de Manejo de la Lengua** efectuado el día veintinueve de noviembre de dos mil diecinueve, en el cual obtuvo la siguiente calificación:

8-

Se extiende la presente a petición de la parte interesada, para los fines escolares y legales que le convengan, en el Campus Aeropuerto de la Universidad Autónoma de Querétaro, el día seis de noviembre de dos mil veinte.

Atentamente,
"Enlazar Culturas por la Palabra"



LIC. LAURA PÉREZ TÉLLEZ



LPT/jals*CL*FLL-C.-1629

SOMOS UAQ
EDUCAR CRECER CONSOLIDAR

Campus Aeropuerto, Anillo Vial Fray Junípero Serra S/N, Querétaro, Qro., C.P. 76140
Tel. 442 192 12 00 Dirección, Ext. 61010, Secretaría Administrativa Ext.61300, Posgrado Ext. 61140,
Licenciatura Ext.61070, Centro de lenguas Ext.61050, Secretaría Académica Ext.61100 y Planeación Ext.61110

Figura A.5: Constancia de la realización del examen de manejo de la lengua por la Facultad de Lenguas y Letras de la UAQ.

La Mecatrónica en México



DICTÁMEN DE EVALUACIÓN

31 de agosto de 2021

OFICIO LMEM/20210831-1

Por este medio, el comité editorial de la revista LMEM se les informa a los autores que el trabajo titulado

Evaluación de modelos convolucionales en la segmentación de vasos sanguíneos cerebrales

Con número de registro **LMEM20210824-1**, cuyos autores son

Oliver Jonathan Quintana-Quintana, Marco Antonio Aceves-Fernández, Saúl Tovar-Arriaga*, Iván Villanueva-Naquid

Ha tenido como dictamen **APROBADO**. Por lo que se solicita apruebe el borrador correspondiente y concluya el trámite realizando el pago de \$2,000.00 MN a la cuenta

Banco: Grupo Financiero INBURSA

Titular: Asociación Mexicana de Mecatrónica AC

Cuenta: 50059161230

CLABE: 0361 8050 0591 6123 04

Revisor 1.

Dictámen: Aprobado

Revisor 2.

Dictamen: Aprobado

ATENTAMENTE,

**Comité Evaluador de los Artículos de la revista
"La Mecatrónica en México"**



Figura A.6: Carta de aceptación del artículo realizado para la revista "La Mecatrónica en México".