



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIERÍA
INDUSTRIAL VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA BIOMÉDICA

DESARROLLO DE UNA RED NEURONAL CONVOLUCIONAL VOLUMÉTRICA PARA LA SÍNTESIS DE IMAGEN EN RESONANCIA MAGNÉTICA NUCLEAR CEREBRAL

AUTORA: IRENE ESTELA GARCÍA GARCÍA

TUTOR: JOSÉ VICENTE MANJÓN HERRERA

Curso Académico: 2019-20

Agradecimientos

“ A mi familia, por haber hecho esto posible.
A Ilán, Álex y Laura, por haber sido fundamentales para llegar hasta aquí.
A mi tutor, por enseñarme y transmitirme su pasión. ”

Resumen

En Resonancia Magnética (RM), dependiendo de los parámetros de adquisición de las imágenes, es posible obtener una gran variedad de contrastes distintos, que muestran información complementaria del objeto bajo estudio. A pesar de que esta información multispectral trae consigo una mejora de la comprensión del estado de los tejidos cerebrales, la adquisición de múltiples contrastes no siempre es posible debido a limitaciones temporales y de comodidad del paciente. La síntesis de contrastes ha surgido recientemente como una solución aproximada para generar otros tipos de imagen diferentes de aquellos adquiridos originalmente pues, a pesar de todavía no haber conseguido resultados aptos para el diagnóstico clínico directo, sí que resulta útil para la mejora de procesos importantes tales como el registro o la segmentación: aumentando el número de modalidades de entrada a estos métodos, se incrementa la cantidad de información disponible mejorando así el desempeño del método. La mayoría de los métodos propuestos previamente para síntesis de imagen son basados en cortes, lo que puede resultar en inconsistencias de intensidad entre cortes vecinos cuando estos son aplicados en 3D. En el presente trabajo se propone el uso de una red neuronal convolucional (CNN) 3D capaz de generar una imagen T2 o FLAIR cerebral a partir de una imagen anatómica común T1 como imagen de origen. La red de aprendizaje profundo propuesta es una variante 3D de la conocida arquitectura UNet que procesa el volumen completo simultáneamente rompiendo con las inconsistencias en los volúmenes de salida resultantes relacionadas con los métodos basados en cortes 2D o patches, incrementando la coherencia en el volumen sintetizado y mejorando la precisión vía una perspectiva tridimensional integrada del contexto. Este enfoque holístico contribuye a generar volúmenes de salida más precisos tanto visual como cuantitativamente, a través de un aumento de la información contextual disponible. Finalmente, se ha validado el método propuesto con un método de segmentación, demostrando así su utilidad en una aplicación directa.

Palabras Clave: Síntesis de imagen, Resonancia Magnética (RM) cerebral, aprendizaje profundo, red neuronal convolucional (CNN), segmentación.

Resum

En Ressonància Magnètica (RM), depenent dels paràmetres d'adquisició de les imatges, és possible obtenir una gran varietat de contrastos diferents, que mostren informació complementària de l'objecte baix estudi. A pesar que aquesta informació multiespectral comporta una millora de la comprensió de l'estat dels teixits cerebrals, l'adquisició de diversos tipus d'imatges no sempre és possible a causa de limitacions temporals i de comoditat del pacient. La síntesi de contrastos ha sorgit recentment com una solució aproximada per a generar altres tipus d'imatge diferents d'aquells adquirits originàriament perquè, a pesar d'encara no haver aconseguit resultats aptes per al diagnòstic clínic directe, sí que resulta útil per a la millora de tasques importants com ara el registre o la segmentació: augmentant el nombre de modalitats d'entrada a aquests mètodes, s'incrementa la quantitat d'informació disponible millorant així els resultats del mètode. La majoria dels mètodes proposats prèviament per a síntesi d'imatge són basats en talls, cosa que pot resultar en inconsistències d'intensitat entre talls veïns quan aquests són aplicats en 3D. En el present treball es proposa l'ús d'una xarxa neuronal convolucional (CNN) 3D capaç de generar una imatge T2 o FLAIR cerebral a partir d'una imatge anatòmica T1 comú o una imatge FLAIR com a imatge d'origen. La xarxa d'aprenentatge profund proposada és una variant 3D de la coneguda arquitectura UNet que processa el volum complet simultàniament trencant amb les inconsistències en els volums d'eixida resultants relacionades amb els mètodes basats en talls 2D o patches, incrementant la coherència en el volum sintetitzat i millorant la precisió via una perspectiva tridimensional integrada del context. Aquest enfocament holístic contribueix a generar volums d'eixida més precisos tant visualment com quantitativament, a través d'un augment de la informació contextual disponible. Finalment, s'ha validat el mètode proposat amb un mètode de segmentació demostrant així la seua utilitat en una aplicació directa.

Paraules clau: Síntesi d'imatge, Ressonància Magnètica (RM) cerebral, aprenentatge profund, xarxa neuronal convolucional (CNN), segmentació.

Abstract

In Magnetic Resonance Imaging (MRI), depending on the image acquisition settings, a large number of image types or contrasts can be generated showing complementary information of the same imaged object. Although this multispectral information is beneficial to improve our understanding of the state of the living brain tissues, acquisition of several contrasts is not always possible due to time limitations and patient comfort constraints. Contrast synthesis has emerged recently as an approximate solution to generate other image types different from those acquired originally. Despite not having yet achieved suitable results for direct clinical diagnosis use, it is useful for the improvement of important tasks such as registration or segmentation: by increasing the number of input image modalities to these methods, the amount of available information is increased, thus improving the performance of the method. Most of the previously proposed methods for contrast synthesis are slice-based which may result in intensity inconsistencies between neighbour slices when applied in 3D. The use of a 3D convolutional neural network (CNN) capable of generating a T2-weighted or FLAIR MRI target contrast from a common anatomical T1-weighted source image is proposed. The proposed deep learning network is a 3D variant of the UNet that processes the whole volume at once breaking with the inconsistency in the resulting output volumes related to 2D slice or patch-based methods, enhancing the coherence in the synthesized volume and improving the accuracy via the integrated three-dimensional context awareness. This holistic approach contributes to generating more accurate output volumes both visually and quantitatively, through an increase in available contextual information. Eventually, the proposed method has been validated with a segmentation method, thus demonstrating its usefulness in a direct application.

Keywords: Image synthesis, brain Magnetic Resonance Imaging (MRI), deep learning, convolutional neural network (CNN), segmentation.

Índice de figuras

1.1. Esquema general del funcionamiento de la imagen de RM.	5
1.2. a) Esquema de los dos estados del espín nuclear del H^1 y b) esquema del movimiento de <i>spinning</i> y de precesión del núcleo de H1. Fuente: Gili 1993.	6
1.3. a) Esquema de los espines orientados en sus dos estados bajo el campo magnético y b) esquema del vector de magnetización \vec{M} en un vóxel. Fuente: Gili 1993.	7
1.4. Esquema del movimiento de nutación realizado por el vector magnetización \vec{M} durante la excitación/relajación nuclear y la FID generada en ésta última. Fuente: Gili 1993.	8
1.5. Estudio de la relajación mediante la evolución en el tiempo de las componentes longitudinal M_z y transversal M_{xy} del vector magnetización \vec{M} . Fuente: Gili 1993	9
1.6. Tres curvas de relajación longitudinal distintas de tres vóxels con la misma D. El vóxel con un T1 menor tiene una relajación más rápida. Al leer la señal tras un tiempo tau, obtenemos valores distintos que reflejan los distintos T1. Fuente: Gili 1993	10
1.7. Dos señales de relajación transversal distintas de dos vóxels con la misma D. El decrecimiento de la señal de relajación transversal es mayor cuanto menor es el T2 del vóxel. Fuente: Gili 1993	11
1.8. Imágenes de RM cerebral potenciadas en T1, T2 y FLAIR de un mismo sujeto. Fuente: School of Medicine, Case Western Reserve University 2006	12
1.9. Partes mayores del encéfalo (a) y división en lóbulos del cerebro (b). Fuente: Modificado de Johns Hopkins Medicine s.f. y Mayo Clinic s.f.	13
1.10. Atlas anatómico del cerebro. Fuente: Craig, Perkins y Netter 2002	14
1.11. Imagen de RM cerebral T1 (izquierda) y su correspondiente segmentación con una red UNet (derecha).	17

1.12. Histograma de una imagen T1 de un cerebro adulto. Los picos de los 3 tejidos se basan en una segmentación manual y las distribuciones difieren ligeramente de la distribución gaussiana debido al efecto de volumen parcial. Fuente: Despotović, Goossens y Philips 2015	18
1.13. Esquema de la relación entre inteligencia artificial, aprendizaje automático y aprendizaje profundo y sus respectivas definiciones.	19
1.14. Esquema de la analogía entre una neurona biológica (arriba) y una neurona artificial (abajo). Fuente: <i>Las redes neuronales</i> 2006 y Deshpande 2017	21
1.15. Esquema de la organización de neuronas en capas interconectadas para formar una red. Fuente: OpenNN, Open Neural Networks s.f.	23
1.16. Esquema ilustrativo de la navegación por la superficie de error o función de coste con la técnica <i>gradient descent</i> . C es la función de coste, ω_1 y ω_2 son los parámetros de la red. Fuente: Modificado de Hutson 2018	25
1.17. Esquema ilustrativo de la operación de convolución 2D entre una imagen de 5x5 y un kernel de 3x3, con <i>zero padding</i> o relleno con ceros. Fuente: Modificado de <i>TensorFlow Conv2D Layers: A Practical Guide</i> s.f. y Hus-sain y col. 2018	28
1.18. Esquema ilustrativo de las operaciones de <i>max pooling</i> y <i>average pooling</i> . Fuente: Saha 2018	29
1.19. Esquema de ejemplo de arquitectura de una CNN. Fuente: <i>Convolutional Neural Networks (LeNet)</i> 2018	29
1.20. Esquema de la arquitectura original de la red U-Net propuesta por Ronneberger, Fischer y Brox 2015 a modo de ejemplo de CAE. Cada caja azul corresponde a un mapa de activación multi-canal. El número de canales se muestra sobre cada caja. Las dimensiones x-y se muestran en la esquina inferior izquierda de cada caja. Las cajas blancas representan los mapas de activación copiados y concatenados. Fuente: Ronneberger, Fischer y Brox 2015	31
1.21. Gráficos ilustrativos de las situaciones de <i>underfitting</i> , <i>overfitting</i> y un buen ajuste a los datos. Los puntos representan datos de entrenamiento pertenecientes a dos clases distintas (rojo y verde) y la línea gris el límite de clasificación establecido por el método para una tarea de clasificación. Fuente: Parmar 2018	31
1.22. Esquema del movimiento en convoluciones con <i>stride 1</i> y <i>2</i> . Fuente: <i>Intuitive understanding of 1D, 2D, and 3D convolutions in convolutional neural networks</i> 1966	33
4.1. Esquema de la metodología seguida en el proyecto.	41

4.2. Corte axial de imagen de RM potenciada en T1 de la base de datos IXI, antes y después del filtrado de ruido con el filtro adaptativo NLM (izquierda y derecha respectivamente).	43
4.3. Proceso de registro de las imágenes: corrección T1-T2 seguido de registro al espacio MNI152.	44
4.4. Esquema normalización de intensidad con <i>histogram matching</i> : paso de espacios nativos de intensidad diferentes a un mismo espacio de intensidad estándar.	45
4.5. Mapeo de intensidades en histogram matching. CDF = <i>Cumulative Density Function</i> (histograma acumulado), en rojo la imagen a normalizar, y el azul la imagen de referencia. Fuente: Gonzalez, Woods y Eddins 2004	46
4.6. Atlas FLAIR de 52 sujetos.	47
4.7. Descomposición del volumen de entrada en 8 sub-volúmenes (8 canales).	48
4.8. Esquema de la CNN 3D propuesta para síntesis de contrastes en IRM.	50
4.9. Esquema de la metodología de validación del método vía aplicación a segmentación.	56
5.1. Imagen T2 objetivo (izquierda) e imágenes T2 sintetizadas para entrada no normalizada (centro) y entrada normalizada con <i>histogram matching</i> (derecha).	60
5.2. Imágenes T2 sintetizadas por la UNet 2D (fila superior) y por el método propuesto (fila inferior). Se aprecian los artefactos inter-corte generados por la UNet 2D, precisamente por la falta de contextualización 3D (flechas rojas).	61
5.3. Volúmenes T2 sintetizados por una red UNet 3D (fila superior) y por nuestro método (fila inferior). Se aprecia la incapacidad de la UNet 3D de reconstruir detalles finos (flechas rojas).	62
5.4. Fila superior, de izquierda a derecha: imagen T2 <i>ground-truth</i> , imágenes sintetizadas por REPLICA, el método propuesto y pGAN. Fila inferior, de izquierda a derecha: imagen de entrada T1, imagen T2 <i>ground-truth</i> , imágenes sintetizadas por pGAN y el método propuesto. Nótese que REPLICA no es capaz de recuperar las pequeñas áreas oscuras en el centro del cerebro y que pGAN introduce artefactos dependientes de los cortes 2D (flechas rojas).	63

5.5. Imagen T1 original e imágenes T2 y FLAIR sintetizadas por el método (fila superior, de izquierda a derecha) e imágenes segmentadas *ground-truth*, segmentada con la UNet con entrada monomodal y con entrada trimodal (fila inferior, de izquierda a derecha). Nótese la mayor precisión de la segmentación trimodal en detalles como el indicado por las flechas rojas. 67

Índice de tablas

1.1. Analogía de las partes de una neurona artificial con una biológica. . . .	21
1.2. Ejemplos de funciones de activación típicas para las neuronas artificiales.	22
1.3. Ejemplos de funciones de coste típicas para el entrenamiento de una RNA.	26
3.1. Especificaciones del ordenador empleado para el proyecto.	39
5.1. Resultados de test de la comparación de las funciones de coste clásicas MAE y MSE, y la función de coste propuesta. Mejores resultados en negrita.	59
5.2. Resultados de test de la comparación de la red propuesta con la UNet 2D y la UNet 3D básica. Mejores resultados en negrita.	60
5.3. Resultados de test del método propuesto, REPLICA y pGAN. Mejores resultados en negrita.	63
5.4. Resultados de test de la comparación de las funciones de coste clásicas MAE y MSE, y la combinación de ambas. Mejores resultados en negrita.	65
5.5. Resultados de test de la comparación de la normalización con HM respecto a 1 sujeto, respecto al atlas de 52 sujetos y sin normalización. Mejores resultados en negrita.	65
5.6. Tabla comparativa mejores resultados obtenidos en la síntesis de T2 y en la síntesis de FLAIR.	66
5.7. Resultados de test de los experimentos con la red de segmentación monomodal, bimodal I y II, y trimodal. Mejores resultados en negrita.	67
6.1. Tabla de precios de mano de obra	74
6.2. Tabla de precios de materiales, equipos y <i>software</i>	74
6.3. Tabla de presupuestos parciales del Capítulo 1.	75

6.4. Tabla de presupuestos parciales del Capítulo 2.	75
6.5. Tabla de presupuestos parciales del Capítulo 3.	75
6.6. Tabla de presupuestos parciales del Capítulo 4.	75
6.7. Tabla de precios unitarios.	76
6.8. Tabla de precios descompuestos.	78
6.9. Tabla de presupuesto de ejecución por contrata.	78

Lista de acrónimos

- AA**, *Aprendizaje Automático*, 20
- AI**, *Artificial Intelligence*, 20
- ANTs**, *Advanced Normalization Tools*, 39, 43, 47
- AP**, *Aprendizaje Profundo*, 3, 19, 20
- API**, *Application Programming Interface*, 38
- BS**, *Brainstem*, 56
- CAE**, *Convolutional Autoencoder*, 19, 30
- CC**, *Coeficiente de correlación de Pearson*, 27, 52–54, 59
- CeGM**, *Cerebelar Grey Matter*, 56
- CeWM**, *Cerebelar White Matter*, 56
- CGM**, *Cortical Grey Matter*, 56
- CNN**, *Convolutional Neural Network*, 3, 19, 27–30, 32, 35, 42, 45, 47, 49, 50
- CSF**, *Cerebrospinal Fluid*, 17, 56
- DL**, *Deep Learning*, 3, 16, 19, 20, 31, 32, 38
- DSE**, *Dual Spin Echo*, 11
- FID**, *Free Induction Decay*, 8
- FLAIR**, *Fluid-Attenuated Inversion Recovery*, 10–12, 34, 35, 37–39, 42, 46, 47, 50–52, 55, 56, 59, 64–67, 69
- GAN**, *Generative Adversarial Network*, 55
- GDL**, *Gradient Difference Loss*, 54, 59
- GM**, *Grey Matter*, 17
- HM**, *Histogram Matching*, 46, 53, 55, 60, 65
- IRM**, *Imagen de Resonancia Magnética*, 3–5, 9, 11, 12, 15, 17, 18, 33–36, 42, 55, 60, 65, 66, 69
- LCR**, *Líquido Cefalorraquídeo*, 10, 11, 13, 14, 17, 63
- MAE**, *Mean Absolute Error*, 27, 51, 54, 55, 59, 64
- ML**, *Machine Learning*, 16, 20, 38
- MNI**, *Montreal Neurological Institute*, 37, 43, 44, 46, 47, 64
- MPRAGE**, *Magnetization-prepared gradient echo*, 10
- MSE**, *Mean Square Error*, 27, 38, 51, 54, 55, 59, 64
- NLM**, *Non Local Means*, 42
- NMI**, *Normalised Mutual Information*, 43, 44
- PSNR**, *Peak Signal-to-Noise Ratio*, 27, 38, 52, 53
- ReLU**, *Rectified Linear Unit*, 22, 28, 30, 48, 49
- RF**, *Radiofrecuencia*, 8, 9
- RM**, *Resonancia Magnética*, 4, 5, 7–9, 14, 15, 35, 37, 42, 44, 45, 61, 67
- RMN**, *Resonancia Magnética Nuclear*, 3
- RMSE**, *Root Mean Square Error*, 27, 52
- RNA**, *Red neuronal artificial*, 20, 23
- SCGM**, *SubCortical Grey Matter*, 56
- SGD**, *Stochastic Gradient Descent*, 25
- SPGR**, *Spoiled gradient recalled*, 10
- SPM**, *Statistical Parametric Mapping*, 39
- SPM**, *Statistical Parametric Mapping*, 53
- SSIM**, *Structural Similarity Index Measure*, 27, 52, 53
- TAC**, *Tomografía Axial Computarizada*, 4, 45

TE, *Tiempo de Eco*, 37, 45

TR, *Tiempo de Repetición*, 9, 37, 45

WM, *White Matter*, 17, 56

Índice general

Resumen	V
Resum	VII
Abstract	IX
Lista de figuras	XI
Lista de Tablas	XIV
Lista de acrónimos	XVI
Índice general	XIX
I Memoria	1
1 Introducción	3
1.1 Imagen de Resonancia Magnética (IRM)	3
1.1.1 Introducción a la IRM	3
1.1.2 Funcionamiento general	4
1.1.3 Fundamentos físicos	5
1.1.4 Potenciaciones en IRM	9
1.1.5 IMR en Neuroimagen	12
1.2 Anatomía del cerebro humano	13
1.3 Síntesis de Imagen	15
1.3.1 Introducción a la síntesis de imagen	15
1.3.2 Perspectiva general de métodos y estado del arte en síntesis	15
1.4 Segmentación	17
1.4.1 Introducción a la segmentación	17
1.4.2 Perspectiva general de métodos y estado del arte en segmentación	17

1.5	<i>Deep Learning</i> en Redes Neuronales Artificiales	19
1.5.1	Introducción al <i>Deep Learning</i>	19
1.5.2	Fundamentos de las RNA	20
1.5.3	Redes Neuronales Convolucionales (CNN)	27
1.5.4	Autoencoders Convolucionales (CAE)	30
1.5.5	Conceptos	31
1.6	Motivación y Justificación	33
2	Objetivos	35
2.1	Objetivo general	35
2.2	Objetivos específicos	35
3	Materiales	37
3.1	Datos	37
3.2	Software	38
3.3	Hardware	39
4	Metodos	41
4.1	Metodología general	41
4.2	Preprocesado	42
4.2.1	Imágenes T1/T2 (Base de datos I)	42
4.2.2	Imágenes T1/FLAIR (Base de datos II)	46
4.3	Arquitectura de red	47
4.4	Entrenamiento y evaluación de las redes	50
4.4.1	Entrenamiento	50
4.4.2	Evaluación	52
4.4.3	Experimentos	53
4.5	Validación del método propuesto: Segmentación	56
5	Resultados y Discusión	59
5.1	Resultados red T1/T2	59
5.2	Resultados red T1/FLAIR	64
5.3	Resultados validación. Segmentación	66
6	Conclusiones	69
II	Presupuesto	71
6.1	Introducción	73
6.2	Cuadro de Mano de Obra	74
6.3	Cuadro de materiales, equipos y <i>software</i>	74
6.4	Cuadro de presupuestos parciales	75
6.5	Cuadro de precios unitarios	76
6.6	Cuadro de precios descompuestos	77
6.7	Presupuesto de ejecución por contrata	78

Parte I

Memoria

Capítulo 1

Introducción

En el presente capítulo se introduce el marco teórico del trabajo. En primer lugar, se realiza una explicación introductoria a la Imagen de Resonancia Magnética (IRM), su funcionamiento y fundamentos físicos, así como su papel en neuroimagen y las potenciaciones relevantes para este trabajo en concreto. Seguidamente, se exponen de forma breve los aspectos principales de la anatomía cerebral, para seguidamente entrar en materia sobre las dos técnicas de análisis de imagen médica de importancia para el presente trabajo: la síntesis de imagen y la segmentación, incluyendo una revisión resumida de los métodos y el estado del arte. Por último, se explica la teoría concerniente a las técnicas de aprendizaje profundo (AP) o Deep Learning (DL), haciendo hincapié en las redes neuronales convolucionales (CNN) y su aplicación en el campo de la imagen médica. Concluido el marco teórico, se expone finalmente la motivación y justificación del trabajo.

1.1 Imagen de Resonancia Magnética (IRM)

1.1.1 Introducción a la IRM

La Imagen de Resonancia Magnética (IRM), Imagen por Resonancia Magnética Nuclear o, menos comúnmente, Tomografía por Resonancia Magnética se trata de una técnica de imagen médica basada, como bien su nombre indica, en el fenómeno fisicoquímico de la Resonancia Magnética Nuclear (RMN).

Dicho fenómeno fue descrito y medido por primera vez en 1938 por Isidor Isaac Rabi, y en 1946, de forma independiente, dos científicos en Estados Unidos, Edward M. Purcell y Felix Bloch refinaron la técnica, siendo galardonados por ello con el premio Nobel en Física en 1952. La historia de la IRM incluye el trabajo de múltiples investigadores que, como los mencionados, contribuyeron al descubrimiento de la resonancia magnética nuclear, así como aquellos que investigaron su aplicación médica y describieron la física subyacente de la imagen por resonancia magnética, hasta llegar a la instalación de las primeras máquinas de IRM en hospitales para uso clínico a principios de los 80 (Rinck 2019). Desde entonces, la técnica ha experimentado un

rápido desarrollo tecnológico, conduciendo a su uso generalizado en la actualidad.

Son varios los factores que influyen en el éxito de esta técnica y la han convertido en una de las más empleadas a nivel clínico. Una de las principales causas de dicho éxito es su inocuidad pues, a diferencia de otras técnicas de imagen médica como son los Rayos X o la Tomografía Axial Computarizada (TAC), la IRM no emplea radiación ionizante para acceder a la información del interior del cuerpo humano.

Otra de las características que han llevado a la IRM a ser comúnmente utilizada es la rapidez en la adquisición de las imágenes. Este corto tiempo de adquisición permite realizar estudios dinámicos más allá de la simple obtención de una imagen estática. Además, el amplio abanico de tipos de imagen adquiribles dota a la IRM de una gran versatilidad, siendo capaz de proporcionar gran cantidad de información muy diversa y complementaria del mismo objeto tridimensional bajo estudio. Todo ello otorga a esta técnica un enorme potencial diagnóstico, siendo actualmente la IRM una de las herramientas de diagnóstico por imagen de uso más extendido en los diversos ámbitos clínicos.

Una explicación rigurosa y en profundidad del funcionamiento de la adquisición de la IRM y los fundamentos físicos subyacentes es larga y compleja, implicando incluso explicaciones de mecánica cuántica, por lo que queda más allá del propósito del presente trabajo. Por ello, a continuación se ha tratado de describir el proceso de una manera clara, pero con un nivel de profundidad suficiente para comprender el funcionamiento básico de la técnica.

1.1.2 Funcionamiento general

La IRM se basa en el fenómeno fisicoquímico de la Resonancia Magnética (RM), por el cual ciertas partículas tales como protones, electrones y núcleos atómicos con un número impar de protones Z y/o un número impar de neutrones N son capaces de absorber energía de radiofrecuencia de forma selectiva al ser sometidos a un campo magnético \vec{B} potente.

Concretamente, las imágenes de RM empleadas en el ámbito clínico explotan esta característica del núcleo de Hidrógeno (H^1) con $Z=1$, presente en los tejidos corporales. A grosso modo, los núcleos son irradiados bajo un campo magnético con pulsos de radiofrecuencia, absorben esta energía (resonancia) y devuelven el exceso energético a través de una liberación de ondas de radiofrecuencia (relajación). Esta liberación energética induce una señal eléctrica en una antena receptora, que constituye la señal de RM a partir de la cual se obtiene la imagen (Figura 1.1). Así, cada vóxel (elemento volumétrico) de la imagen se adquiere a partir de una señal que proviene de la relajación de los núcleos de H^1 de un cierto volumen de tejido.

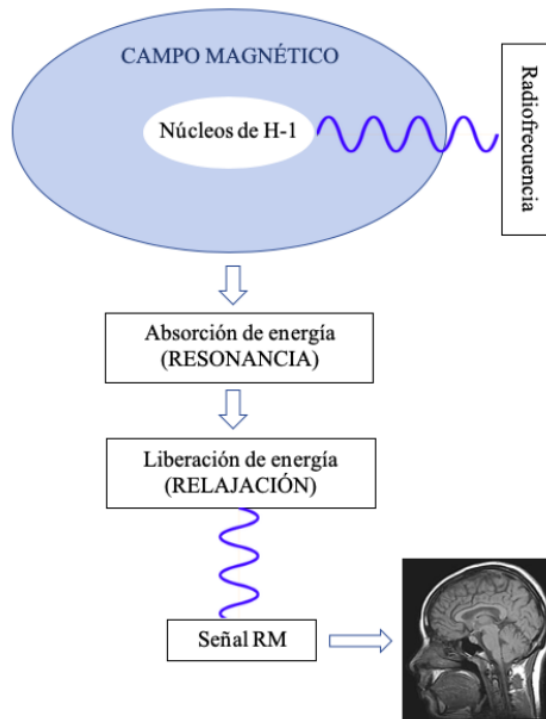


Figura 1.1: Esquema general del funcionamiento de la imagen de RM.

Esta señal de relajación producida por los núcleos de H^1 depende de múltiples parámetros tanto externos (propios de la máquina) como internos (propios del tejido), como son la intensidad del campo magnético y el tipo de molécula en la que se encuentra el núcleo de H^1 , respectivamente. Esto implica que la señal detectada contenga multitud de información acerca de los tejidos subyacentes desde distintos puntos de vista en función de los parámetros externos seleccionados. Existen múltiples potenciaciones posibles de las imágenes según los parámetros de interés, de ahí la riqueza y versatilidad de la IRM, siendo posible obtener imágenes muy diversas tanto morfológicas como funcionales.

1.1.3 Fundamentos físicos

La máquina de RM posee un imán que crea un potente campo magnético en su interior. Cuando colocamos un paciente bajo el campo magnético, en cada volumen de su organismo que contenga núcleos de H^1 aparecen propiedades magnéticas por tener el núcleo una carga eléctrica, se dice así que se produce una magnetización del vóxel. El origen de estas propiedades magnéticas es la resultante del comportamiento a nivel cuántico de los millones de núcleos de H1 contenidos en un vóxel de tejido.

Desde una perspectiva cuántica, los núcleos de H^1 rotan sobre sí mismos en un movimiento de *spinning* (Figura 1.2b) que se representa por un vector de spin \vec{s} en la dirección del eje de giro que, al tener el núcleo una carga eléctrica, implica unas propiedades magnéticas que se representan por un vector momento magnético $\vec{\mu}$ también orientado sobre el eje de giro, siendo estos dos vectores proporcionales. El efecto Zeeman enuncia que si colocamos un núcleo con

$s \neq 0$ en un campo magnético \vec{B} , podemos encontrarlo en $2s + 1$ estados energéticos. El valor del número cuántico s del núcleo de H^1 es $1/2$, por lo que este se puede hallar en 2 estados energéticos ($m = -1/2$, $m = +1/2$): una posición de mínima energía (estado *up* o paralelo a \vec{B}) y una de máxima energía (estado *down* o antiparalelo a \vec{B}), tal y como se puede observar en la Figura 1.2a. Además, los núcleos realizan un movimiento de giro alrededor de la dirección de \vec{B} , denominado movimiento de precesión (Figura 1.2b), a una determinada frecuencia de giro denominada frecuencia de precesión o de resonancia (f_p), proporcional al valor de campo magnético percibido por el núcleo, tal y como establece la Ley fundamental de la Resonancia Magnética o Ley de Larmor (ecuación 1.1).

$$f_p = \gamma \cdot \frac{B}{2\pi} \text{ (Hz)} \quad (1.1)$$

Donde B es el valor del campo magnético percibido por el núcleo y γ es el cociente giromagnético específico del núcleo.

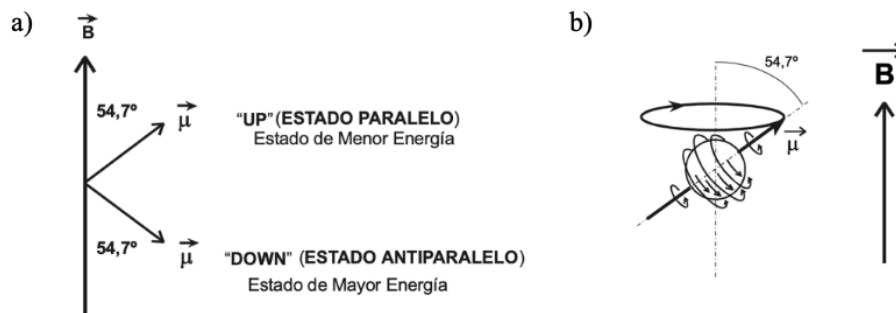


Figura 1.2: a) Esquema de los dos estados del espín nuclear del H^1 y b) esquema del movimiento de *spinning* y de precesión del núcleo de H^1 . Fuente: Gili 1993.

Este campo magnético percibido por el núcleo está formado por tres componentes (Ecuación 1.2):

- El campo magnético del imán (\vec{B}_0) uniforme, del orden de unidades de Teslas.
- El campo magnético de las bobinas o gradientes (\vec{B}_{GRAD}) que generan una variación del campo magnético para la codificación espacial de los vóxeles, del orden de miliTeslas.
- El campo del entorno bioquímico (\vec{B}_{BIOQ}) del tejido en el que se hallan los núcleos, del orden de microTeslas.

$$\vec{B} = \vec{B}_0 + \vec{B}_{GRAD} + \vec{B}_{BIOQ} \quad (1.2)$$

Los núcleos de H^1 pueden estar ubicados en radicales $-OH$ de moléculas de agua o en radicales CH_2CH_3 de lípidos, siendo su entorno bioquímico distinto y percibiendo así distintos \vec{B}_{BIOQ} . Por tanto, al someter un elemento de volumen del paciente o vóxel a un campo magnético \vec{B}_0 , los espines de los diversos núcleos de H^1 contenidos en él se orientarán en una de las dos posiciones (la mayor parte en estado *up* o de mínima energía) y se ven forzados a precesar (Figura 1.3a). No obstante, según la Ley de Larmor (ecuación 1.1), sus frecuencias de precesión serán

ligeramente distintas al depender del entorno bioquímico, por lo que se desfasan unos respecto a los otros, alcanzando una orientación al azar. Así, si imaginamos todos los espines de un vóxel trasladados al punto central de éste, tendríamos un vector resultante que representa la magnetización del elemento de volumen (\vec{M}), cuya magnitud depende de la densidad de núcleos. Al haber una mayor cantidad de los estados menos energéticos o *up*, la magnetización del vóxel tiene el sentido y la dirección de \vec{B} . Esto queda ilustrado de forma esquemática en la Figura 1.3b.

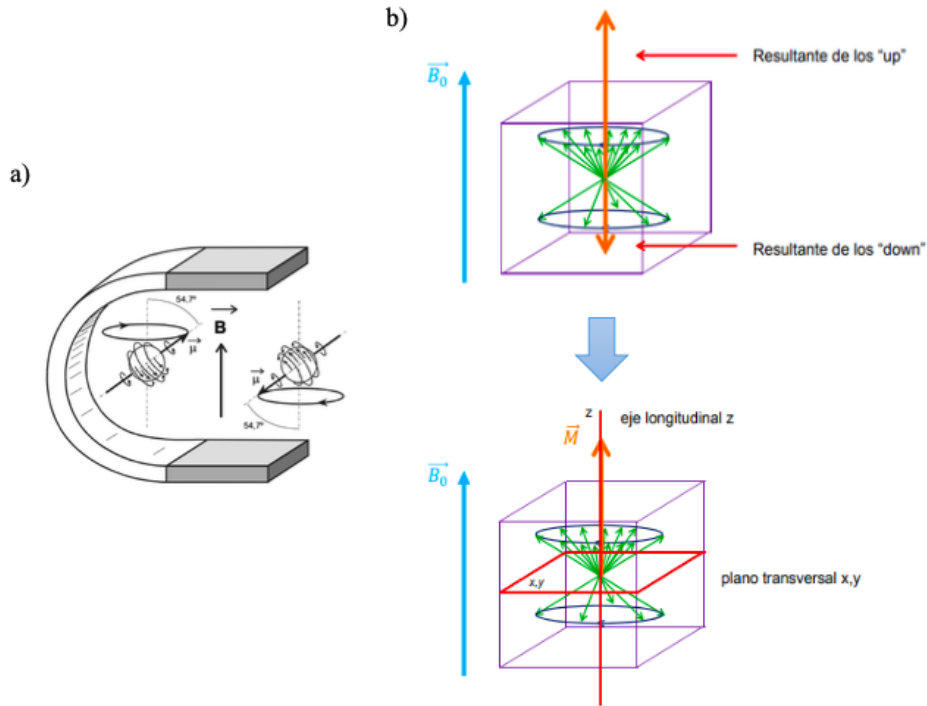


Figura 1.3: a) Esquema de los espines orientados en sus dos estados bajo el campo magnético y b) esquema del vector de magnetización \vec{M} en un vóxel. Fuente: Gili 1993.

En estado de reposo, el vector \vec{M} está sobre la dirección z (la del campo magnético), y su valor es la componente longitudinal, mientras que su proyección sobre el plano transversal es nula (Figura 1.3b), por la orientación al azar de los espines desfasados. Los núcleos deberán ser excitados para que el vector magnetización cambie. Un núcleo de H^1 bajo un campo magnético puede pasar del estado menos energético al estado más energético absorbiendo una determinada energía ΔE , proceso denominado excitación (Figura 1.4). Esta variación energética coincide con la de una onda electromagnética de frecuencia f :

$$f = \frac{\Delta E}{h} = \gamma \cdot \frac{B}{2\pi} (\text{Hz}) \quad (1.3)$$

Esta es la frecuencia de resonancia f que coincide con la de precesión (ecuación 1.1), donde h es la constante de Planck. Teniendo en cuenta que la constante γ vale $42,58 \text{ MHz/T}$ para el núcleo de H^1 y los campos utilizados en RM clínica están entre $1,5$ y $3T$, se deduce que las frecuencias que permiten el paso entre estados energéticos están en el rango de $63,4 \text{ MHz}$ a $126,7 \text{ MHz}$, es

decir, en la zona de radiofrecuencia (RF) del espectro electromagnético.

Así, una vez el paciente se ha colocado en el interior del imán, se perturba el estado de equilibrio de los núcleos con la emisión de pulsos de RF desde una antena emisora presente en la máquina. Al emitir a la frecuencia exacta de la frecuencia del movimiento de precesión, los núcleos son capaces de absorber energía, es decir, de entrar en resonancia y así pasar del estado *up* al *down*. Cuando los núcleos del vóxel entran en resonancia comienzan a precesar a la misma frecuencia, por lo que el vector magnetización \vec{M} se desplaza de manera que su componente longitudinal va disminuyendo a la vez que la transversal va aumentando. Así, el vector \vec{M} va realizando un movimiento conocido como nutación (Figura 1.4), desviándose en un ángulo de inclinación alfa o *flip angle* de la posición de equilibrio (son comunes los pulsos de 90° y de 180° , también denominado inversor).

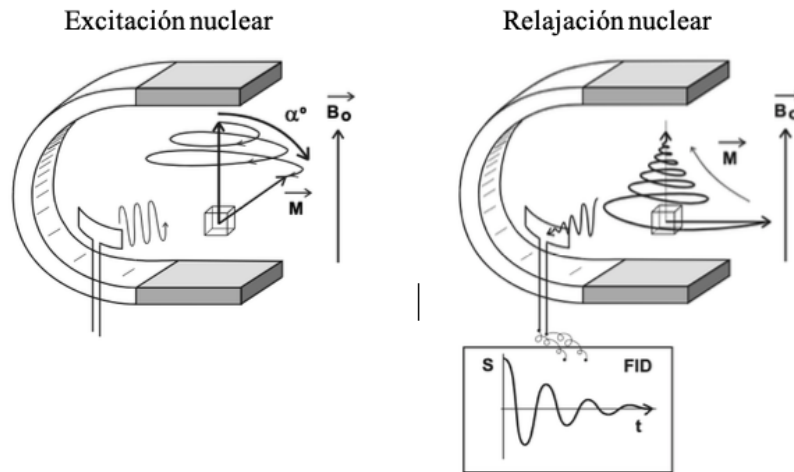


Figura 1.4: Esquema del movimiento de nutación realizado por el vector magnetización \vec{M} durante la excitación/relajación nuclear y la FID generada en ésta última. Fuente: Gili 1993.

Una vez finalizada la emisión de RF, la magnetización retorna a su posición inicial mediante un proceso de liberación energética conocido como relajación (Figura 1.4). El cambio del estado más energético al menos energético de los núcleos se produce vía la liberación del exceso energético al medio mediante la interacción dipolo-dipolo, retornando al estado de equilibrio. Esto produce unas modificaciones de campo magnético que pueden ser recogidas mediante una antena receptora en la que estas variaciones inducen una señal eléctrica conocida como FID (*Free Induction Decay*), a partir de la cual se obtiene la imagen de RM. La liberación energética se verá muy influenciada por el medio histoquímico en el que se encuentren los núcleos, por lo que la forma en que la magnetización vuelve a su posición inicial tras un pulso de RF es muy variable. Este es el origen del contraste entre tejidos en la imagen resultante.

Dos vóxeles que estén bajo campos magnéticos distintos en el momento de la relajación, tendrán frecuencias de relajación distintas y, en consecuencia, sus señales pueden ser diferenciadas mediante un análisis que discrimine por frecuencia como ocurre con el análisis de Fourier. Las señales recibidas se codifican en frecuencia en un espacio matemático denominado espacio-k.

Posteriormente, se obtiene la imagen en coordenadas espaciales mediante la aplicación de la Transformada Inversa de Fourier.

1.1.4 Potenciones en IRM

Estudiando la señal de relajación podemos obtener información sobre la densidad (D) de núcleos de H^1 en un vóxel así como información relacionada con el medio mediante los parámetros llamados $T1$, $T2$ y $T2^*$, que se obtienen estudiando la relajación nuclear sobre el eje longitudinal (M_z) y sobre el plano transversal (M_{xy}) (Figura 1.5). Las imágenes de RM reflejan en mayor o menor grado cada uno de estos parámetros pero, mediante el uso de secuencias de pulsos, podemos hacer que prevalezca uno de ellos en el contraste de la imagen, lo que equivale a potenciar la imagen en un parámetro determinado. Las secuencias de pulsos consisten en módulos básicos formados por pulsos de RF de valores concretos separados intervalos de tiempo adecuados. Estos módulos se repiten a lo largo de la obtención de la imagen con un Tiempo de Repetición (TR).

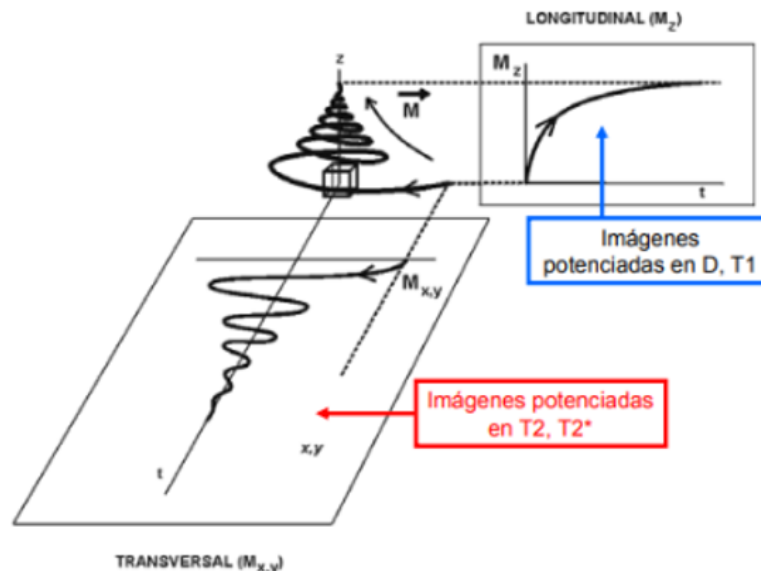


Figura 1.5: Estudio de la relajación mediante la evolución en el tiempo de las componentes longitudinal M_z y transversal M_{xy} del vector magnetización \vec{M} . Fuente: Gili 1993

Así, en la técnica de IRM existen multitud de potenciones (también llamadas tipos o modalidades de RM) en función de los parámetros de la etapa de relajación que se potencien. Las modalidades principales y más generalizadas son tres: las imágenes potenciadas en densidad protónica (D), $T1$ y $T2$ (y su similar $T2^*$). En la actualidad, existe una lista innumerable de modalidades en IRM, cada una adecuada para un propósito concreto, desde el análisis morfológico de una cierta estructura anatómica concreta o la detección de una lesión, hasta un análisis a nivel funcional de un órgano o tejido. Así, la estrategia en IRM diagnóstica consiste en programar las potenciones óptimas en las imágenes para obtener el mejor contraste entre las estructuras bajo estudio.

No entraremos en detalle acerca de las múltiples secuencias existentes y los parámetros con los que se juegan, pues queda fuera del propósito del presente trabajo. A continuación, se explican en mayor profundidad las modalidades de imagen empleadas en el presente trabajo: T1, T2 y FLAIR.

Imágenes potenciadas en T1

Las imágenes potenciadas en T1 se basan en el estudio de la relajación de los núcleos en la componente longitudinal (M_z) del vector de magnetización, conocida como relajación longitudinal, que nos da una idea de la rapidez con que se alcanza de nuevo el estado de equilibrio. Esta relajación longitudinal sigue una exponencial creciente (ecuación 1.4), como se puede observar en la Figura 1.6, regulada por una constante temporal T1, conocido como tiempo de relajación longitudinal. La intensidad de la imagen en esta modalidad se obtiene como el módulo de la componente longitudinal $M_z(T1)$ en el instante de tiempo $t=T1$, que coincide con la recuperación del 63% de la posición inicial de la componente longitudinal de la magnetización (ecuación 1.5).

$$M_z(t) = M_z(0)(1 - e^{-\frac{t}{T1}}) \quad (1.4)$$

$$I_{T1} = M_z(T1) = 0,63M_z(0) \quad (1.5)$$

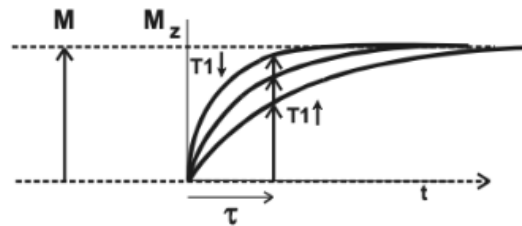


Figura 1.6: Tres curvas de relajación longitudinal distintas de tres vóxeles con la misma D. El vóxel con un T1 menor tiene una relajación más rápida. Al leer la señal tras un tiempo tau, obtenemos valores distintos que reflejan los distintos T1. Fuente: Gili 1993

Así, la intensidad de la señal es inversamente proporcional a la facilidad de relajacion. La grasa, con gran facilidad de relajacion (T1 corto), aparece hiperintensa, mientras que el agua libre, con mucha dificultad en la relajacion, (T1 largo) aparece hipointensa. La sustancia blanca tiene mayor facilidad en la relajacion que la sustancia gris, por lo que la primera aparecerá con intensidades mayores, y el líquido cefalorraquídeo (LCR) aparece muy oscuro por su largo T1 (Figura 1.8).

Para la adquisición de imágenes potenciadas en T1, se emplean en el campo de la neuroimagen distintas secuencias de pulsos como son MPRAGE (*magnetization-prepared gradient echo*) o SPGR (*spoiled gradient recalled*).

Imágenes potenciadas en T2

Por otro lado, las imágenes potenciadas en T2 se basan en el estudio de las variaciones en el tiempo de la componente sobre el plano transversal de la magnetización durante la relajación, conocida como relajación transversal (M_{xy}). Cuando M_{xy} sea cero, implica que el vector magnetización se ha alineado sobre el eje z con el campo magnético principal. La señal generada durante la relajación transversal corresponde a una senoide a la frecuencia de relajación, amortiguada por una exponencial decreciente, y la velocidad de decrecimiento depende inversamente de la constante de tiempo T2 (ecuación 1.6) (Figura 1.7). En esta modalidad la intensidad de la imagen viene dada por el módulo de la componente transversal M_z en el instante de tiempo $t = T2$ que, como se puede extraer de la ecuación 1.7, corresponde a una pérdida del 63% de su valor.

$$M_{xy}(t) = M_{xy}(0)e^{-\frac{t}{T2}} \quad (1.6)$$

$$I_{T2} = M_{xy}(T2) = 0,63M_{xy}(0) \quad (1.7)$$

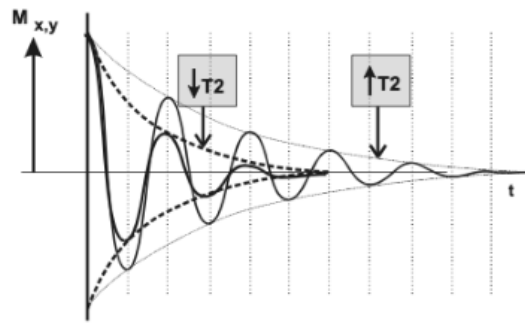


Figura 1.7: Dos señales de relajación transversal distintas de dos vóxeles con la misma D. El decrecimiento de la señal de relajación transversal es mayor cuanto menor es el T2 del vóxel. Fuente: Gili 1993

Son las heterogeneidades en el entorno bioquímico de los núcleos de H^1 las que hacen que estos perciban campos magnéticos ligeramente distintos y que liberen la energía a distintas frecuencias, originándose esta asincronía en la relajación. Así, en la modalidad T2 aquellos tejidos que presenten una mayor heterogeneidad en su composición aparecerán hiperintensos, mientras que aquellos más homogéneos aparecerán hipointensos (Figura 1.8).

Para la adquisición de imágenes potenciadas en T2, se emplean en el campo de la neuroimagen distintas secuencias de pulsos como la *dual spin echo* (DSE).

Imágenes FLAIR

Fluid-Attenuated Inversion Recovery (FLAIR), en castellano recuperación de inversión atenuada de fluido, es una secuencia de IRM que permite atenuar la señal proveniente de fluidos. Podemos observar, por ejemplo, como el LCR aparece oscuro o hipointenso (Figura 1.8). Las imágenes

obtenidas con este tipo de secuencia nos permiten distinguir si una cierta zona o lesión se trata de una hemorragia o no, al haber sido atenuadas las señales de fluidos como es la sangre. Se trata de una secuencia potenciada en T2 que utiliza la recuperación de la inversión como mecanismo para mejorar el contraste de las lesiones de la sustancia blanca en la imagen. La esclerosis múltiple es un ejemplo de enfermedad neurológica cuyo estudio se beneficia del uso de imágenes FLAIR para detectar y cuantificar lesiones en la sustancia blanca (Lladó y col. 2012).

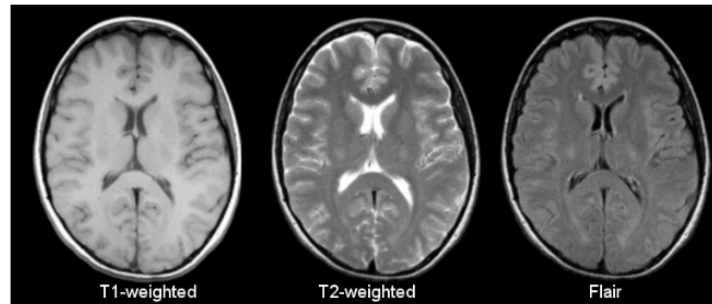


Figura 1.8: Imágenes de RM cerebral potenciadas en T1, T2 y FLAIR de un mismo sujeto. Fuente: School of Medicine, Case Western Reserve University 2006

1.1.5 *IMR en Neuroimagen*

Desde los inicios de la IRM, el campo de la neuroimagen ha sido el de mayor impacto de la técnica, por la posibilidad de visualizar tejidos blandos, como son los que componen el cerebro, de forma no invasiva, sin el uso de radiación ionizante y con una buena resolución, mayor que la que ofrecen otras técnicas de imagen médica como puede ser el TAC.

Así, las distintas modalidades de la IRM permiten no solo distinguir las distintas estructuras anatómicas del cerebro humano incluyendo la detección de lesiones, como es el caso de las modalidades T1 y T2, sino también estudiar este órgano a nivel funcional de formas muy diversas, como es el caso de las imágenes DTI (*Diffusion Tensor Imaging*) o las PWI (*Perfusion Weighted Imaging*).

Por todas estas razones, la IRM es la modalidad de imagen médica predominante para el estudio de la anatomía y función del cerebro.

1.2 Anatomía del cerebro humano

A continuación, se realiza una explicación de las principales partes que componen el órgano principal del sistema nervioso central, el cerebro, con el fin de proveer los conceptos necesarios para comprender posteriores explicaciones en el análisis de los resultados.

El cerebro es un importante y complejo órgano que controla el pensamiento, la memoria, las emociones, los sentidos, las habilidades motoras, la respiración, la temperatura, el hambre e infinidad de procesos que regulan nuestro cuerpo. Forma parte del encéfalo, que se puede dividir en tres partes principales: el cerebro como tal (o *cerebrum*), el tronco encefálico y el cerebelo (Figura 1.9a).

El cerebro es la parte de mayor tamaño y se encuentra dividido en hemisferio derecho e izquierdo. La corteza cerebral o córtex es una capa exterior de sustancia gris, recubriendo el interior de sustancia blanca. Cada uno de los hemisferios se divide por convención en cuatro lóbulos: frontal, temporal, parietal y occipital (Figura 1.9b), asociados a distintos grupos de funciones. En cada uno de los lóbulos, las áreas corticales están asociadas a funciones específicas, como son las sensoriales y motoras. Ambos hemisferios se encuentran conectados por tractos nerviosos, siendo el principal el llamado cuerpo calloso.

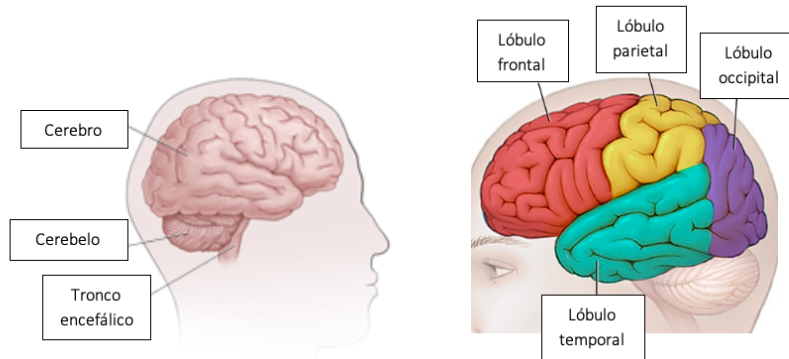


Figura 1.9: Partes mayores del encéfalo (a) y división en lóbulos del cerebro (b). Fuente: Modificado de Johns Hopkins Medicine s.f. y Mayo Clinic s.f.

El *cerebrum* está conectado a la espina dorsal a través del tronco encefálico, compuesto por el mesencéfalo, la protuberancia anular (o puente troncoencefálico) y el bulbo raquídeo. A su vez, el cerebelo está conectado con el tronco encefálico por pares de tractos. En el interior del cerebro se encuentra el sistema ventricular, consistente en cuatro ventrículos interconectados en los cuales se produce y circula el líquido cefalorraquídeo (LCR). Bajo la corteza cerebral hay varias estructuras importantes, que incluyen el tálamo, el epitálamo, la glándula pineal y la pituitaria, el hipotálamo y el subtálamo; las estructuras límbicas, incluyendo la amígdala y el hipocampo; y los varios núcleos de los ganglios basales. En la Figura 1.10 se pueden observar estas y más estructuras en detalle en el atlas anatómico de Netter.

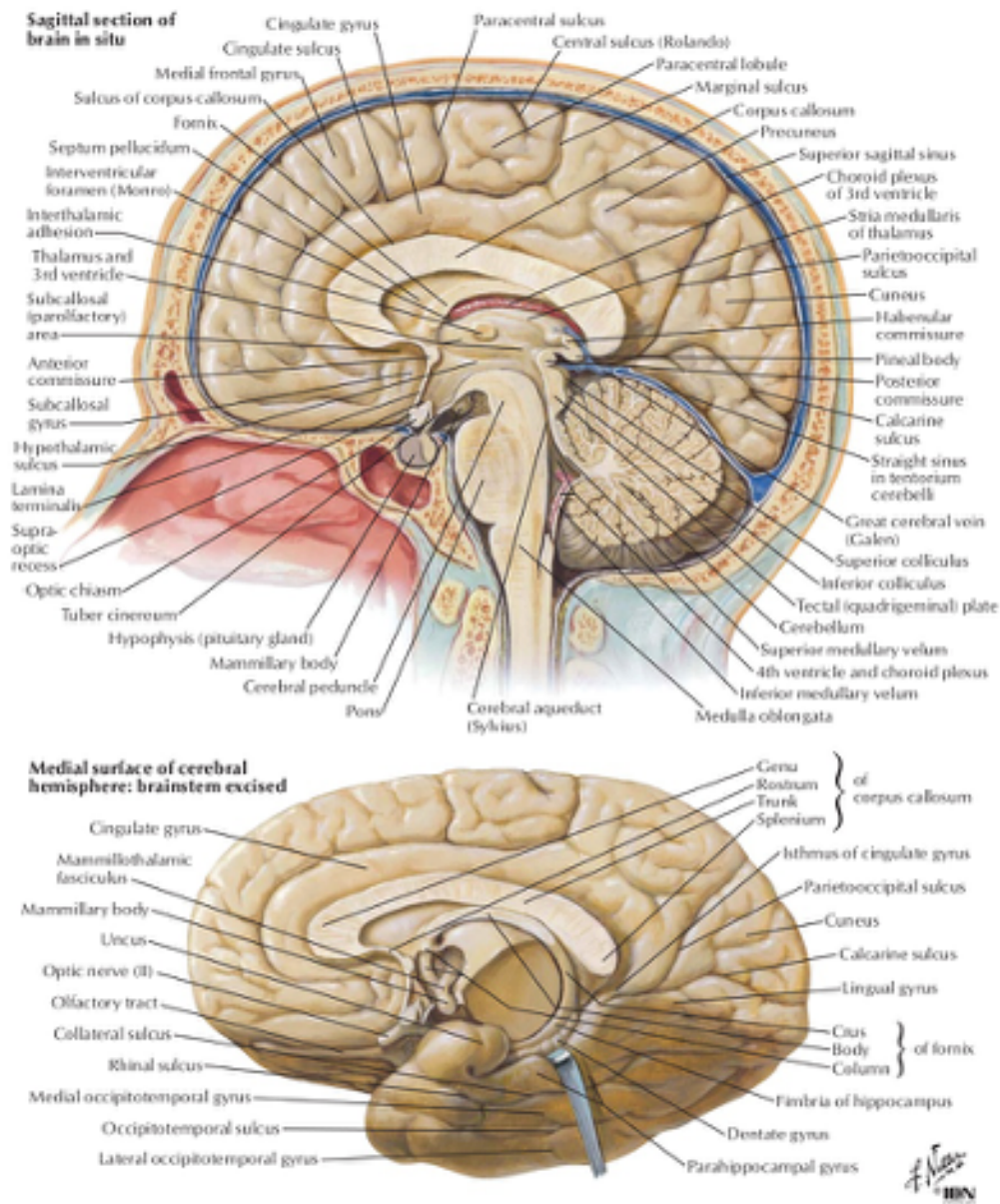


Figura 1.10: Atlas anatómico del cerebro. Fuente: Craig, Perkins y Netter 2002

El cerebro está protegido por el cráneo, suspendido en el líquido cefalorraquídeo (LCR) y aislado de la circulación sanguínea por la barrera hematoencefálica (en inglés *blood-brain barrier*). Sin embargo, el cerebro es susceptible de sufrir daños, enfermedades e infecciones. Los daños pueden ser causados por traumas o una pérdida de riego sanguíneo conocido como infarto. El cerebro es bastante susceptible de sufrir enfermedades degenerativas, como son la enfermedad de Parkinson, Alzheimer o esclerosis múltiple. El cerebro también puede dar lugar a tumores, tanto benignos como malignos. Todas estas patologías afectan en cierta manera a la anatomía y función del cerebro, por lo que el estudio vía imagen de RM suele ser una potente herramienta de diagnóstico y seguimiento.

1.3 Síntesis de Imagen

1.3.1 *Introducción a la síntesis de imagen*

La IMR ha ido ganando importancia a lo largo de los años debido a su capacidad de adquirir un rango muy diverso de contrastes de la misma anatomía subyacente de manera no invasiva. La variedad de contrastes en imágenes de RM que se pueden obtener en función de los ajustes de adquisición proporciona una mayor disponibilidad de información complementaria, que generalmente conduce a una perspectiva más amplia de cara al diagnóstico. Por ejemplo, las imágenes T1 delimitan claramente materia gris de materia blanca, mientras las imágenes T2 distinguen mejor entre tejidos fluidos y tejido cortical, y las FLAIR permiten una mejor diferenciación de las lesiones.

Desafortunadamente, no siempre es posible la adquisición de múltiples contrastes debido a limitaciones de tiempo y de comodidad del paciente, o bien algunos contrastes adquiridos pueden verse afectados por artefactos que los hacen inservibles. Aparece así la síntesis de imagen, definida como la aplicación de una transformación de intensidad a una o múltiples imágenes de origen para generar una o múltiples nuevas imágenes con un contraste determinado diferente del original (Jog, Carass y col. 2017).

La síntesis de una imagen perfectamente idéntica a su equivalente verdadera no es posible y, por esta razón, las imágenes sintéticas no se generan con el objetivo de ser utilizadas con fines diagnósticos. El objetivo es generar imágenes sintéticas que sean aproximaciones suficientemente similares a las imágenes reales para permitir o mejorar tareas relevantes en el análisis y procesamiento automático de imagen médica. Así, la capacidad de sintetizar contrastes inexistentes o corrompidos a partir de otros adquiridos exitosamente tiene un valor potencial para aumentar la utilidad de la IRM, mejorando el desempeño de algoritmos en tareas fundamentales en análisis automático de imagen médica como son el registro o la segmentación (Iglesias, Konukoglu y col. 2013). Entraremos más en detalle en esta última más adelante en un apartado dedicado al tema.

1.3.2 *Perspectiva general de métodos y estado del arte en síntesis*

La síntesis de imagen médica ha ido ganando atención en la última década y el número de métodos desarrollados para esta tarea ha ido aumentando progresivamente (Jog, Carass y col. 2017, Dar y col. 2019, Chertsias y col. 2017, Nie y col. 2018, Jog, Carass y col. 2015, Jog, Roy y col. 2013, Roy, Carass y Prince 2011, Van Nguyen, Zhou y Vemulapalli 2015, Joyce, Chertsias y Tsafaris 2017, Huang, Shao y Frangi 2017). En neuroimagen, uno de los primeros usos de la síntesis de imagen fue el uso de paquetes de software diseñados para un contraste específico de entrada diferente del disponible. En Iglesias, Konukoglu y col. 2013 los autores generaron imágenes T1 a partir de imágenes T2 para hacer posible el uso del software Freesurfer con el fin de segmentar el volumen cerebral completo.

Tradicionalmente, los métodos desarrollados para la generación de imágenes sintéticas se pueden dividir en dos familias en función de cómo abordan el problema: los métodos basados en registro (Miller y col. 1993, Burgos y col. 2014) y los métodos basados en transformaciones de intensidad (Jog, Carass y col. 2017, Dar y col. 2019).

El primer grupo tiene su origen en el trabajo de Miller y col. 1993, en el cual la síntesis de imagen se llevó a cabo alineando en el espacio, es decir, registrando una imagen de un sujeto y una imagen formada a partir de un conjunto de N imágenes de múltiples sujetos corregistradas, lo que se conoce como atlas. Así, dada una imagen de un sujeto $B1$ de una modalidad $M1$ y dos atlas $A1$ y $A2$ de imágenes de N sujetos corregistradas de las modalidades $M1$ y $M2$, $A1$ se registra a $B1$ y se aplica esa misma transformación a $A2$ para generar la modalidad $M2$. Este tipo de métodos asumen que las imágenes de distintos sujetos están relacionadas entre sí a través de una transformación geométrica. Así, el uso del registro para la síntesis de imagen puede llevar a errores significativos en las regiones más finas del cerebro, como es la corteza cerebral, donde existe una notable variabilidad inter-sujeto y el registro no siempre es preciso. Además, la síntesis basada en registro es válida únicamente bajo hipótesis de normalidad, es decir, falla en presencia de anomalías anatómicas en los tejidos como puede ser un tumor, pues los atlas no presentan dichas anomalías (Jog, Carass y col. 2017).

La alternativa es el uso de métodos basados en intensidad que no se basan en una relación estrictamente geométrica de los sujetos. A su vez, dentro de este amplio grupo podemos encontrar métodos basados en técnicas distintas. Un subgrupo son los métodos de síntesis basados en la representación *sparse* de las imágenes a partir de la combinación lineal de bases de un diccionario aprendido (Roy, Carass y Prince 2011, Huang, Shao y Frangi 2017), técnica altamente extendida para aplicaciones muy diversas en campos como el procesamiento de señales y de imágenes o *Machine Learning* (ML). Sin embargo, estos métodos mostraron limitaciones en los resultados y requerían un tiempo de computación relativamente alto, dando así paso a los métodos más recientes basados en mapeos no lineales, con técnicas de regresión no lineal (Jog, Carass y col. 2017, Jog, Carass y col. 2015, Jog, Roy y col. 2013) introduciéndose más recientemente las redes neuronales profundas (*Deep Learning* (DL)) (Dar y col. 2019, Nie y col. 2018]. Estos aprenden mapeos no lineales de intensidad entre pares de modalidades o tipos de imagen.

En Jog, Carass y col. 2017 los autores proponen REPLICA, un método basado en *Random Forest* (Breiman 2001) por *patches*. El contraste sintetizado depende del promediado de las salidas predichas de árboles de *Random Forest* independientes, lo que puede producir una pérdida de detalle de alta frecuencia. Más recientemente, se han propuesto métodos basados en *Deep Learning* (explicado más adelante en mayor profundidad en la sección 1.5 dedicada al tema) para llevar a cabo esta tarea, mostrando resultados prometedores debido a la habilidad de aprender características estructurales automáticamente. La mayoría de estos métodos son basados en cortes o *slice-based*, en los cuales cada corte del volumen es sintetizado de forma independiente (o con apoyo 3D limitado) lo que puede dar lugar a inconsistencias entre cortes vecinos. Además, el uso de un contexto espacial limitado (2D) puede resultar en resultados subóptimos debido a la falta de una perspectiva holística del contexto.

1.4 Segmentación

1.4.1 Introducción a la segmentación

La segmentación de imágenes es una de las tareas más importantes en el campo del análisis de imagen médica y, frecuentemente, es el primer y más crítico paso en múltiples aplicaciones clínicas. Concretamente, en análisis de IRM cerebral, la segmentación se emplea comúnmente para medir y visualizar las estructuras anatómicas del cerebro, para analizar cambios cerebrales, para delinear regiones patológicas y para la planificación quirúrgica e intervenciones guiadas por imagen. El análisis cuantitativo de la IRM cerebral es esencial para el diagnóstico y seguimiento de múltiples enfermedades y afecciones neurológicas, y se basa principalmente en una segmentación precisa de las estructuras anatómicas de interés.

La segmentación se refiere al proceso de etiquetado de los píxeles o vóxeles de una imagen con etiquetas biológicamente significativas, como estructuras anatómicas y tipos de tejidos. El resultado de la segmentación es una imagen de etiquetas que identifican cada región homogénea (Figura 1.11) o un conjunto de contornos que describen los límites de las regiones. En el caso del cerebro o encéfalo, los elementos de la imagen se clasifican generalmente en tres tipos principales de tejido: materia blanca (WM), materia gris (GM) y líquido cefalorraquídeo (LCR, CSF por sus siglas en inglés), aunque también se suelen realizar distinciones más precisas dependiendo de a qué estructura cerebral pertenece ese tejido.

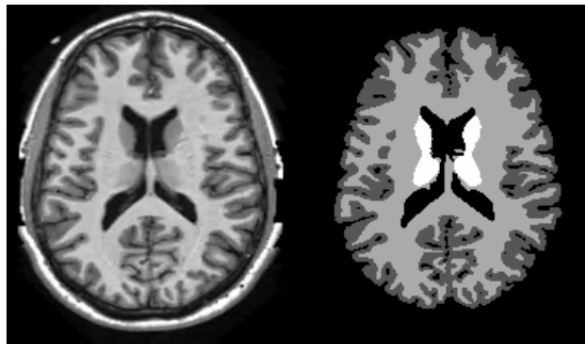


Figura 1.11: Imagen de RM cerebral T1 (izquierda) y su correspondiente segmentación con una red UNet (derecha).

1.4.2 Perspectiva general de métodos y estado del arte en segmentación

La segmentación manual resulta una tarea laboriosa y lenta, propensa a errores, difícil de reproducir, costosa y no escalable (Iglesias y Sabuncu 2015). Además, la calidad de los resultados dependerá del desempeño del experto. Por lo tanto, la segmentación manual generalmente no es adecuada para llevarse a cabo en conjuntos de datos a gran escala o en aplicaciones donde el tiempo es crítico, como la planificación del tratamiento. Los algoritmos de segmentación automáticos o semiautomáticos pueden abordar estos desafíos, acelerando el proceso, reduciendo el costo y ofreciendo fiabilidad, reproducibilidad y escalabilidad. En las últimas décadas, se han desarrollado y publicado en la literatura diversas técnicas de segmentación de distinta precisión

y grado de complejidad (Iglesias y Sabuncu 2015, Ronneberger, Fischer y Brox 2015, Dou y col. 2017, Coupé y col. 2011).

Los primeros y más sencillos métodos automáticos de segmentación en IRM cerebral eran basados únicamente en intensidad. Sin embargo, la clasificación basada en la intensidad de los píxeles resulta insuficiente para romper la ambigüedad entre estructuras de intensidad similar o con distribuciones de intensidad con solapamiento, incluso en la distinción más simple de los 3 tejidos cerebrales principales (Figura 1.12).

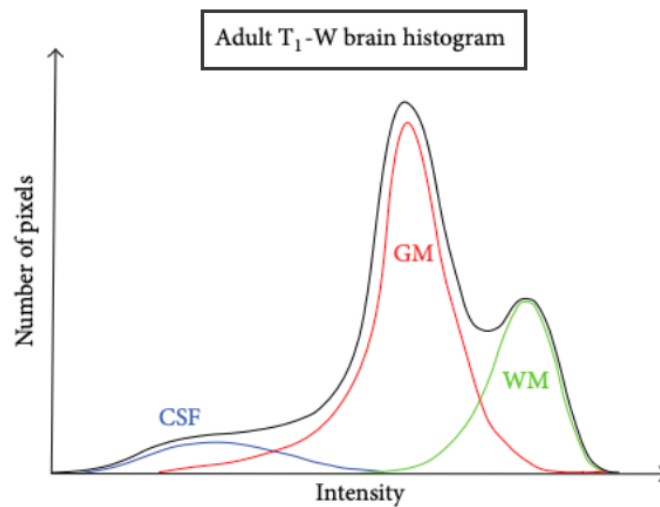


Figura 1.12: Histograma de una imagen T₁ de un cerebro adulto. Los picos de los 3 tejidos se basan en una segmentación manual y las distribuciones difieren ligeramente de la distribución gaussiana debido al efecto de volumen parcial. Fuente: Despotović, Goossens y Philips 2015

Con tal de romper esta ambigüedad en intensidad, aparecen los métodos basados en atlas incorporando información espacial a priori. Un atlas es una imagen etiquetada que nos sirve a modo de mapa para localizar espacialmente las estructuras en la imagen. Así se desarrollaron múltiples métodos basados en atlas probabilísticos o en *single* atlas (registro atlas-caso seguido de transferencia de etiquetas). Los errores de registro y la variabilidad biológica inter-sujeto imponen limitaciones en la precisión de este tipo de técnicas, surgiendo la estrategia multi-atlas para mejorarlas.

Así, dentro de las técnicas del estado del arte podemos destacar los métodos de segmentación multi-atlas (Coupé y col. 2011) y los basados en *Deep Learning* (Dou y col. 2017). El primer grupo requiere de la generación de un conjunto de atlas (imágenes de ‘entrenamiento’) previamente etiquetadas manualmente. Un método multi-atlas estándar se basa en registros por pares calculados de forma independiente entre atlas individuales y la imagen objetivo a segmentar. Estos registros se utilizan luego para propagar las etiquetas del atlas al espacio objetivo y fusionarlas en una única segmentación final. Las técnicas de fusión de etiquetas empleadas se basan generalmente en la similitud entre los valores de intensidad de los atlas y la imagen a segmentar.

El registro suele ser una tarea computacionalmente costosa que requiere tiempos muy largos. Así surgen los métodos basados en *Deep Learning*, computacionalmente más eficientes y con resultados prometedores. Los enfoques de segmentación basados en aprendizaje profundo para la resonancia magnética cerebral están ganando interés debido a su capacidad de autoaprendizaje y generalización sobre grandes cantidades de datos. A medida que las arquitecturas de aprendizaje profundo se vuelven más maduras, van superando gradualmente a los algoritmos de aprendizaje automático clásicos de última generación.

Sin embargo, la segmentación automática de las estructuras cerebrales y la detección de las anomalías siguen siendo un problema sin resolver debido a las variaciones anatómicas normales en la morfología del cerebro, las variaciones en la configuración de adquisición y los escáneres de resonancia magnética, las imperfecciones de adquisición de las imágenes y las variaciones patológicas.

1.5 *Deep Learning* en Redes Neuronales Artificiales

En el presente apartado se pretende proporcionar una explicación general de las redes neuronales artificiales (RNA) y el aprendizaje profundo (AP) o *Deep Learning* (DL), para a continuación entrar en mayor profundidad acerca de las redes neuronales convolucionales (CNN) y, en particular, los autoencoders convolucionales (CAE).

1.5.1 *Introducción al Deep Learning*

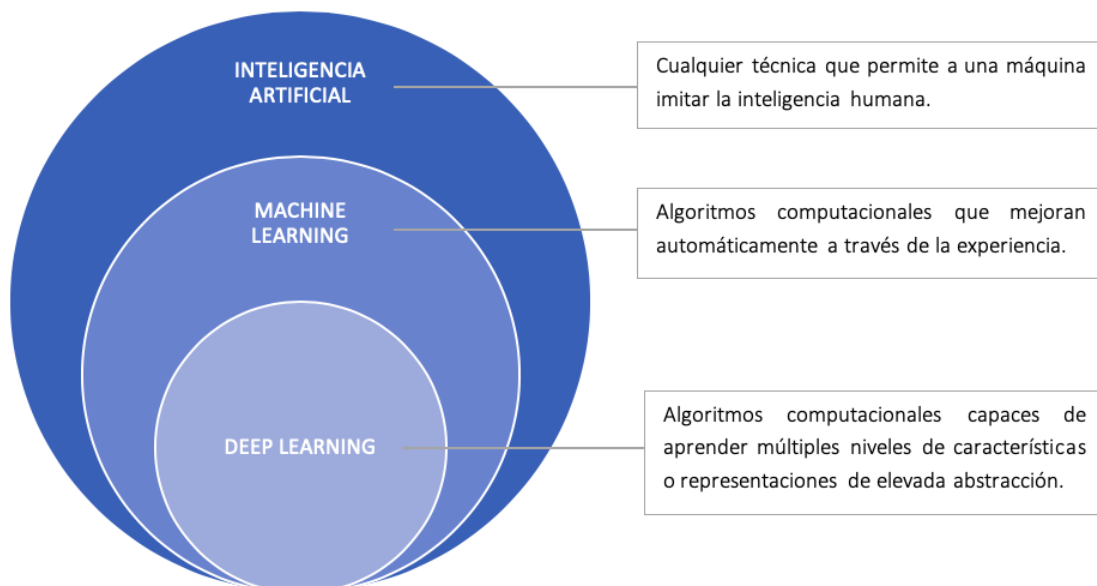


Figura 1.13: Esquema de la relación entre inteligencia artificial, aprendizaje automático y aprendizaje profundo y sus respectivas definiciones.

El aprendizaje profundo (AP) o *Deep Learning* (DL) se enmarca en el campo del aprendizaje automático (AA) o Machine Learning (ML), que a su vez se considera un subgrupo de la inteligencia artificial (AI). En la Figura 1.13 se ilustran estas relaciones y se proporciona una definición de cada uno de estos conceptos tan ampliamente extendidos en la actualidad que tienden fácilmente a confusión.

Las técnicas convencionales de aprendizaje automático presentaban una capacidad limitada para procesar datos en su forma natural o *raw data*. Durante décadas, la construcción de un sistema de reconocimiento de patrones o AA requería una ingeniería minuciosa y una experiencia considerable en el dominio para diseñar un extractor de características que transformara los datos originales sin procesar (como los valores de píxeles en una imagen) en una representación interna adecuada o un vector de características a partir del cual el subsistema, a menudo un clasificador, podría detectar o clasificar patrones en esa entrada.

El aprendizaje de características es un conjunto de métodos que permite que una máquina sea alimentada con datos brutos y descubra o aprenda automáticamente las representaciones necesarias para la detección o clasificación. Los métodos de AP son métodos de aprendizaje de características con múltiples niveles de representación, obtenidos por composición de módulos simples pero no lineales que transforman la representación en un nivel (comenzando con la entrada sin procesar o *raw data*) en una representación más abstracta en un nivel superior. Con la composición de suficientes transformaciones de este tipo, es posible el aprendizaje de funciones de alta complejidad. El aspecto clave del aprendizaje profundo es que estas capas de extracción de características no están diseñadas por ingenieros, sino que se aprenden a partir de los datos mediante un procedimiento de aprendizaje de propósito general. (LeCun, Bengio e Hinton 2015)

Las técnicas de aprendizaje automático se dividen tradicionalmente en dos amplias familias:

- **Aprendizaje supervisado:** los algoritmos aprenden en base a ejemplos dato-solución. Es el caso de las técnicas de DL.
- **Aprendizaje no supervisado:** los algoritmos aprenden en base a los datos sin “etiquetar” (sin la solución). Aprenden patrones y representaciones intrínsecas de los propios datos.

A continuación se explican las redes neuronales artificiales (RNA), enmarcadas en las técnicas de DL.

1.5.2 Fundamentos de las RNA

La Neurona Artificial: del modelo biológico al matemático

Las redes neuronales artificiales (RNA) son sistemas computacionales inspirados en las redes neuronales biológicas que constituyen el sistema nervioso de los animales.

Se basan en una colección de unidades fundamentales conectadas denominadas neuronas artificiales o perceptrones, que modelan en cierto nivel las neuronas biológicas. Cada conexión, tal y como ocurre con las sinapsis en un cerebro biológico, puede transmitir una “señal” a otras neu-

ronas. Una neurona artificial recibe una “señal”, la procesa y la puede transmitir a las neuronas conectadas a ella. Esa “señal” en cada conexión es un valor real, y la salida de cada neurona se calcula como función no lineal de sus entradas. En cada conexión o enlace el valor de salida de la neurona anterior se multiplica por un valor de peso. Estos pesos en los enlaces pueden incrementar o inhibir el estado de activación de las neuronas adyacentes. Asimismo, a la salida de la neurona existe una función que modifica el valor resultante o impone un límite que no se debe sobrepasar antes de propagarse a otra neurona (función de activación). Esta analogía queda ilustrada en la Figura 1.14 junto con la Tabla 1.1.

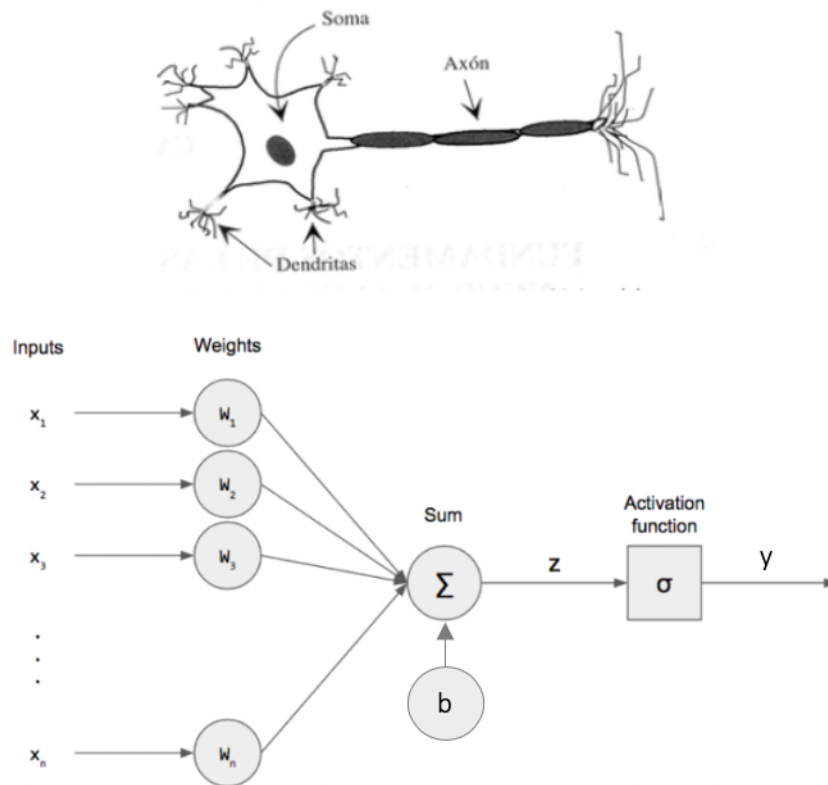


Figura 1.14: Esquema de la analogía entre una neurona biológica (arriba) y una neurona artificial (abajo). Fuente: *Las redes neuronales* 2006 y Deshpande 2017

RNA	Sistema Nervioso
Entrada	Dendrita
Función de activación	Soma
Salida	Axón
Interconexiones neuronales	Sinapsis
Pesos de las interconexiones	Fuerzas sinápticas

Tabla 1.1: Analogía de las partes de una neurona artificial con una biológica.

Así, una vez entendida la analogía biológica-artificial, el modelo matemático general que siguen las neuronas (acorde a la notación en la Figura 1.14) se puede expresar como en la ecuación 1.8.

$$y_j = f \left(\sum_{i=1}^N \omega_{ij} y_i + b_j \right) \quad (1.8)$$

Donde y_j es el valor de salida de la neurona j , y_i son las salidas de las neuronas i previas y conectadas a la neurona j , ω_{ij} es el peso correspondiente a la conexión de las neuronas i y j , b_j es el valor del *bias* asociado a la neurona j y f es la función de activación de la neurona j .

Esta función de activación resulta de gran importancia, pues es la que otorga al modelo la no-linealidad. Existen múltiples posibilidades en cuanto a la elección de la función de activación para una neurona en función de la salida deseada para la tarea concreta a realizar. En la Tabla 1.2 podemos observar algunas de las más comúnmente empleadas con sus expresiones matemáticas, su representación gráfica y sus rangos de valores correspondientes. La ReLU (*Rectified Linear Unit*) (Agarap 2018) es la función de activación de uso más extendido actualmente, empleada en la mayoría de redes neuronales convolucionales. Esto se debe a que hace las redes profundas más rápidas ayudando a evitar el problema del desvanecimiento de gradiente o *gradient vanishing* (Hochreiter y col. 2001).


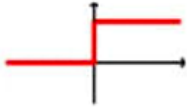

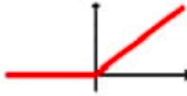
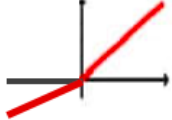
Nombre	Función	Rango	Representación gráfica
Lineal	$f(x) = x$	$[-\infty, +\infty]$	
Escalón binario	$f(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases}$	$[0, 1]$	
Sigmoide (logística)	$f(x) = \frac{1}{1 + e^{-x}}$	$[0, +\infty]$	
ReLU	$f(x) = \max(0, x)$	$[0, +\infty]$	
Leaky ReLU	$f(x) = \begin{cases} \alpha x & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases}$	$[-\infty, +\infty]$	

Tabla 1.2: Ejemplos de funciones de activación típicas para las neuronas artificiales.

Arquitectura de una RNA: de la neurona a la red

Típicamente, las neuronas se organizan formando capas. El término arquitectura de una RNA se refiere a la descripción del número de neuronas en cada capa, el número de capas en la red, la función de activación de cada capa y las conexiones entre capas. Así, en función de todos estos parámetros, existen innumerables arquitecturas posibles para una RNA, en función del problema a resolver (clasificación, regresión, etc.).

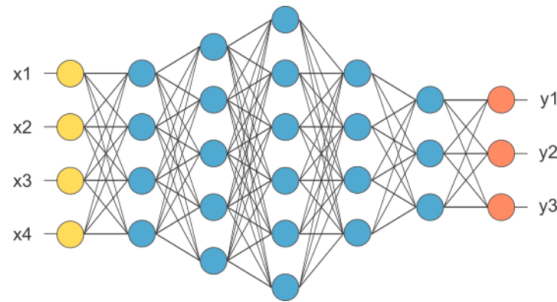


Figura 1.15: Esquema de la organización de neuronas en capas interconectadas para formar una red. Fuente: OpenNN, Open Neural Networks s.f.

Tal y como observamos en la Figura 1.15, podemos distinguir tres tipos de capas en una red:

- La capa de entrada: aquella capa de neuronas que reciben como entrada los propios datos iniciales, por lo que no presentan conexiones previas a otras neuronas.
- La capa de salida: aquella capa final de neuronas que tienen como salida el resultado final de la red. Las redes neuronales profundas (*Deep Learning*) se tratan de redes con múltiples capas ocultas.
- Las capas ocultas: aquellas capas intermedias entre la capa de entrada y la de salida, que presentan conexiones en ambos sentidos (entrada y salida), con las capas adyacentes.

Aprendizaje de una RNA

El aprendizaje de una red se consigue modificando los pesos y *bias* de la red de forma iterativa para conseguir las salidas deseadas, y consiste en que la red aprende a realizar ciertas funciones mediante un entrenamiento con ejemplos (pares dato-solución).

Típicamente, la base de datos a emplear se divide en un conjunto de entrenamiento, uno de validación y uno de test para ser empleados en diferentes fases. Durante el entrenamiento de la red, el modelo se ajusta inicialmente a al conjunto de datos de entrenamiento, empleados para ajustar los pesos. El modelo ajustado se emplea sucesivamente para predecir los resultados de los datos de validación, que proporciona una evaluación imparcial del desempeño de la red. Esto sirve para “guiar” el entrenamiento de alguna forma. El objetivo es ajustar la red de manera que ésta tenga el mejor rendimiento con datos nuevos, evitando el sobreajuste u *overfitting*. Por último, el conjunto de test es empleado tras el entrenamiento para evaluar la red ya ajustada

sobre datos nuevos nunca vistos por ésta.

Una vez realizada esta partición de los datos, comienza el proceso de entrenamiento de la red como se explica a continuación:

I. Propagación *forward* o hacia delante:

El entrenamiento comienza pasando los datos del conjunto de entrenamiento secuencialmente como entrada a las neuronas de la capa de entrada, habiendo inicializado previamente los pesos de forma aleatoria a un valor cercano a cero. Estas entradas pasan a través de las capas ocultas de la red, que van calculando sus respectivas salidas, hasta llegar a la capa final que proporciona el valor de salida resultante. Cada neurona pasará su valor únicamente si la suma de pesos supera un umbral. Este proceso recibe el nombre de propagación *forward* o hacia delante. Una vez finalizado, se compara el resultado obtenido con los valores reales o *ground-truth*, es decir, el resultado deseado. Para ello, se emplea una función de coste o *loss function* con el fin de comprobar cuánto error está cometiendo la red. Así, el problema de aprendizaje o entrenamiento de una red es equivalente al problema de minimizar la función de coste.

II. Retropropagación del error o *backpropagation*:

A continuación, se lleva a cabo la retropropagación del error o *backpropagation* para reajustar los pesos en función del error calculado. Existe diversidad de algoritmos que optimizan funciones. Estos algoritmos suelen ser basados en el gradiente, en el sentido de que no utilizan únicamente la información proporcionada por la función de coste, sino también por su gradiente.

La técnica más comúnmente empleada es la de descenso del gradiente o *Gradient Descent*. Estos algoritmos se basan en el cálculo de la derivada de la función de coste C (gradiente ∇C) que representa el error total cometido por la red con respecto a cada peso ω_j^i de la red en cada iteración i , y la aplicación de una modificación de los pesos ($\Delta\omega$) en la dirección opuesta a éste, de manera que se garantiza la disminución de la función de coste ($\Delta C < 0$), tal y como se muestra en las siguientes ecuaciones:

Modificación de los pesos:

$$\Delta\omega = -\alpha \cdot \nabla C \quad (1.9)$$

$$\Delta\omega_j^{i+1} = \omega_j^i + \Delta\omega_j^i = \omega_j^i - \alpha \cdot \nabla C \quad (1.10)$$

Consecuente modificación del error total:

$$\Delta C = \frac{\partial C}{\partial \omega} \cdot \Delta\omega = -\alpha \cdot \nabla C \cdot \nabla C = -\alpha \|\nabla C\|^2 \quad (1.11)$$

$$\|\nabla C\|^2 > 0 \rightarrow \Delta C < 0 \quad (1.12)$$

El factor α es conocido como *learning rate* y es el encargado de modular la magnitud con que se modifican los pesos, es decir, lo grande que es el “paso” en cada iteración. Se trata de un importante hiperparámetro, pues un valor demasiado pequeño implica un aprendizaje muy lento de la red y un entrenamiento demasiado largo pudiendo ésta quedar atascada, mientras que un valor demasiado alto podría resultar en un ajuste subóptimo de la red o un proceso de entrenamiento inestable. (Brownlee 2020)

Se denomina retropropagación o *backpropagation* porque el algoritmo calcula en primer lugar las modificaciones a aplicar sobre los pesos de las neuronas de la capa de salida y, a continuación, aplica la regla de la cadena de la derivación para calcular las modificaciones de los pesos de las capas inmediatamente anteriores de forma sucesiva, hasta acabar en la capa de entrada. Así, el algoritmo va “navegando” por la superficie de error en dirección opuesta al gradiente, minimizando en cada iteración la función de coste, tal y como se puede observar en la Figura 1.16.

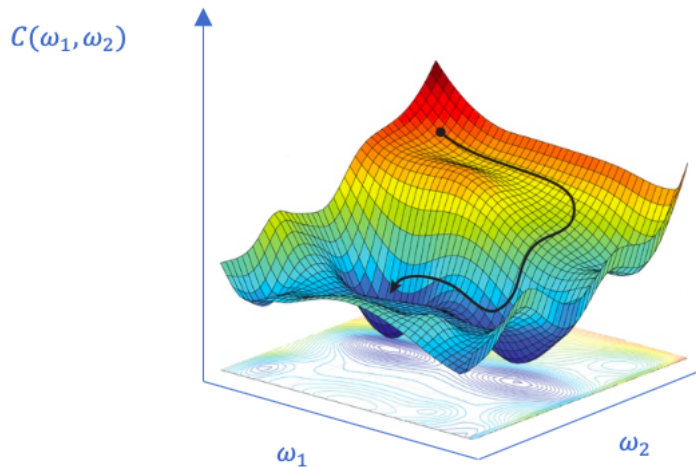


Figura 1.16: Esquema ilustrativo de la navegación por la superficie de error o función de coste con la técnica *gradient descent*. C es la función de coste, ω_1 y ω_2 son los parámetros de la red. Fuente: Modificado de Hutson 2018

En la actualidad son ampliamente empleados para el entrenamiento de redes neuronales profundas algoritmos de optimización por lotes o *batches*. Dos ejemplos destacables son el método *Stochastic Gradient Descent* (SGD) (Bottou 2010), en castellano descenso estocástico del gradiente, o el más reciente Adam (Kingma y Ba 2014), basado en el primero pero con un *learning rate* adaptativo y más eficiente computacionalmente. Frecuentemente en la práctica, el cálculo de la función de coste y el gradiente para el conjunto de entrenamiento completo puede resultar un proceso extremadamente lento e incluso intratable para conjuntos de datos de gran dimensión. Es por ello que se emplean esta clase de métodos de optimización que seleccionan de forma aleatoria un subconjunto de los datos de entrenamiento o *batch* para realizar los cálculos en cada iteración.

III. Este proceso se repite iterativamente hasta que se minimiza el error (la función de coste) y la red ha “aprendido” los datos ajustando sus pesos a estos. Estas iteraciones se denominan *epochs*. En cada *epoch* la red ajusta los pesos a un *batch* de datos y al finalizar, se predicen las soluciones para el conjunto de datos de validación, desconocidos para la red. Si la función de coste o error de la red disminuye en el conjunto de validación con respecto a la *epoch* anterior, los pesos actuales de la red son guardados y si, por el contrario, es mayor, se mantienen los pesos calculados en la anterior.

En cuanto a la función de coste, existe una gran variedad y es totalmente personalizable para el problema concreto a resolver por la red. En la Tabla 1.3 se exponen algunos ejemplos de funciones de coste comunes junto con algunos casos en los cuales es adecuada su utilización.

<i>Nombre</i>	<i>Ecuación</i>	<i>Aplicación</i>
<i>Mean Absolute Error (MAE)</i> (Norma L1)	$MAE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n y_i - \hat{y}_i $	<i>Regresión</i>
<i>Mean Squared Error (MSE)</i> (Norma L2)	$MSE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$	<i>Regresión</i>
<i>Binary Cross-Entropy (BCE)</i>	$BCE(y, \hat{y}) = (y)(-\log(\hat{y})) + (1 - y)(-\log(1 - \hat{y}))$	<i>Clasificación binaria</i>
<i>Divergencia de Kullback-Leibler</i>	$KLD(y, \hat{y}) = - \sum_{i=1}^n y^{(i)} \cdot \log \left(\frac{y^{(i)}}{\hat{y}^{(i)}} \right)$	<i>Comparación de distribuciones de probabilidad</i>

Tabla 1.3: Ejemplos de funciones de coste típicas para el entrenamiento de una RNA.

Evaluación de una RNA

Una vez finalizado el entrenamiento de la red, se procede a su evaluación. En el ámbito médico, la toma de decisiones clínicas es un proceso extremadamente complejo. En este contexto, es imprescindible conocer detalladamente la exactitud de los métodos.

Para la evaluación, se ha reservado previamente el conjunto de datos de test, desconocidos para la red, que se procesa con la red entrenada para obtener los resultados predichos por ésta según los pesos aprendidos. Una vez obtenidos los resultados, se debe cuantificar de alguna manera el desempeño de la red en su tarea concreta. Para ello, existen métricas de evaluación muy diversas en función de dicha tarea a realizar por la red. A continuación se mencionan algunas de ellas, sin

entrar en una mayor explicación por no ser estrictamente necesario para el presente trabajo.

Para modelos de clasificación se emplean métricas tales como:

- Precisión o *accuracy*.
- Sensibilidad y especificidad.
- Análisis de la curva ROC (*Receiver Operating Curve*).
- *F1-Score*.

Por otra parte, para los modelos de regresión, se emplean métricas diferentes, ya explicadas como función de coste, tales como:

- *Mean squared error* (MSE).
- *Root mean squared error* (RMSE).
- *Mean absolute error* (MAE).

Para el caso concreto en regresión de evaluación de la similitud entre dos imágenes, se emplean métricas tales como el PSNR (*Peak Signal-to-Noise Ratio*), el SSIM (*Structural Similarity Index Measure*) o el coeficiente de correlación de Pearson (CC), que se explican más adelante en la sección ?? *Evaluación de la red*.

1.5.3 Redes Neuronales Convolucionales (CNN)

Una red neuronal convolutiva (CNN) es un tipo de red neuronal empleada en visión artificial en la que las neuronas corresponden a campos receptivos de una manera muy similar a las neuronas en la corteza visual primaria (V1). Así, las CNN están diseñadas para procesar datos en forma de múltiples matrices. Muchas modalidades de datos se encuentran en esta forma: 1D para señales y secuencias, como es un electroencefalograma (ECG) o el lenguaje; 2D para imágenes o espectrogramas de audio; y 3D para video o imágenes volumétricas. Una aplicación típica es la clasificación de imágenes, campo en el cual han conseguido posicionarse como la técnica con mejores resultados. A continuación, la explicación se centra en su aplicación a imágenes.

Las CNN se basan en el uso de *kernels* o filtros para la extracción de información o características de las imágenes mediante operaciones de convolución. Un *kernel* no es más que una matriz de pequeñas dimensiones. La convolución es una operación matemática que, dadas dos señales H y F (en el caso de las CNN la imagen y el *kernel*), devuelve un valor proporcional a la cantidad de información compartida entre ambas señales. La operación de convolución se define matemáticamente como en 1.13, y en la Figura 1.17 se trata de ilustrar su funcionamiento.

Convolución 2D:

$$G[i, j] = \sum_{u=-k}^k \sum_{v=-k}^k H[u, v] F[i - u, j - v] \quad (1.13)$$

Notación:

$$G = H \otimes F \quad (1.14)$$

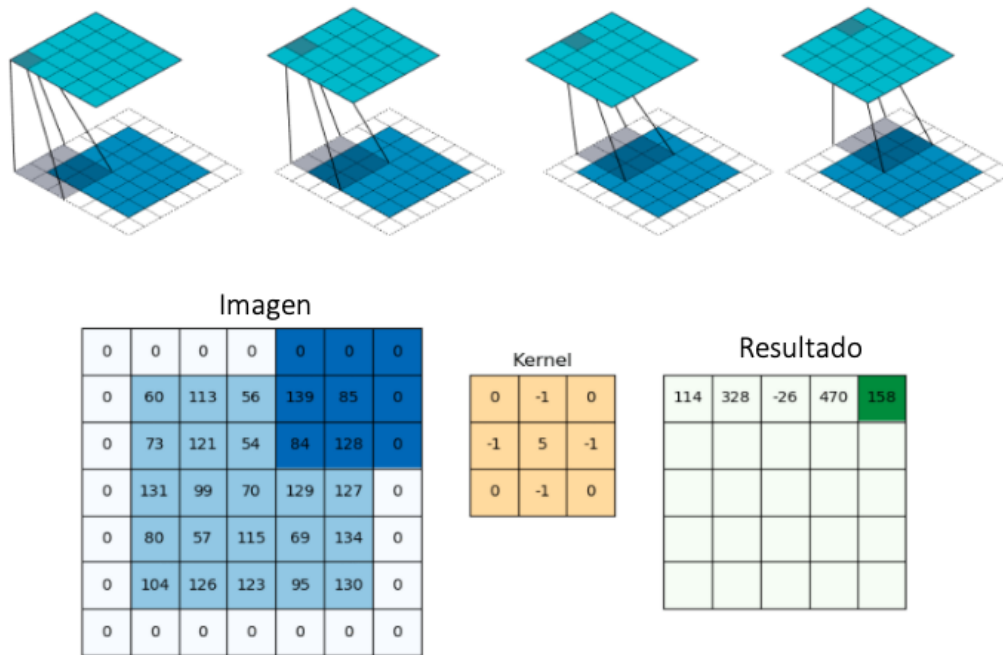


Figura 1.17: Esquema ilustrativo de la operación de convolución 2D entre una imagen de 5x5 y un kernel de 3x3, con *zero padding* o relleno con ceros. Fuente: Modificado de *TensorFlow Conv2D Layers: A Practical Guide* s.f. y Hussain y col. 2018

La arquitectura de una CNN típica (Figura 1.19) está estructurada en una serie de etapas. Las primeras etapas se componen de dos tipos de capas:

- **Capas convolucionales:** en una capa convolucional j se realizan las operaciones de convolución sobre las imágenes (en el caso de la capa inicial) o los mapas de características o activación Y_i de la capa anterior i (en el caso de las capas intermedias) con los filtros K_{ij} (parámetros entrenables). El resultado de la convolución se denomina mapa de características o de activación (*feature/activation maps*). Las unidades en una capa convolucional se organizan en mapas de características o de activación, dentro de los cuales cada unidad está conectada a parches o *patches* locales en los mapas de características de la capa anterior a través de un conjunto de pesos llamados *kernels* o filtros. El resultado de la suma ponderada local pasa luego a través de una función de activación no lineal como la ReLU.
- **Capas de *pooling*:** a continuación de las capas convolucionales, sirven a modo de submuestreo o *downsampling*. Los valores de los píxeles de una determinada zona o vecindario se agregan empleando una función invariante de permutación, generalmente el valor máximo (*max pooling*) o promedio (*average pooling*) (Figura 1.18).

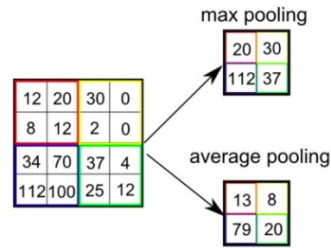


Figura 1.18: Esquema ilustrativo de las operaciones de *max pooling* y *average pooling*. Fuente: Saha 2018

Finalmente, tras varias capas de convolución y *pooling*, se incluyen capas completamente conectadas o *fully connected layers*, en las cuales todas las neuronas de la capa están conectadas a todas las activaciones de la siguiente. Son éstas las que realizan el razonamiento de alto nivel a partir de las características o patrones extraídos por las capas anteriores.

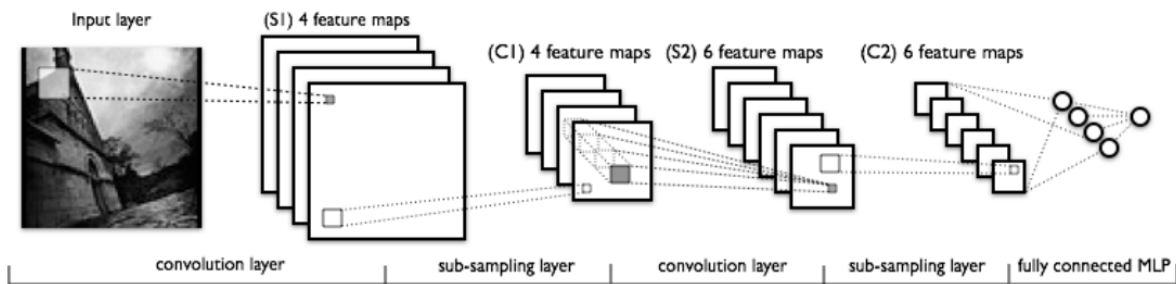


Figura 1.19: Esquema de ejemplo de arquitectura de una CNN. Fuente: *Convolutional Neural Networks (LeNet)* 2018

El modelo matemático general para una neurona artificial anteriormente expresado en 1.8 queda como en 1.15 para una CNN, en el cual se ha sustituido la operación de multiplicación por la de convolución, y los pesos a aprender son en este caso los *kernels*.

$$Y_j = g \left(b_j + \sum_{i=1} K_{ij} y_i \otimes Y_i \right) \quad (1.15)$$

Donde Y_j son los mapas de activación resultantes de la capa j , Y_i son los mapas de activación de la capa anterior i , K_{ij} son los filtros o *kernels* correspondientes a la conexión entre las capas i y j , b_j es el valor de *bias* asociado a la capa j y g es la función de activación.

1.5.4 Autoencoders Convolucionales (CAE)

Los autoencoders convolucionales (CAE) son un tipo de CNN que aprende a codificar la entrada en una representación comprimida para después tratar de reconstruir la entrada a partir de ésta. Su funcionamiento general puede expresarse por tanto como $decodificar(codificar(x))$.

Un conocido ejemplo es la U-Net (Ronneberger, Fischer y Brox 2015), una especie de CAE con conexiones adicionales en cada nivel de resolución, cuya arquitectura queda ilustrada en la Figura 1.20. Consiste en un camino codificador, de contracción o descendente (parte izquierda) y un camino decodificador, expansivo o ascendente (parte derecha) que son simétricos. Así, estas redes quedan organizadas en niveles análogos en ambas etapas e interconectados.

- **Camino codificador:** sigue la arquitectura típica de una CNN para codificar la información de la imagen original en un espacio latente. Consiste en la aplicación repetida de convolución + ReLU + *pooling* para *downsampling*. Así, se van reduciendo las dimensiones de la entrada en cada nivel, hasta obtener una representación comprimida de la entrada de dimensiones menores.
- **Camino decodificador:** en este caso aparece una modificación con respecto a las CNN, para decodificar la información de la representación en el espacio latente al espacio nativo o inicial. Como se trata de obtener un resultado de resolución completa con las dimensiones iniciales de la entrada a partir de una representación comprimida, cada nivel consiste en una operación de *upsampling* (mediante técnicas diversas, como interpolación o *padding*) + concatenación de la salida del nivel correspondiente en el tramo codificador + dos convoluciones sucesivas + ReLU. De esta manera en cada nivel van aumentando las dimensiones de los mapas de activación hasta llegar a las dimensiones originales de la entrada. Cabe destacar la conexión que se establece entre los niveles análogos de los caminos descendente y ascendente mediante la concatenación de la salida del primero a la entrada del segundo. Esta estrategia evita la pérdida de información espacial de la imagen original durante el proceso de codificación-decodificación.

Este tipo de redes se emplean en el campo del análisis de imagen médica en importantes tareas como la segmentación o la síntesis de imagen, en las que el resultado deseado presenta las mismas dimensiones que la imagen de entrada.

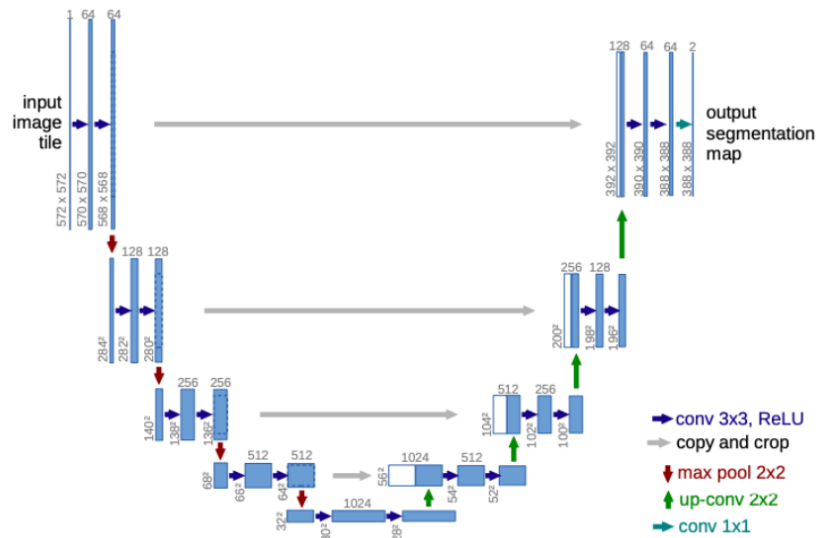


Figura 1.20: Esquema de la arquitectura original de la red U-Net propuesta por Ronneberger, Fischer y Brox 2015 a modo de ejemplo de CAE. Cada caja azul corresponde a un mapa de activación multi-canal. El número de canales se muestra sobre cada caja. Las dimensiones x-y se muestran en la esquina inferior izquierda de cada caja. Las cajas blancas representan los mapas de activación copiados y concatenados. Fuente: Ronneberger, Fischer y Brox 2015

1.5.5 Conceptos

A continuación se describen algunos conceptos típicos en el campo del DL, que se mencionarán más adelante durante el trabajo.

- Sobreajuste u *overfitting*:** Uno de los problemas más comunes a la hora de entrenar una red neuronal profunda es el sobreajuste u *overfitting*. Este fenómeno ocurre cuando la red se ajusta demasiado a los datos de entrenamiento, de tal manera que pierde su capacidad de generalización a otros datos, tal y como se puede observar en la Figura 1.21. Con el fin de evitar este problema, existen técnicas de regularización tales como el *dropout*, cuya idea esencial es desactivar neuronas aleatoriamente durante el entrenamiento de la red en cada iteración, de manera que se evita que la red memorice parte de los datos de entrada.

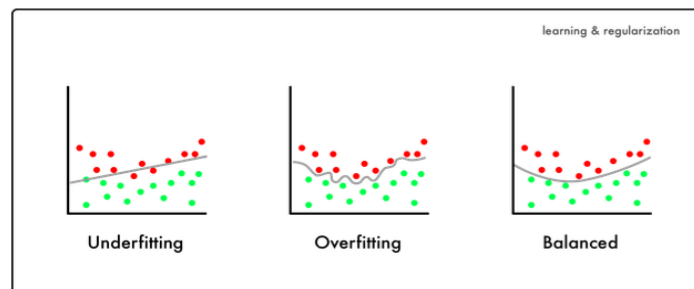


Figura 1.21: Gráficos ilustrativos de las situaciones de *underfitting*, *overfitting* y un buen ajuste a los datos. Los puntos representan datos de entrenamiento pertenecientes a dos clases distintas (rojo y verde) y la línea gris el límite de clasificación establecido por el método para una tarea de clasificación. Fuente: Parmar 2018

- **Data augmentation:** *Data augmentation* o generación de datos artificiales en castellano, es una técnica para aumentar la diversidad y cantidad del conjunto de datos de entrenamiento mediante la aplicación de transformaciones aleatorias pero realistas a los mismos. Este tipo de técnicas surge para superar los problemas de falta de datos en DL, pues este tipo de redes requiere de un gran conjunto de datos de entrenamiento.
- **Batch Normalization:** La activación mediante la función de activación en cada capa puede resultar en distribuciones de datos distintas a la salida de las capas de la red. Así, con el fin de aumentar la estabilidad de las redes neuronales profundas, es necesario normalizar los datos previamente a la entrada de cada capa. Para ello, se añaden capas previas de batch normalization a las convolucionales, con dos parámetros aprendibles que mapean la distribución de forma automática.
- **Fine-tuning:** La técnica de *fine-tuning* consiste en tomar una red previamente entrenada para una determinada tarea, y entrenarla de nuevo para una segunda tarea similar partiendo de los parámetros pre-entrenados. Asumiendo que la tarea original es similar a la nueva, el uso de una red pre-entrenada nos permite aprovechar la extracción de características más generales que se lleva a cabo en las capas iniciales de la red sin tener que realizar el entrenamiento completo desde cero.
- **Deep Supervision** (Lee y col. 2015, Dou y col. 2017): Una de las formas para mejorar el desempeño de las CNN es aumentando el número de capas convolucionales. Sin embargo, el incremento de capas hace el entrenamiento más dificultoso y aumenta el coste computacional. Con el fin de entrenar redes más profundas (con más capas), la supervisión profunda o *Deep Supervision* consiste en añadir ramas auxiliares de supervisión tras ciertas capas intermedias de la red durante el entrenamiento. La arquitectura resultante facilita el entrenamiento y proporciona mejores resultados.
- **Padding:** Un problema que surge al aplicar capas convolucionales en una CNN es que se tiende a perder píxeles del perímetro de la imagen con su correspondiente información. Puesto que normalmente se emplean *kernels* de pequeñas dimensiones, se pierden pocos píxeles en cada convolución, pero en las CNN esto tiene un efecto acumulativo al presentar múltiples capas convolucionales. Una solución sencilla a este problema es rellenar con píxeles adicionales alrededor de los bordes de la imagen de entrada, aumentando así el tamaño efectivo de la imagen. Esta técnica se conoce como *padding* y existen diversos tipos: desde rellenar con ceros (*zero padding*) hasta replicar los bordes de la imagen. En la Figura 1.18 podemos observar cómo gracias al *zero padding* aplicado sobre la imagen de entrada, el *kernel* alcanza todos los píxeles de la imagen y el tamaño final del resultado es el mismo que el de entrada.
- **Stride:** El salto o movimiento entre aplicaciones sucesivas del filtro a la imagen o mapa de activación durante la convolución se denomina *stride* y casi siempre es simétrico en las di-

mensiones de altura, anchura y profundidad. A mayor *stride*, menores serán las dimensiones del resultado de la convolución. Se puede observar un ejemplo en la Figura 1.22.

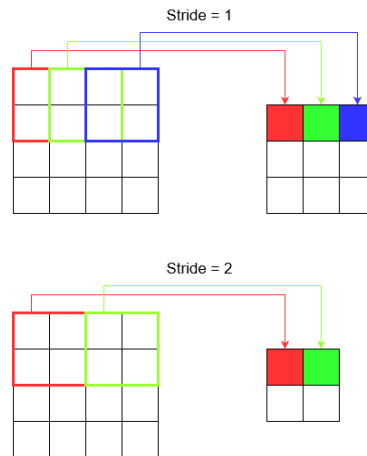


Figura 1.22: Esquema del movimiento en convoluciones con *stride* 1 y 2. Fuente: *Intuitive understanding of 1D, 2D, and 3D convolutions in convolutional neural networks* 1966

1.6 Motivación y Justificación

La IRM es una potente técnica para el estudio del cerebro humano y sus múltiples patologías asociadas tanto morfológica como funcionalmente, siendo idónea por sus numerosas ventajas: buena resolución de los tejidos blandos de la mano de inocuidad para el paciente, así como su variedad inherente.

Por otra parte, el campo del análisis automático de imagen médica y, concretamente de IRM, ha experimentado un auge en la última década gracias a la introducción de métodos basados en *Deep Learning*. El aprendizaje profundo, en particular las redes convolucionales, se han convertido rápidamente en una metodología elegida para el análisis de imagen médica, con aplicaciones tareas diversas, para muchas de las cuales representan actualmente el estado del arte.

El análisis cuantitativo de la IRM cerebral es esencial para el diagnóstico y seguimiento de múltiples enfermedades y afecciones neurológicas, y se basa principalmente en una segmentación precisa de las estructuras anatómicas de interés. Sin embargo, la segmentación automática de las estructuras cerebrales sigue siendo un problema sin resolver debido a las variaciones anatómicas normales en la morfología del cerebro, las variaciones en la configuración de adquisición y los escáneres de resonancia magnética, las imperfecciones de adquisición de las imágenes y las variaciones patológicas.

La IRM ofrece un amplio abanico de posibilidades en lo que se refiere a adquisición de contrastes en neuroimagen. La disponibilidad de esta información multispectral trae consigo una mejora de la comprensión del estado de los tejidos cerebrales por parte de los profesionales clínicos. De la misma manera, los resultados de los métodos automáticos de segmentación se ven en general

mejorados al recibir una entrada multimodal, es decir, si en lugar de proporcionar al método una única imagen del caso a segmentar, se proporcionan varias imágenes, siendo éstas distintos contrastes del mismo sujeto. Esto ocurre por la aportación de información complementaria sobre el mismo sujeto bajo estudio desde las perspectivas de los distintos contrastes. Por ejemplo, las imágenes T1 delimitan claramente materia gris de materia blanca, mientras las imágenes T2 distinguen mejor entre tejidos fluidos y tejido cortical, y las FLAIR permiten una mejor diferenciación de las lesiones.

No obstante, la adquisición de múltiples contrastes no siempre es posible debido a limitaciones temporales y de comodidad del paciente. Así, la síntesis de contrastes constituye una potencial herramienta para generar imágenes de distintas modalidades a partir de una original que, a pesar de todavía no haber conseguido resultados aptos para el diagnóstico clínico directo, sí que resulta útil para la mejora de procesos importantes tales como la segmentación. A pesar de haberse publicado un número considerable de métodos de síntesis de IRM, la mayoría de los desarrollados hasta la fecha son basados en cortes (o *slice-based*), en los cuales cada corte del volumen se predice de forma independiente (o con apoyo 3D limitado) lo que puede conducir a inconsistencias entre cortes vecinos. Además, el uso de un contexto espacial limitado (2D) puede resultar en resultados subóptimos debido a la falta de información del contexto en la tercera dimensión.

Para tratar el problema mencionado, se propone un método basado en *Deep Learning*, concretamente en redes neuronales convolucionales, para llevar a cabo la tarea de síntesis de contrastes cerebrales en IRM, capaz de generar imágenes T2 y FLAIR a partir de una imagen anatómica común T1 como imagen de origen. La red de aprendizaje profundo propuesta es una variante 3D de la conocida arquitectura UNet (Ronneberger, Fischer y Brox 2015) que procesa el volumen completo rompiendo con las inconsistencias en los volúmenes de salida resultantes relacionadas con los métodos basados en cortes 2D o *patches*, aumentando la coherencia en el volumen sintetizado y mejorando la precisión vía una perspectiva integrada tridimensional del contexto.

El fin último de la generación de imágenes artificiales es hacer útil la diversidad inherente de la IRM siendo aplicadas a tareas relevantes para su mejora. Así, se valida el método propuesto alimentando de forma multimodal a un método de segmentación con las imágenes sintetizadas (T2 y FLAIR) como entrada junto con la T1 original.

Capítulo 2

Objetivos

En el presente capítulo se expone el objetivo general a alcanzar del trabajo así como los objetivos específicos del mismo.

2.1 Objetivo general

El objetivo principal del presente trabajo de fin de grado consiste en el diseño e implementación de un método de síntesis de IRM cerebral basado en técnicas de aprendizaje profundo o *Deep Learning*, concretamente en redes neuronales convolucionales (CNN) 3D, capaz de generar volúmenes T2 y FLAIR a partir de un volumen anatómico común T1 como origen. El propósito final de este método es mejorar la precisión de la tarea de segmentación de los tejidos cerebrales mediante la introducción adicional de las imágenes sintetizadas como entrada multimodal (T1, T2 y FLAIR) al método de segmentación.

2.2 Objetivos específicos

- Familiarizarme con las imágenes de RM y sus diversas modalidades, así como con las técnicas para su análisis automático.
- Investigar y formarme en el campo del aprendizaje profundo o *Deep Learning*, y en particular en las redes neuronales convolucionales (CNN), así como sus aplicaciones en análisis de imagen médica, para ser capaz de comprender su funcionamiento y diseñar mis propias redes.
- Aprender a implementar CNNs en el lenguaje de programación Python con librerías dedicadas a aprendizaje profundo, principalmente Keras y Tensorflow.
- Diseñar y llevar a cabo un adecuado preprocesado para las imágenes de RM cerebral, incluyendo esto aplicar conocimientos previamente adquiridos durante el grado y aprender acerca de las técnicas del estado del arte en las numerosas etapas del preprocesado de IRM.
- Experimentar con las diversas posibilidades de la red implementada, introduciendo novedades y analizando su efecto sobre su desempeño, con el fin de optimizar el método.

- Estudiar y demostrar que la síntesis de imágenes de distintas modalidades de IRM cerebral mejoran la precisión de la segmentación de los tejidos cerebrales.

Capítulo 3

Materiales

En el presente capítulo se exponen los materiales empleados para la realización del trabajo, divididos en las categorías de datos, software y hardware.

3.1 Datos

Los datos de partida empleados en el presente trabajo consisten en imágenes de RM volumétricas del cerebro de las modalidades T1, T2 y FLAIR, procedentes de dos conjuntos de datos distintos. Con estas imágenes en formato NIfTI (*NIfTI: - Neuroimaging Informatics Technology Initiative* s.f.), previamente registradas a un espacio estándar MNI152 (Montreal Neurological Institute) (Fonov y col. 2009) y preprocesadas tal y como se detalla más adelante en el apartado 4.2, se ha entrenado y evaluado la red neuronal propuesta.

Base de datos I

Por una parte, para los experimentos de síntesis de T2 a partir de T1, se ha empleado la base de datos de libre acceso IXI (*IXI Dataset* s.f.). Este *dataset* contiene imágenes emparejadas T1/T2 de 580 sujetos sanos de varios hospitales de Londres (Reino Unido), tanto de máquinas de 1.5T como de 3T. Las imágenes de 3T fueron adquiridas en un escáner Philips Intera 3T (TR = 9.6ms, TE = 4.6ms, ángulo de giro = 8°, grosor del corte = 1.2 mm, tamaño del volumen = 256x256x150 vóxels, dimensiones del vóxel = 0.94x0.94x1.2mm³). Las imágenes de 1.5T fueron adquiridas en un escáner Philips Gyroscan 1.5T (TR = 9.8 ms, TE = 4.6 ms, ángulo de giro = 8, grosor del corte = 1.2 mm, tamaño del volumen = 256x256x150, dimensiones del vóxel = 0.94x0.94x1.2 mm³). Aquellas imágenes de sujetos que presentaban artefactos excesivos fueron excluidas de la selección, siendo el número final de sujetos empleados de 557. Este conjunto seleccionado fue a continuación dividido en conjuntos de entrenamiento, validación y test (520, 20 y 17 sujetos respectivamente).

Base de datos II

Por otra parte, para los experimentos de síntesis de imágenes FLAIR a partir de T1, se ha empleado la base de datos de volBrain (Manjón y Coupé 2016) formada por los distintos casos de parejas de imágenes T1/FLAIR que los usuarios que utilizan el software suben y dan permiso para ser utilizadas en investigación. Así, se trata de un conjunto de datos altamente heterogéneo, de sujetos tanto sanos como con distintos tipos de lesiones presentes, y procedentes de hospitales y escáneres muy diversos. En un principio se disponía de un total de 5.463 sujetos (pares de imágenes T1/FLAIR). Tras un proceso de selección en base a la calidad que se detalla más adelante en la sección 4.2 de Preprocesado, quedó un conjunto de datos seleccionado de 523 sujetos, que a continuación fue dividido en conjuntos de entrenamiento, validación y test (493, 10 y 20 sujetos respectivamente).

3.2 Software

Para llevar a cabo el desarrollo de la red neuronal convolutiva y todos los experimentos, se ha utilizado el lenguaje de programación Python en su versión 3.7.3, que en la actualidad se ha establecido como el lenguaje más utilizado en *Machine Learning*. Esto es debido a que, además de ser de código abierto, proporciona las herramientas necesarias para llevar a cabo este tipo de procesos de manera efectiva y relativamente sencilla, con potentes bibliotecas dedicadas a ello. En concreto, para nuestro proyecto de *Deep Learning* se han empleado como bibliotecas principales TensorFlow en su versión 1.12.0 y Keras en su versión 2.2.4., además de otras bibliotecas comunes con sus respectivas funciones. A continuación, se explican brevemente las principales.

- **Tensorflow** y **Keras**: Tensorflow es una biblioteca desarrollada por Google Brain para aplicaciones de aprendizaje automático y redes neuronales profundas o DL. Construido sobre TensorFlow, Keras es un *framework* de DL escrito en Python ampliamente utilizado debido a su facilidad de uso y su enfoque dirigido al usuario, así como su flexibilidad a la hora de realizar experimentos e ideas muy diversos de aprendizaje profundo. Keras es una abstracción, una API (*Application Programming Interface*) diseñada para ofrecer un uso fácil para el ser humano. Estas bibliotecas han sido empleadas para implementación, entrenamiento y evaluación de las redes con todos los experimentos pertinentes.
- Otras bibliotecas de Python empleadas:
 - **Numpy**: es la librería principal para informática científica, proporcionando estructuras de datos para cálculos (*arrays* multidimensionales) y operaciones matemáticas.
 - **Matplotlib**: es una librería dedicada a gráficas con un amplio abanico de opciones, empleada para la visualización de datos y resultados.
 - **Scikit-learn**: es la librería por excelencia de ML en Python, proporcionando diferentes tipos de herramientas para análisis de datos. En nuestro caso, la hemos empleado para el cálculo de métricas comunes como el MSE.
 - **Scikit-image**: es una librería dedicada al procesamiento de imagen en Python, que hemos empleado para el cálculo de métricas típicas de evaluación en imagen como es el PSNR.

- **Nibabel**: este paquete proporciona acceso de lectura/escritura a formatos de archivo médicos comunes en neuroimagen, como es el caso de nuestras imágenes en formato NIFTI.

Además del lenguaje de programación Python, también se ha utilizado Matlab en su versión R2019b para tareas fuera de la implementación, entrenamiento y evaluación de la red, tales como aquellas pertenecientes al preprocesado de las imágenes, la creación de máscaras del cerebro, la creación de un atlas anatómico de FLAIR y la obtención de resultados con uno de los métodos bajo comparación implementado en Matlab (REPLICA (Jog, Carass y col. 2017)).

Para el preprocesado de las imágenes, además de Matlab, se han utilizado herramientas software del estado del arte para tareas concretas:

- **Advanced Normalization Tools (ANTs)** (Avants y col. 2008) es un software basado en el algoritmo *Symmetric Normalization* (SyN) que representa el estado del arte en registro de imagen médica. Ha sido empleado en el trabajo para el registro de las imágenes al espacio estándar MNI152.
- **N4** (Tustison y col. 2010) es un método del estado del arte para corrección de inhomogeneidad, basado en las propiedades del histograma. Lleva a cabo un proceso iterativo de deconvolución del histograma, cálculo de *offsets* y regularización mediante el ajuste de *B-Splines*. Se ha empleado para la normalización en intensidad de las imágenes.
- **Statistical Parametric Mapping (SPM)** (*SPM - Statistical Parametric Mapping* s.f.) es un *software* para análisis de imagen cerebral de IRM con funcionalidades muy diversas. En este caso, ha sido empleado para segmentar los volúmenes cerebrales para crear las máscaras, cuya función se explicará más adelante.

Para la visualización de las imágenes, se han empleado indistintamente dos *softwares* de funcionalidad similar: **ITK-SNAP** (Yushkevich y col. 2006) y **MRICro** (Chris Rorden's Neuropsychology Lab, University of South Carolina s.f.).

3.3 Hardware

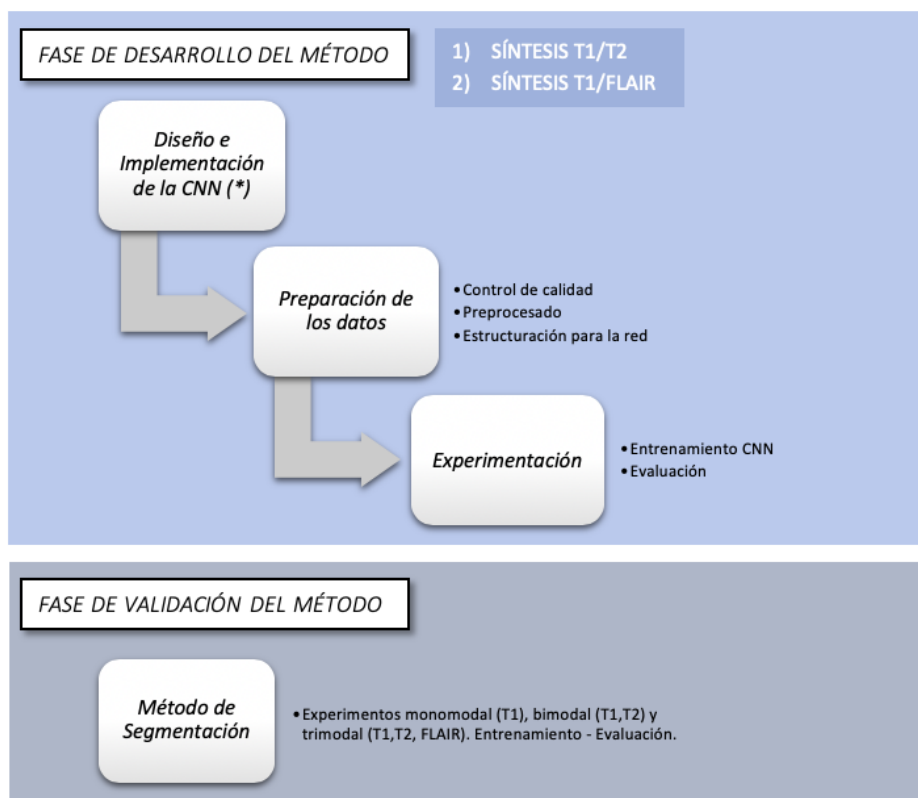
Todos los experimentos fueron realizados con un ordenador de sobremesa con las especificaciones que se detallan en la Tabla 3.1. Para los experimentos con la red neuronal convolutiva se ha utilizado una tarjeta gráfica o *Graphics Processing Unit* (GPU) NVIDIA GeForce GTX 1080 Ti con 11 GB de memoria.

Especificaciones	
Procesador	AMD Ryzen 7 1700 Eight-Core (16 threads), 3 GHz
Memoria RAM	16 GB
SSD (Almacenamiento en disco sólido)	Samsung SSD 860 EVO 250 GB
HDD (Almacenamiento en disco mecánico)	WDC WD20EFRX-68EUZN0 2 TB
Sistema Operativo	Windows 10 Enterprise

Tabla 3.1: Especificaciones del ordenador empleado para el proyecto.

En el presente capítulo se exponen los métodos empleados para la realización del trabajo, comenzando por el preprocesado llevado a cabo a los datos, para a continuación exponer la arquitectura final de la red implementada, explicar el entrenamiento y evaluación de la red y finalmente acabar con una breve explicación de la experimentación llevada a cabo.

4.1 Metodología general



(*) Sujeta a modificaciones durante la etapa de experimentación.

Figura 4.1: Esquema de la metodología seguida en el proyecto.

El esquema de la Figura 4.1 muestra la metodología general de trabajo seguida durante el desarrollo del proyecto. Se han llevado a cabo dos fases diferenciadas: la de desarrollo del método en sí y la de validación de éste mediante su aplicación a la segmentación del cerebro, tal y como se explicará más adelante en el apartado 4.5. A su vez, la fase de desarrollo tiene dos partes diferenciadas y análogas que siguen un procedimiento general similar: en primer lugar, se realizó el desarrollo del método de síntesis de imágenes T2 a partir de imágenes T1, y a continuación el desarrollo del método de síntesis de imágenes FLAIR a partir de imágenes T1.

4.2 Preprocesado

Una adecuada preparación de los datos previa a la realización de los experimentos es, en general y en nuestro caso concreto, esencial para la obtención de buenos resultados en el proyecto. En el campo del análisis de imagen médica y, más concretamente en imágenes de RM, las posibilidades de preprocesado son innumerables, con diversos pasos o etapas, y se debe seleccionar el proceso que mejor se adecúe al propósito específico del trabajo. Así, a continuación se detallan y justifican los dos preprocesados llevados a cabo para cada una de las vías del trabajo (pares T1/T2 y pares T1/FLAIR), siendo ligeramente distintos al tratarse de diferentes modalidades de imagen procedentes de diferentes bases de datos.

4.2.1 Imágenes T1/T2 (Base de datos I)

La calidad de los datos de entrenamiento de una CNN influye de manera muy significativa en sus resultados. Previamente al preprocesado propiamente dicho, se llevó a cabo un control de calidad de los datos sencillo para descartar aquellas imágenes que presentaran artefactos excesivos. Así, aquellos pares de imágenes T1/T2 que presentaban entre ellas un coeficiente de correlación de Pearson significativamente menor que la media fueron directamente excluidos del conjunto de datos, siendo el número final de sujetos empleados de 557.

I. Filtrado de ruido.

En el caso de las imágenes T1 y T2 de la base de datos I, se realizó en primer lugar un filtrado del ruido, primer paso esencial en el preprocesado de IRM. Las imágenes de RM están generalmente corrompidas por ruido aleatorio procedente del proceso de adquisición. Dicho ruido introduce una incertidumbre a la hora de medir parámetros cuantitativos, dificultando la estimación de las distintas propiedades de los tejidos estudiados. Además, es conveniente realizar la reducción del ruido como paso inicial del preprocesado, previo a otros pasos que, de no ser así, se verían dificultados por la presencia de este ruido, como es el registro. En el caso de nuestra CNN, la presencia de ruido aleatorio en las imágenes tanto de entrada como de salida dificultaría el aprendizaje del mapeo, proporcionando así resultados subóptimos.

Así, para reducir el ruido de las imágenes, se ha aplicado a todas ellas el filtro adaptativo Non Local Means (NLM), propuesto por Manjón, Coupé y col. 2010, una adaptación del conocido filtro Non Local Means propuesto inicialmente por Buades, Coll y Morel 2011 para lidiar con imágenes de RM con niveles de ruido variables espacialmente, para distribuciones de ruido tanto

Gaussianas como de Rician. En la Figura 4.2 podemos observar el efecto, visualmente perceptible, del filtrado sobre la imagen.

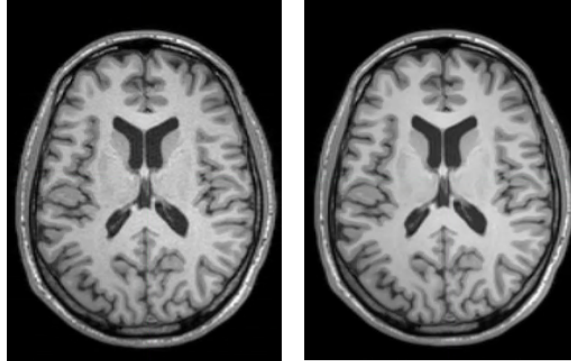


Figura 4.2: Corte axial de imagen de RM potenciada en T1 de la base de datos IXI, antes y después del filtrado de ruido con el filtro adaptativo NLM (izquierda y derecha respectivamente).

II. Registro.

El registro se define como la determinación de la transformación geométrica que alinea los puntos de un objeto con los correspondientes del mismo u otro objeto, con el fin de que las imágenes estén en el mismo marco de referencia espacial. Para el presente proyecto, es de vital importancia disponer de imágenes registradas a un espacio estándar para poder localizar las estructuras anatómicas con unas coordenadas determinadas en dicho espacio. Podríamos referirnos al registro como una normalización espacial de las imágenes.

Así, una vez reducido el ruido, se realizó primero la estimación de un registro rígido entre las imágenes T1 y T2 y se concatenó con un registro afín de todo el *dataset* al espacio estándar MNI152 empleando el software ANTs, tal y como se muestra en el esquema de la Figura 4.3.

Se optó por un corregistro rígido entre los pares de imágenes T1/T2 por tratarse de casos intra-sujeto (registro de una imagen de un sujeto respecto a otra del mismo sujeto). Por ello, no es necesario un escalado, al ser el volumen de la cabeza invariante, siendo suficiente con 6 grados de libertad (las 3 dimensiones de traslación y las 3 de rotación). Como función de coste a maximizar, se optó por la Información Mutua Normalizada o Normalised Mutual Information (NMI) (Ecuación 4.2) por tratarse de un registro intermodal (T1/T2), método basado en la teoría de la información que utiliza la entropía de Shanon para estimar el grado de coherencia espacial entre dos objetos.

Entropía conjunta:

$$H(A, B) = - \sum_{i,j} p(i, j) \cdot \log [p(i, j)] \quad (4.1)$$

Siendo A y B las dos imágenes y $p(i, j)$ la distribución de probabilidad conjunta de las intensidades i y j .

Información mutua normalizada:

$$NMI(A, B) = \frac{H(A) + H(B)}{H(A, B)} \quad (4.2)$$

Siendo $H(A)$ y $H(B)$ las entropías individuales de las imágenes A y B y $H(A, B)$ la entropía conjunta de A y B .

Una vez estimada la transformación del corregistro rígido, que alinea los pares de imágenes T1/T2 de cada sujeto, se concatena a un registro afín (12 grados de libertad: translación (3), rotación (3), escalado (3) y shear (3)) del conjunto de imágenes completo al espacio estándar MNI152 del *Montréal Neurological Institute*, también maximizando la NMI. Así, las dimensiones de las imágenes quedan en 181x217x181 vóxeles (con una resolución de 1 mm³), que corresponden a las dimensiones del atlas de T1 MNI152, y quedan todas situadas en un marco de referencia espacial estándar.

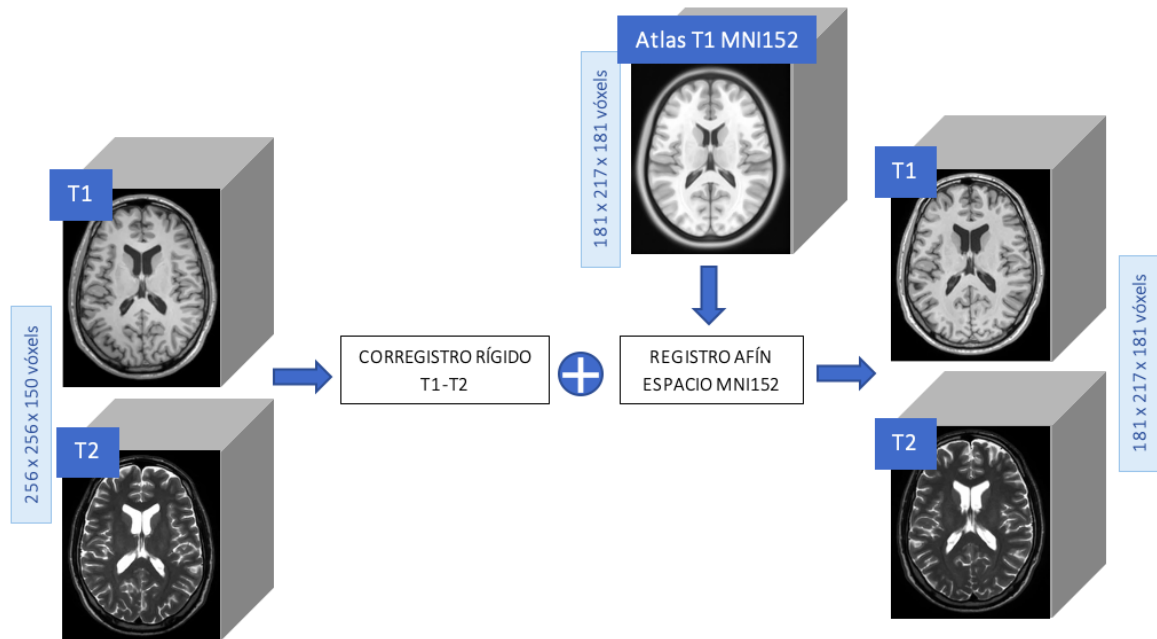


Figura 4.3: Proceso de registro de las imágenes: corregistro T1-T2 seguido de registro al espacio MNI152.

III. Corrección de inhomogeneidad

Además del ruido aleatorio aditivo anterior, las imágenes de RM se ven comúnmente afectadas por variaciones no anatómicas de la señal, debidas a causas como un campo magnético o una sensibilidad de la antena receptora no completamente uniformes. En consecuencia, la imagen de RM presenta un campo multiplicativo de variación lenta (campo de baja frecuencia B) de la intensidad en la imagen que dificulta la obtención de información cuantitativa de ésta.

$$Y = X \cdot B + n \quad (4.3)$$

Donde Y es la imagen adquirida, X la imagen original deseada, B en ruido de inhomogeneidad mutiplicativo y n el ruido aleatorio aditivo.

Así, una vez registradas las imágenes, se realizó una corrección de inhomogeneidad de las imágenes registradas empleando el método N4 para eliminar este ruido de baja frecuencia.

IV. Normalización de intensidad

Como se ha mencionado previamente, la red propuesta aprende un mapeo no lineal entre imágenes T1 y T2. Sin embargo, los resultados del mapeo se ven dificultados si los espacios fuente y objetivo no son coherentes en intensidad. A diferencia de otras modalidades de imagen médica como TAC en la cual existe una escala de intensidades estandarizada (Hounsfield), en RM las distribuciones de intensidad exhiben variaciones significativamente grandes dependiendo de varios factores técnicos como la fuerza del campo magnético o los parámetros de adquisición (TE, TR, ángulo de giro, etc.). De ahí surge la importancia de la estandarización en intensidad de los volúmenes de entrada y salida a un espacio de intensidad canónico de referencia (Figura 4.4). El hecho de que las imágenes tanto de entrada como de salida presenten un rango y distribución de intensidad determinados, sugiere que podría facilitar el aprendizaje del mapeo de la CNN, permitiendo la obtención de mejores resultados.

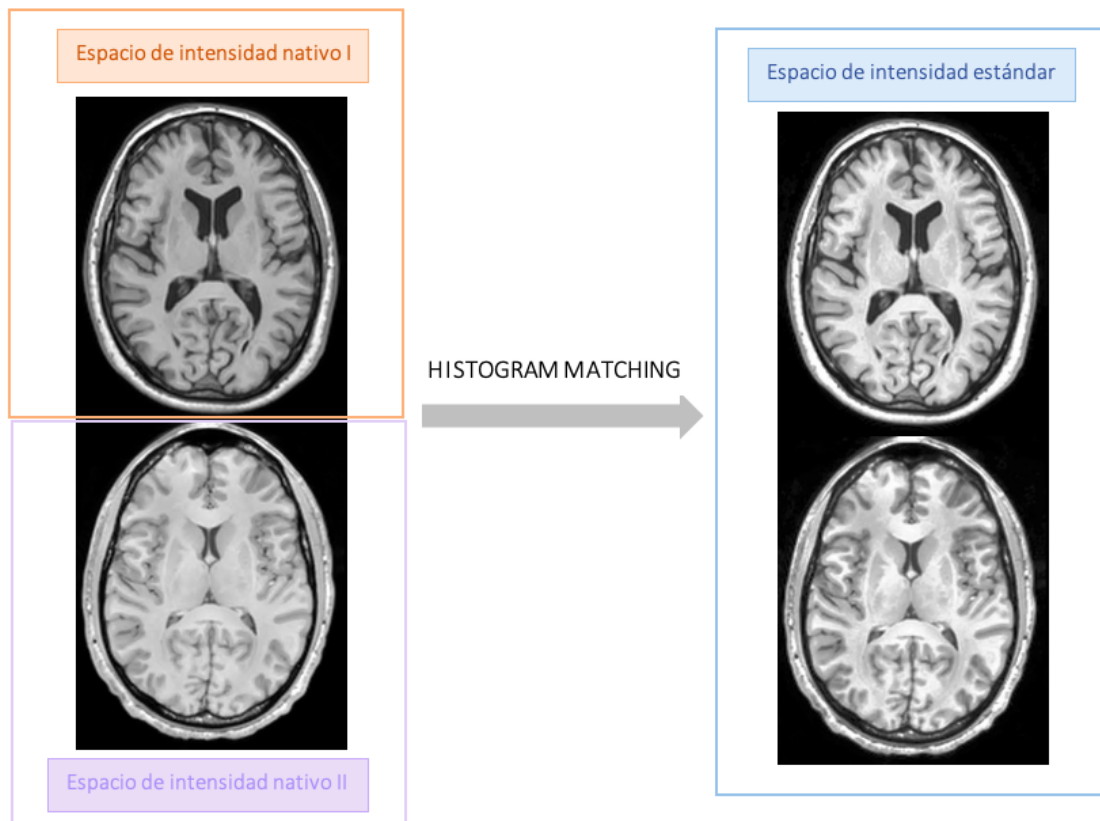


Figura 4.4: Esquema normalización de intensidad con *histogram matching*: paso de espacios nativos de intensidad diferentes a un mismo espacio de intensidad estándar.

En el presente trabajo se propone el uso de *histogram matching* (HM) (Gonzales y Fittes 1977) como método potente de normalización de intensidad, que consiste en la transformación de una imagen de modo que su histograma coincide con un histograma especificado de una imagen de referencia (Figura 4.5). En este caso, se han empleado como volúmenes de referencia los modelos o *templates* no lineales MNI152 de T1 y T2, para las imágenes T1 y T2 respectivamente. Finalmente, tras este mapeo de intensidad, aplicamos una normalización en proporción a la media consistente en dividir el volumen completo por su media.

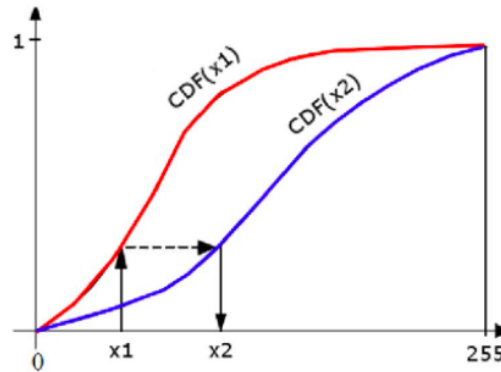


Figura 4.5: Mapeo de intensidades en histogram matching. CDF = *Cumulative Density Function* (histograma acumulado), en rojo la imagen a normalizar, y el azul la imagen de referencia. Fuente: Gonzalez, Woods y Eddins 2004

V. Otros.

Cabe mencionar que previamente a la entrada a la red, todos los volúmenes son sometidos a un *padding* simétrico o de replicación, que rellena con el reflejo o espejo de la imagen a lo largo de los bordes. Se ha empleado este tipo de *padding* por su menor interferencia en la distribución de los datos en comparación a un *zero padding*. El volumen se modifica añadiendo una fila en cada dimensión, de manera que su tamaño queda de 182x218x182 vóxeles, siendo así divisible por 2 en todas sus dimensiones, lo que facilita las posteriores operaciones de *pooling* en la red.

4.2.2 Imágenes T1/FLAIR (Base de datos II)

El preprocesado de los pares de imágenes T1/FLAIR resultó ligeramente distinto al tratarse de bases de datos muy diferentes. Tal y como se ha explicado previamente en el apartado Datos, ésta se trata de una base de datos muy heterogénea construida a base de las imágenes subidas por los usuarios de volBrain, por lo que cabía esperar no solo imágenes muy diferentes sino también errores o equivocaciones a la hora de la subida de las imágenes. Así, esta base de datos requirió un control de calidad más exhaustivo, que se realizó de forma manual visualizando una por una todas las parejas de imágenes T1/FLAIR y descartando del conjunto de datos final aquellas que o bien presentaban artefactos excesivos, o bien no alcanzaban una calidad de imagen suficiente en cuanto a resolución espacial, o bien no se correspondían con la modalidad de imagen especificada.

En cuanto al preprocesado propiamente dicho, las imágenes ya habían sido previamente sometidas a un filtrado del ruido y una corrección de inhomogeneidad y registradas al espacio MNI152. Así, únicamente fue necesario realizar la normalización en intensidad.

Para este paso del preprocesado, siguiendo un procedimiento análogo al de la base de datos I, se ha optado por el método *histogram matching* tomando como referencia un atlas. Para las imágenes T1 se ha empleado el atlas MNI152 de T1 al igual que en la base de datos I, sin embargo, no se dispone del equivalente para FLAIR. Así, se creó un atlas FLAIR propio realizando la mediana, por ser una medida más robusta que la media, de 52 imágenes FLAIR de la base de datos registradas de forma no lineal al atlas MNI152 T1 con el software ANTs, y su versión invertida horizontalmente. En la Figura 4.6 podemos observar un corte axial, sagital y coronal del atlas FLAIR generado.

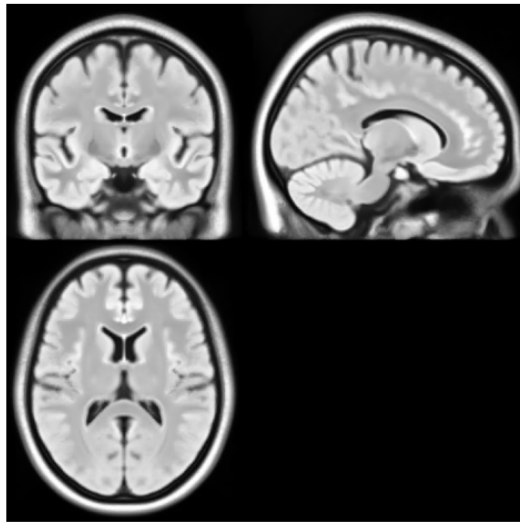


Figura 4.6: Atlas FLAIR de 52 sujetos.

Finalmente, de la misma manera que en la base de datos I, aplicamos una normalización en proporción a la media consistente en dividir el volumen completo por su media, y se realiza el *padding* simétrico añadiendo una fila por dimensión (nuevas dimensiones del volumen 182x218x182 vóxels).

4.3 Arquitectura de red.

En el presente apartado se expone la arquitectura final propuesta para la CNN de síntesis T1/T2 y T1/FLAIR. Cabe mencionar que esta arquitectura final es el resultado de un largo proceso de experimentación a base de introducir modificaciones, siendo ésta aquella que mejores resultados proporcionó para los dos casos de síntesis. Tal y como se ha mencionado en el apartado de Software, esta arquitectura de red ha sido implementada en Python con las librerías TensorFlow y Keras.

El método propuesto es una red 3D tipo U-Net (Figura 4.8), que aprende un mapeo no lineal entre la imagen de entrada (T1) y la imagen objetivo (T2-FLAIR), con un camino codificador y uno decodificador. Sin embargo, el uso de volúmenes de entrada y salida de resolución completa (182x218x182 vóxeles) limita el número de filtros de las capas de convolución por razones de capacidad computacional (en nuestros experimentos iniciales podíamos usar un número máximo de 8 filtros), proporcionando así resultados subóptimos. Para lidiar con esta limitación, se propone una descomposición del volumen de entrada en 8 sub-volúmenes diezmando el volumen original por un factor de 2 en las tres dimensiones con un salto de 1 vóxel en cada una de ellas (Figura 4.7). Estos 8 sub-volúmenes se estructuran como un tensor de entrada de dimensiones 91x109x91x8, conteniendo todos los vóxeles del volumen de entrada original. Esta descomposición en 8 canales nos permite usar un número de filtros mayor en las capas de codificación y decodificación y al mismo tiempo utilizar todos los datos del volumen de entrada 3D.

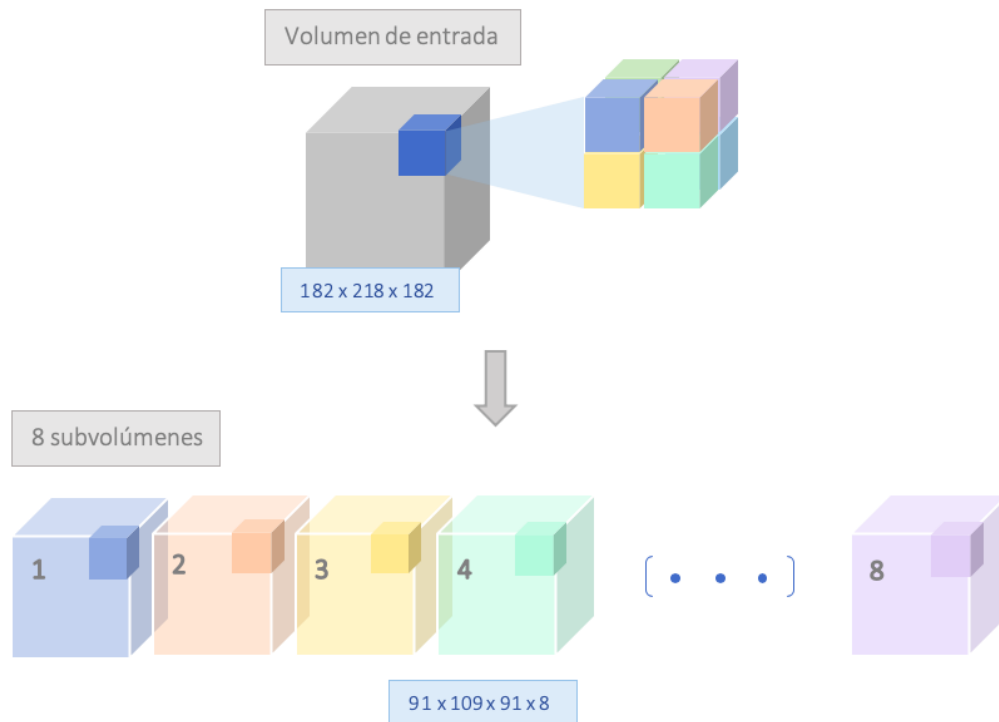


Figura 4.7: Descomposición del volumen de entrada en 8 sub-volúmenes (8 canales).

En particular, el camino descendente o codificador (*encoder path*) se compone de 3 niveles de resolución consistentes en una capa convolutiva 3D (tamaño de *kernel* de 3x3x3 vóxeles) para cada nivel de resolución con activación ReLU y capas de *batch normalization* seguidas de una capa de convolución *strided* (*stride* = 2) para *downsampling* o disminución de la resolución.

Entre el camino descendente y el ascendente, tenemos el nivel latente con dos bloques sucesivos de convolución 3D (tamaño de *kernel* 3x3x3) con activación ReLU y *batch normalization*.

De forma simétrica, el camino ascendente o decodificador se compone de 3 niveles consistentes en una capa de upsampling para cada nivel de resolución, se concatena la salida del nivel correspondiente del camino descendente, y finalmente dos bloques de una capa convolutiva 3D (tamaño de kernel 3x3x3 vóxeles) con activación ReLU y *batch normalization*.

El número de filtros usados comienza con 40 filtros en el primer nivel de resolución y se va incrementando por un factor de 2 en los subsiguientes niveles de menor resolución con el fin de compensar la pérdida de resolución espacial. De la misma manera, el número de filtros va disminuyendo por un factor de 2 en cada nivel de resolución en el camino decodificador.

Como el uso de la misma descomposición en canales en la salida podría resultar en inconsistencias locales de intensidad en la imagen sintetizada, se ha añadido un bloque de reconstrucción final que consiste en una capa de interpolación trilineal seguida de una capa de convolución 3D (con 8 filtros y tamaño de *kernel* 3x3x3 vóxeles) más una capa con ReLU y *batch normalization*. Finalmente, se usa una última capa de convolución 3D con un único filtro de tamaño 1x1x1 para producir la salida. Utilizamos un enfoque de *Deep supervision* para entrenar la red. La primera salida es la salida de baja resolución en canales, y la segunda es el volumen de resolución completa tras el bloque de reconstrucción. Así, finalmente la CNN diseñada tiene un número total de 9.135.865 parámetros, de los cuales 9.131.769 son entrenables.

En las convoluciones se emplea *zero padding* (parámetro *padding* = ‘*same*’) de manera que la imagen quede totalmente cubierta por el filtro y el *stride* especificado. Por otra parte, en referencia al *downsampling*, se pueden utilizar tanto capas de *pooling* clásico como de convolución *strided* para resumir los datos. No obstante, la ventaja de emplear una capa de convolución *strided* en lugar de una capa de *pooling* consiste en que la primera tiene parámetros a optimizar mientras la segunda es una operación fija. De esta manera, la CNN puede aprender a resumir los datos, mejorando así la precisión de la red (Ayachi y col. 2018).

Se puede observar un esquema de la red completa en la Figura 4.8.

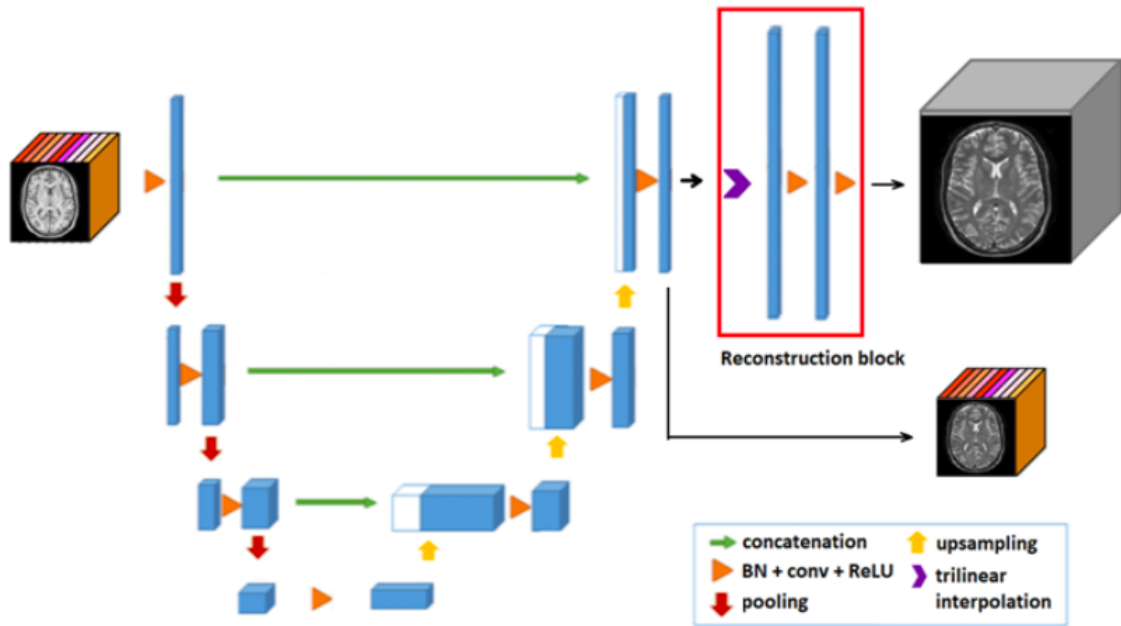


Figura 4.8: Esquema de la CNN 3D propuesta para síntesis de contrastes en IRM.

4.4 Entrenamiento y evaluación de las redes.

Tras el preprocesado de los datos y el diseño e implementación de la arquitectura de la CNN, se procede al entrenamiento de las redes y su subsiguiente evaluación.

Tal y como hemos mencionado en el apartado Datos, para la red T1/T2 seleccionamos de forma aleatoria 520 casos para entrenamiento, 20 para validación y 17 para test, y para la red T1/FLAIR 493 casos para entrenamiento, 10 para validación y 20 para test.

4.4.1 Entrenamiento

Red de síntesis T1/T2

Entrenamos la red proporcionando como datos de entrada los 8 sub-volumenes T1 de cada uno de los 520 casos de entrenamiento, y como datos de referencia de salida los volúmenes T2 pre-procesados completos, así como descompuestos en los 8 sub-volumenes (8 canales).

Para entrenar la red, empleamos el optimizador Adam (Kingma y Ba 2014) de la librería Keras con un *learning rate* de 0.001. Adam, ya mencionado en el apartado 1.5 de la introducción, es un método de descenso del gradiente estocástico basado en la estimación adaptativa de los momentos de primer y segundo orden. Además, el método es eficiente computacionalmente, requiere poca memoria y es adecuado para problemas de gran tamaño en términos de datos o parámetros, como es el caso.

Usamos una función tipo generador de datos para alimentar la red durante 200 *epochs* (50 casos por *epoch*). Empleamos *data augmentation* dentro del generador consistente en voltear horizontalmente (izquierda-derecha) los pares de volúmenes de entrada y salida de forma aleatoria, aprovechando la pseudo-simetría del cerebro.

En cuanto a la función de coste o *loss function*, se ha empleado el clásico error cuadrático medio (MSE), que mide el promedio de los errores al cuadrado (ecuación 4.4).

$$MSE(\hat{y}_i, y_i) = \frac{1}{n} \sum_i (\hat{y}_i - y_i)^2 \quad (4.4)$$

Siendo \hat{y}_i e y_i los valores de intensidad del píxel i de las imágenes sintetizada y *ground-truth* respectivamente, y n el número de píxeles de la imagen.

Red de síntesis T1/FLAIR

Para el caso de síntesis de FLAIR, de la misma manera que para la síntesis de T2, entrenamos la red proporcionando como datos de entrada los 8 sub-volúmenes T1 de cada uno de los 493 casos de entrenamiento, y como datos de referencia de salida los volúmenes FLAIR preprocesados completos, así como descompuestos en los 8 sub-volúmenes (8 canales).

En cuanto al optimizador con el cual se entrena la red, se emplea de nuevo el método Adam por su adecuación al problema y por haber mostrado un buen desempeño en el entrenamiento de la red para la síntesis de T2, siendo la síntesis de FLAIR una tarea cuasi idéntica a nivel de optimización.

Al igual que para la síntesis de T2, usamos la misma función tipo generador de datos para alimentar la red durante 200 *epochs* (50 casos por *epoch*). Se emplea asimismo la técnica de *data augmentation* recién descrita.

En cuanto a la función de coste, se ha empleado una combinación del error absoluto medio (MAE o norma L1), y el error cuadrático medio (MSE o norma L2), por ser la que mejores resultados ha proporcionado.

$$Loss(\hat{y}, y) = \|\hat{y}, y\|_1 + \|\hat{y}, y\|_2 \quad (4.5)$$

Siendo \hat{y} e y los volúmenes sintetizado y *ground-truth* respectivamente.

4.4.2 Evaluación

Para medir el desempeño del método basado en la calidad de los resultados, se ejecutan las predicciones con las redes entrenadas para el conjunto de imágenes de test (17 sujetos en el caso de T1/T2 y 20 en el caso de T1/FLAIR), obteniendo así las imágenes sintetizadas, y seguidamente se calculan 3 métricas diferentes que proporcionan información desde distintas perspectivas sobre la similitud entre el volumen sintetizado y el *ground-truth*.

- Proporción máxima de señal a ruido o *Peak Signal-to-Noise Ratio* (PSNR): se define como la relación entre la energía máxima posible de una señal y la potencia del ruido corruptor que afecta a la fidelidad de su representación. Puesto que muchas señales tienen un rango dinámico muy amplio, el PSNR generalmente se expresa en escala logarítmica, utilizando el decibelio como unidad, y su rango de valores se extiende de $[0, +\infty]$. Se usa para medir la calidad de la imagen sintetizada, siendo en este caso la señal la imagen original o *ground-truth* y el ruido el error en la imagen sintetizada, calculado como la raíz del error cuadrático medio (RMSE). (Hore y Ziou 2010)

$$PSNR = 20 \cdot \log_{10}(I/RMSE) \quad (4.6)$$

- *Structural Similarity Index Measure* (SSIM) (Wang y col. 2004): esta métrica propone la combinación de la similitud entre dos imágenes y e \hat{y} en tres ámbitos: luminancia, contraste o intensidad y estructura, como marco de evaluación de la calidad de una imagen \hat{y} con respecto a una imagen y . Se trata de una métrica adimensional cuyo rango de valores se extiende entre $[-1, 1]$, significando el valor 1 que las imágenes son idénticas.

$$SSIM(x, y) = \frac{(2\mu_y\mu_{\hat{y}} + C_1)(2\sigma_{y\hat{y}} + C_2)}{(\mu_y^2 + \mu_{\hat{y}}^2 + C_1)(\sigma_y^2 + \sigma_{\hat{y}}^2 + C_2)} \quad (4.7)$$

Donde μ_y , $\mu_{\hat{y}}$, σ_y , $\sigma_{\hat{y}}$ son las medias y varianzas de las imágenes y e \hat{y} respectivamente, $\sigma_{y\hat{y}}$ es la covarianza entre las imágenes y e \hat{y} , y C_1 y C_2 son dos constantes que aportan estabilidad al cálculo de la métrica.

- Coeficiente de correlación de Pearson (CC): es un estadístico que mide la correlación lineal entre dos variables \hat{y} e y , siendo en nuestro caso las imágenes sintetizada y *ground-truth* respectivamente. Tiene un rango de valores de $[-1, +1]$, siendo $+1$ una correlación positiva total, 0 correlación lineal nula y -1 correlación negativa total.

$$CC = \frac{\frac{1}{N} \sum_i (I_{\hat{y}}(i) - \mu_{\hat{y}})(I_y(i) - \mu_y)}{\sqrt{\frac{1}{N} \sum_i (I_{\hat{y}}(i) - \mu_{\hat{y}})^2} \sqrt{\frac{1}{N} \sum_i (I_y(i) - \mu_y)^2}} \quad (4.8)$$

Siendo $I_{\hat{y}}(i)$, $I_y(i)$, $\mu_{\hat{y}}$, μ_y los valores de intensidad del píxel i y de intensidad media de las imágenes \hat{y} (sintetizada) e y (*ground-truth*) respectivamente, y N el número de píxeles de la imagen.

Además de las métricas clásicas en imagen PSNR y SSIM, se ha incluido el Coeficiente de Correlación (CC) como una potente métrica para medir la información estructural compartida entre los volúmenes *ground-truth* y sintetizado de forma independiente de la intensidad, por lo que se ha considerado la métrica más robusta de entre las tres para la evaluación de la tarea concreta de síntesis, pues se trata de generar los contrastes correctos entre tejidos independientemente del nivel de intensidad.

Estas métricas de evaluación fueron calculadas tanto en los volúmenes completos o *full-head* como en los volúmenes exclusivamente cerebrales enmascarados con las máscaras del cerebro creadas o *skull-stripped* (sin el cráneo), pues esta es la zona de interés principal a sintetizar. Para ello, se han generado máscaras binarias del cerebro para todos los volúmenes objetivo T2 siguiendo el siguiente proceso:

- I. Segmentación de los volúmenes cerebrales con SPM.
- II. Suma de los mapas de probabilidad de los tejidos correspondientes a sustancia blanca, sustancia gris y líquido cefalorraquídeo, generando así un mapa de probabilidad del cerebro.
- III. Se seleccionan aquellos píxeles cuya probabilidad sea mayor de 0.5, generando así una primera máscara binaria.
- IV. Para solventar los posibles errores de segmentación, se selecciona únicamente la mayor de todas las componentes conectadas (conectividad a 26). Para ello se ha empleado la función *bwconncomp* de Matlab.
- V. Finalmente, se realiza un relleno de agujeros con la función *imfill* de Matlab.

4.4.3 Experimentos

Múltiples experimentos fueron realizados para comprobar el impacto de las diversas modificaciones que se fueron introduciendo en el método, tanto en el preprocesado y estructuración de los datos como en la arquitectura y entrenamiento de la red. Se trata de un proceso incremental a través del cual se ha ido modificando, entrenando y evaluando la red y se han ido estableciendo aquellos cambios que proporcionaron mejoras en los resultados, hasta llegar al modelo entrenado final.

Red de síntesis T1/T2

En el caso de la red de síntesis de T2 a partir de T1, se han ido realizando experimentos introduciendo modificaciones diversas:

- Modificaciones de los hiperparámetros de la red como el tamaño de *kernel* (3x3x3 y 5x5x5) o el número de filtros.
- Modificaciones en la normalización de intensidad (HM).
- Modificaciones diversas de la arquitectura de la red.
- Imposición de distintos umbrales de correlación para los datos de entrenamiento.

- Experimentos con distintas funciones de coste.

Así, se fue evaluando el impacto de cada modificación sobre los resultados, estableciendo aquellos que introducían mejoras significativas en las métricas de evaluación, hasta finalmente llegar al modelo de red final.

En el capítulo 5 de Resultados y Discusión, se expondrán los resultados de los experimentos más significativos, que se describen a continuación.

En cuanto a la función de coste, se han realizado experimentos con distintas opciones y combinaciones lineales de ellas:

- *Mean Square Error* o error cuadrático medio (MSE) (ecuación 4.4).
- *Mean Absolute Error* o error absoluto medio (MAE)(Tabla 1.3).
- Diferencia de gradiente (GDL, *Gradient Difference Loss*).
- Correlación inversa (1-CC).

Además de las clásicas MAE y MSE, se propone una nueva función de coste mixta (ecuación 4.9), que combina información local y global, minimizando el error absoluto medio tanto de las intensidades como de los gradientes locales a la vez que maximiza la correlación global de la imagen. Se incorpora esta función de diferencia de gradiente de la imagen con el fin de mejorar la estimación de información de alta frecuencia y evitar la borrosidad en el volumen sintetizado.

$$Mix(\hat{y}, y) = (1 - CC(\hat{y}, y)) (\|\hat{y}, y\|_1 + \|\nabla\hat{y}, \nabla y\|_1) \quad (4.9)$$

Donde CC es el coeficiente de correlación, \hat{y} e y son las imágenes sintetizada y original respectivamente.

Por otra parte, se realizó un experimento para probar el impacto de la normalización de intensidad del *dataset* con *histogram matching* en el desempeño de la red. Para ello, se entrenó y evaluó la red con los volúmenes con y sin la normalización con *histogram mathing*.

Asimismo, para evaluar la importancia del contexto 3D comparamos el método 3D propuesto con su correspondiente versión 2D. Para hacer esto, simplemente intercambiamos las convoluciones 3D por convoluciones 2D y usamos una UNet 2D estándar eliminando la descomposición en canales, la reconstrucción y la *deep supervision*. Para hacer una comparación justa, incrementamos el número de filtros a 70 (en lugar de 40) para tener un número similar de parámetros entrenables (9.323.161 parámetros totales, de los cuales 9.316.021 entrenables, frente a los 9.131.769 del método propuesto).

Para evaluar el impacto de la arquitectura 3D propuesta, la comparamos con una UNet 3D básica (sin descomposición en canales ni reconstrucción) con el máximo número de filtros que cabían en memoria ($nf = 8$).

Con tal de comparar el desempeño de nuestro método en la tarea de sintetizar imágenes T2 dadas imágenes T1 como entrada, se han usado dos métodos representativos del estado del arte:

- El primero es REPLICA (Jog, Carass y col. 2017), un método de síntesis de imagen super-resuelta basado en *Random Forest* (Breiman 2001), que aprende una regresión no lineal para predecir las intensidades de tejidos alternos dadas unas intensidades de entrada específicas. Entrenamos REPLICA usando imágenes del *dataset* IXI (base de datos I) con el código de los autores. En este método diferentes regresores realizan un análisis de características en un rango de niveles de resolución, y finalmente las salidas individuales son promediadas para proporcionar el contraste de salida final.
- El segundo método bajo comparación es pGAN (Dar y col. 2019), un método basado en *Deep Learning* para síntesis de múltiples contrastes en IRM, empleando redes adversarias generativas condicionales (*generative adversarial networks*, GAN). El conjunto de imágenes de test fue amablemente procesado por los autores, que nos proporcionaron los resultados de test que más adelante se exponen en Resultados.

Red de síntesis T1/FLAIR

En el caso de la red de síntesis de FLAIR a partir de T1, se siguió una experimentación similar a la primera red en cuanto a la aplicación de la dinámica modificación – entrenamiento – evaluación, pero partiendo de una arquitectura de red ya consolidada en los experimentos de T1/T2, por lo que no fue necesaria tanta experimentación.

Los experimentos llevados a cabo incluyen:

- Experimentos respecto a la normalización de intensidad con *histogram matching*, de forma análoga a la experimentación con la red T1/T2: entrenamiento - evaluación con las imágenes sin normalización, con normalización HM respecto a 1 sujeto y con normalización HM con el atlas de 52 sujetos.
- Experimentos con distintas funciones de coste: MSE (ecuación 4.4), MAE (Tabla 1.3) y combinación de ambas (ecuación 4.5).
- Imposición de distintos umbrales de correlación para los datos de entrenamiento.

4.5 Validación del método propuesto: Segmentación.

Tal y como se ha mencionado ya previamente, una de las aplicaciones más inmediatas de la síntesis de imagen médica es la segmentación, por lo que finalmente se valida el método propuesto demostrando la mejora que introduce en la precisión de una red U-Net 3D de segmentación del volumen cerebral en 7 tejidos o estructuras anatómicas: líquido cefalorraquídeo (CSF), sustancia blanca (WM), sustancia gris cortical (CGM), sustancia gris subcortical (SCGM), sustancia blanca del cerebelar (CeWM), sustancia gris cerebelar (CeGM) y tronco encefálico (BS).

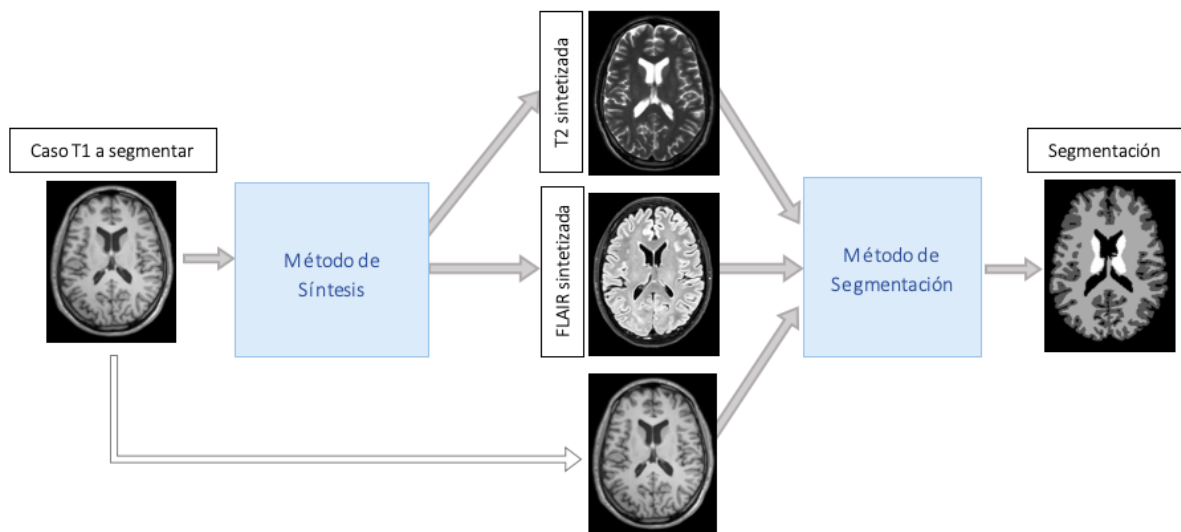


Figura 4.9: Esquema de la metodología de validación del método vía aplicación a segmentación.

Se dispone de una base de datos de 70 volúmenes T1 y sus correspondientes segmentaciones manuales, que se divide en 45 casos para entrenamiento, 5 para validación y 20 para test. Así, se realizan 3 experimentos para demostrar el impacto de la introducción de más modalidades de imagen como entrada a la red de segmentación a parte de la original (T1), siendo éstas las imágenes generadas por nuestras redes de síntesis (T2 y FLAIR):

- **Monomodal:** entrenamiento de la red de segmentación durante 400 *epochs* con un único volumen de entrada T1 y subsiguiente evaluación.
- **Bimodal:** entrenamiento de la red de segmentación durante 400 *epochs* con 2 volúmenes de entrada: T1 original y su correspondiente T2 sintetizado, y subsiguiente evaluación.
- **Trimodal:** entrenamiento de la red de segmentación durante 400 *epochs* con 3 volúmenes de entrada: T1 original y sus correspondientes T2 y FLAIR sintetizados, y subsiguiente evaluación.

Para realizar la evaluación de la red de segmentación, se emplea como métrica el clásico Índice o Coeficiente de Dice (ecuación 4.10), que mide la similitud de dos muestras a partir de la intersección de dichas muestras, siendo éstas en nuestro caso la segmentación *ground-truth* y la

generada por la red. El Índice de Dice es calculado para cada uno de los 7 tejidos de forma individual y finalmente se calcula el valor promedio.

$$S = \frac{\sum(xANDy)}{\sum x + \sum y} \quad (4.10)$$

Siendo x e y la imagen segmentada automáticamente y la imagen *ground-truth* respectivamente.

Resultados y Discusión

En el presente capítulo se presentan y analizan los resultados, tanto cuantitativos como visuales, de los experimentos llevados a cabo con las redes de síntesis T1/T2 y T1/FLAIR, así como los resultados de la validación con la red de segmentación.

5.1 Resultados red T1/T2

Impacto de la función de coste

Se realizaron experimentos con las funciones de coste descritas en el apartado de experimentación, con tal de comprobar con cuál de ellas la red proporcionaba los mejores resultados. Además de la clásica MAE y MSE, se propone una nueva función de coste mixta (ecuación 4.9) que combina el MAE, la correlación (CC) y la diferencia de gradiente (GDL), tal y como se ha explicado previamente.

Así, se entrenó la red empleando cada una de estas tres funciones de coste, y se obtuvieron los resultados mostrados en la Tabla 5.1. Como se puede apreciar en los valores para las 3 métricas de evaluación, la función de coste propuesta, a pesar de proporcionar resultados bastante buenos superando claramente al error absoluto medio (MAE) en todas las métricas, se queda cerca de los resultados del clásico error cuadrático medio (MSE) siendo éstos ligeramente superiores. Por ello, se decidió emplear como función de coste en el modelo final el MSE, siendo los resultados mostrados en la Tabla 5.1 los mejores obtenidos con la red de síntesis de T2.

	MAE		MSE		MIX (propuesta)	
	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>
CC	0.9728	0.9931	0.9740	0.9938	0.9736	0.9934
PSNR	25.8408	31.9231	26.0575	32.3327	25.9996	32.1004
SSIM	0.8419	0.9621	0.8445	0.9634	0.8423	0.9621

Tabla 5.1: Resultados de test de la comparación de las funciones de coste clásicas MAE y MSE, y la función de coste propuesta. Mejores resultados en negrita.

Impacto normalización de intensidad con HM

Durante la experimentación, probamos el impacto de la normalización de intensidad del *dataset* con histogram matching (HM) en los volúmenes sintetizados resultantes. La red mostró una clara mejoría cuando las imágenes habían sido normalizadas en intensidad usando HM (Figura 5.1). Como se puede observar, las imágenes sintetizadas usando HM permiten a la red inferir mejor los detalles de la imagen. Esto podría deberse al hecho de que la estandarización de la intensidad a un espacio canónico común facilita a la red el aprendizaje del mapeo de las intensidades de origen a las de destino, de manera que ésta puede profundizar en la información de alta frecuencia.

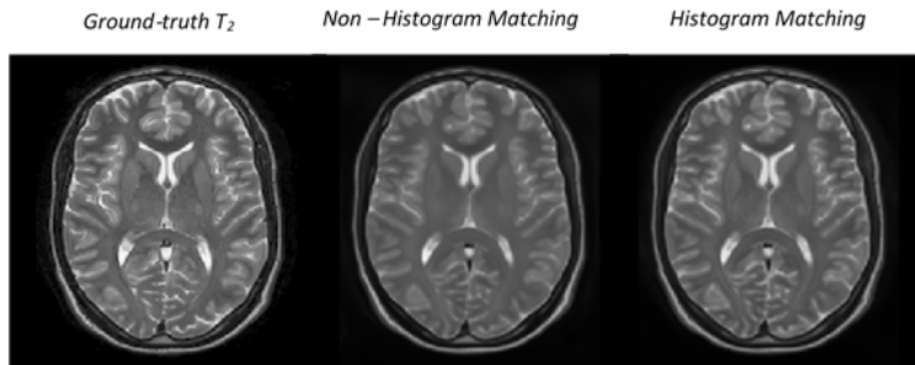


Figura 5.1: Imagen T2 objetivo (izquierda) e imágenes T2 sintetizadas para entrada no normalizada (centro) y entrada normalizada con *histogram matching* (derecha).

Impacto del contexto 3D

Tal y como se ha explicado en el apartado 4.4.3 Experimentos, con el fin de evaluar el impacto de un contexto tridimensional completo, realizamos una comparación de nuestro método 3D propuesto con su correspondiente versión 2D: una UNet 2D estándar con un número de parámetros aprendibles similar.

En la Tabla 5.2, se muestran los resultados de esta comparación. Como se puede observar, el enfoque 2D obtuvo peores resultados para todas las métricas, tanto en *full-head* como en *skull-stripped*, y las imágenes resultantes muestran los artefactos inter-corte esperados a causa de la falta de contexto en la tercera dimensión, tal y como se puede apreciar en la Figura 5.2. Así, queda demostrada la importancia de una perspectiva tridimensional integrada del contexto para la tarea de síntesis de IRM cerebral.

	Método propuesto		UNet 2D		UNet 3D	
	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>
CC	0.974	0.994	0.970	0.991	0.968	0.992
PSNR	26.057	32.333	22.881	26.881	25.203	31.096
SSIM	0.844	0.963	0.784	0.954	0.815	0.955

Tabla 5.2: Resultados de test de la comparación de la red propuesta con la UNet 2D y la UNet 3D básica. Mejores resultados en negrita.

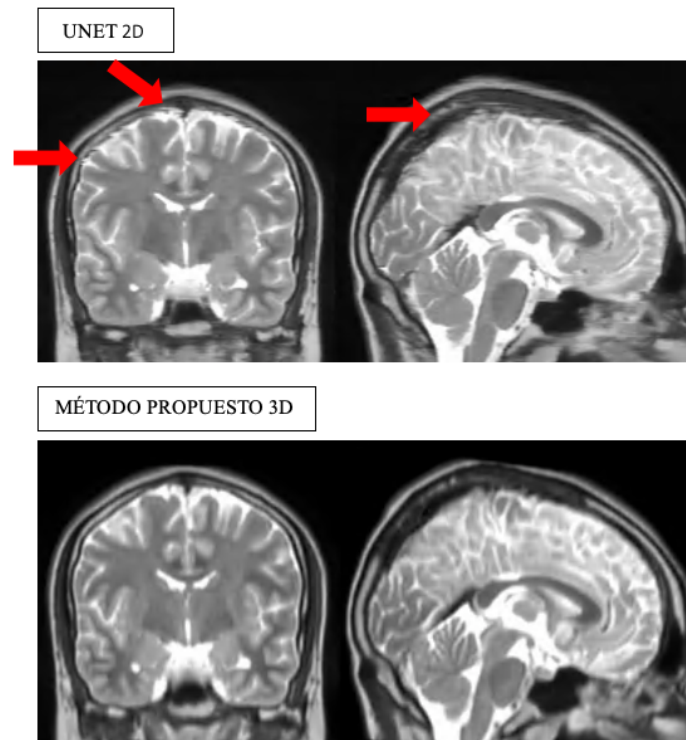


Figura 5.2: Imágenes T2 sintetizadas por la UNet 2D (fila superior) y por el método propuesto (fila inferior). Se aprecian los artefactos inter-corte generados por la UNet 2D, precisamente por la falta de contextualización 3D (flechas rojas).

Impacto de la arquitectura propuesta

De la misma manera, se comparan los resultados de nuestra arquitectura propuesta frente a una UNet 3D básica, sin la descomposición en canales ni el bloque de reconstrucción.

En la Tabla 5.2, se muestran los resultados de esta comparación. Como se puede observar, la UNet 3D obtuvo resultados mejores que su versión 2D, resaltando la importancia de la información del contexto 3D, pero peores que el método propuesto, probablemente debido al menor número de filtros ($nf = 8$, frente a $nf = 40$ con la arquitectura propuesta). Visualmente, esta diferencia de desempeño de ambas redes es observable en las imágenes T2 resultantes. Tal y como se aprecia en la Figura 5.3, a diferencia del método propuesto, la UNet 3D no es capaz de generar los detalles de altas frecuencias de las imágenes por tener un número tan limitado de filtros. Dichos detalles son fundamentales en las imágenes de RM cerebrales, pues constituyen estructuras importantes tales como vasos sanguíneos.

Así, el diseño de la arquitectura de red propuesta ha permitido aumentar considerablemente el número de filtros, y consecuentemente conseguir resultados significativamente mejores, principalmente en referencia al contenido de alta frecuencia.

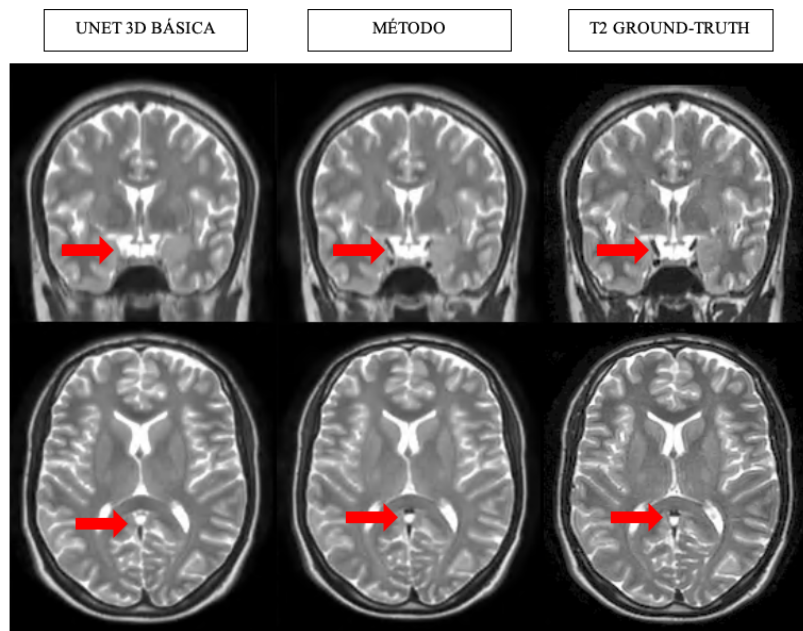


Figura 5.3: Volúmenes T2 sintetizados por una red UNet 3D (fila superior) y por nuestro método (fila inferior). Se aprecia la incapacidad de la UNet 3D de reconstruir detalles finos (flechas rojas).

Comparativa con el estado del arte

Tal y como se ha explicado previamente en el capítulo 4.4.3 de experimentos, se ha realizado una comparativa del método propuesto con dos métodos representativos del estado del arte: REPLICIA y pGAN, ambos enfoques 2D o por *patches* pero basados en técnicas distintas (*Random Forest* y redes adversarias (*Deep Learning*)).

En la Tabla 5.3 se muestran los resultados cuantitativos de la comparación. Como puede apreciarse, el método propuesto supera claramente a los métodos bajo comparación tanto para el volumen completo (*full-head*) como para el volumen exclusivamente cerebral (*skull-stripped*). Sin embargo, se debe aclarar que el método REPLICIA fue entrenado empleando un preprocesado ligeramente diferente en cuanto a la normalización de intensidad se refiere: se normalizaron las imágenes de tal manera que el pico de la sustancia blanca en el histograma se correspondiera con el valor 1, tal y como indican los autores en el artículo original. Por esta razón, los valores de SSIM y PSNR no pueden ser comparados con exactitud, a diferencia de la correlación, que es una medida independiente del rango de intensidad.

En REPLICIA, el volumen sintetizado se genera promediando las salidas individuales de diferentes regresores que realizan un análisis de características en un rango de niveles de resolución. Este funcionamiento conduce a una cierta borrosidad en las imágenes resultantes y un déficit importante a la hora de generar detalles de alta resolución, como se puede observar en la Figura 5.4. Además, como cada píxel en el contraste de salida se predice a partir de *patches* 2D de la imagen de entrada, el volumen construido a partir de las imágenes sintetizadas individualmente

tiende a desatender la coherencia en la tercera dimensión.

Por otra parte, en cuando al método pGAN, la estrategia de entrenamiento adversaria es capaz de crear imágenes más realistas e incluso de replicar el ruido presente en la imagen de entrada, generando imágenes sintetizadas perceptualmente realistas. Sin embargo, a pesar de esto, se pueden apreciar artefactos significativos en el volumen generado (Figura 5.4), como zonas brillantes confundiendo el fondo de la imagen con el LCR. A pesar de generar las imágenes visualmente más realistas de entre los tres métodos, la aparición de estos artefactos resulta en valores bajos en las métricas de evaluación, pues la información estructural en el volumen sintetizado no es completamente fiel a la realidad en el volumen *ground-truth* u original. Esto podría deberse a la falta de una perspectiva general del contexto 3D del volumen completo.

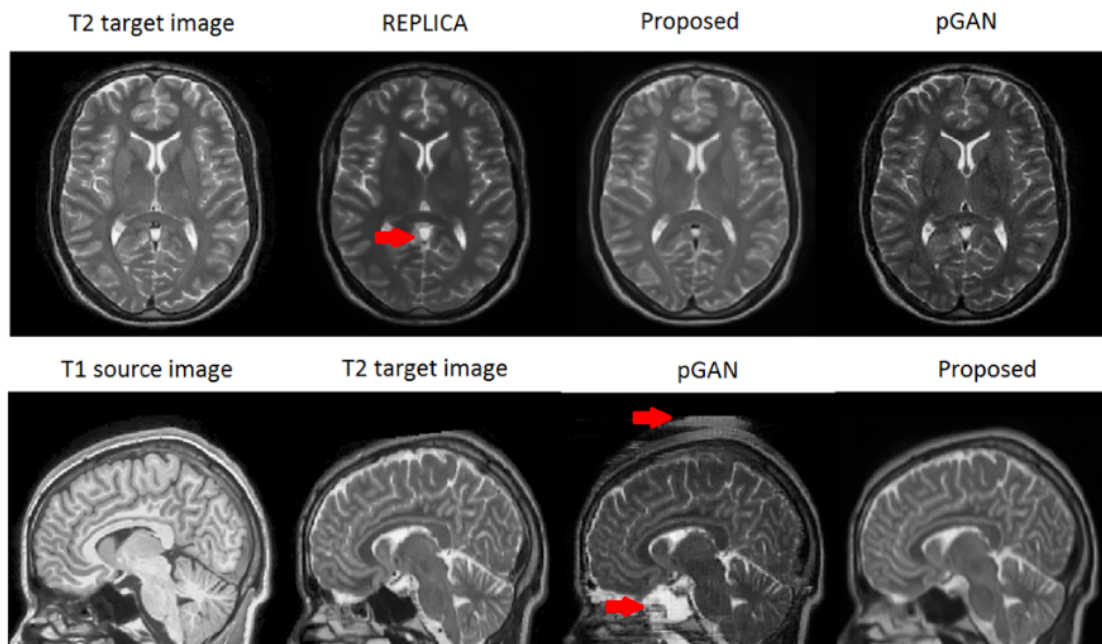


Figura 5.4: Fila superior, de izquierda a derecha: imagen T2 *ground-truth*, imágenes sintetizadas por REPLICA, el método propuesto y pGAN. Fila inferior, de izquierda a derecha: imagen de entrada T1, imagen T2 *ground-truth*, imágenes sintetizadas por pGAN y el método propuesto. Nótese que REPLICA no es capaz de recuperar las pequeñas áreas oscuras en el centro del cerebro y que pGAN introduce artefactos dependientes de los cortes 2D (flechas rojas).

	Método propuesto		REPLICA		pGAN	
	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>
CC	0.974	0.994	0.881	0.906	0.842	0.909
PSNR	26.057	32.333	23.964	27.756	18.147	24.570
SSIM	0.844	0.963	0.694	0.903	0.602	0.893

Tabla 5.3: Resultados de test del método propuesto, REPLICA y pGAN. Mejores resultados en negrita.

Otro aspecto que cabe mencionar es que los resultados del método propuesto son muy coherentes en intensidad, mostrando distribuciones de intensidad acordes al atlas de referencia T2 MNI (Figura 5.4). Esto es así debido al hecho de haber realizado previamente una estandarización de la intensidad a un espacio canónico común con el método de *histogram matching*. Además, las imágenes resultantes del método propuesto no presentan grandes artefactos notables en contraposición a las de REPLICA y pGAN.

Desde un punto de vista de eficiencia, el método propuesto tarda alrededor de 0.5 segundos en predecir un volumen T2 3D en el espacio MNI, mientras que REPLICA tarda aproximadamente 15 minutos y pGAN alrededor de 10 segundos, siendo por tanto el método propuesto el más rápido de entre los tres comparados.

5.2 Resultados red T1/FLAIR

Impacto de la función de coste

Tras haber probado ya en los experimentos del método de síntesis de T2 la ligera superioridad de la función de coste clásica MSE frente a la función de coste propuesta, esta última queda descartada. Así, en los experimentos de síntesis de FLAIR, experimentamos con las siguientes funciones de coste: MAE o norma L1, MSE o norma L2 y una combinación de ambas (ecuación 4.5).

Los resultados obtenidos con cada una de ellas se pueden comparar en la Tabla 5.4. Como se puede apreciar, la función de coste combinada ha proporcionado los mejores resultados en la gran mayoría de las métricas, ligeramente superiores a los obtenidos con el MSE. Se ha observado que con el MAE como función de coste el proceso de optimización del entrenamiento converge más rápido, proporcionando un buen resultado con relativamente pocas *epochs*, pero quedando estancado en un mínimo local sin ser capaz la red de obtener resultados más precisos en siguientes *epochs*. En cambio, con la opción del MSE el aprendizaje es más lento, pero tras un mayor número de *epochs* consigue llegar a un mejor resultado. Un problema al usar el MAE como función de coste es que su gradiente es el mismo para todos sus valores, lo que significa que éste será elevado incluso para valores pequeños de la función de coste, lo que no resulta beneficioso para el aprendizaje. En cambio, el gradiente del MSE es mayor para valores altos de la función de coste y disminuye a medida que ésta se acerca a cero, haciéndola más precisa al final del entrenamiento.

Al combinar ambas medidas como función de coste, se preserva la ventaja de la rapidez inicial del MAE y la ventaja de la precisión final del MSE, proporcionando mejores resultados que cada uno de ellos por separado en un número razonable de *epochs*.

	MAE		MSE		MAE+MSE	
	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>
CC	0.9578	0.9891	0.9629	0.9906	0.9634	0.9908
PSNR	21.6518	27.3854	22.1917	28.1054	22.3147	28.1550
SSIM	0.6624	0.9129	0.6719	0.9259	0.7203	0.9239

Tabla 5.4: Resultados de test de la comparación de las funciones de coste clásicas MAE y MSE, y la combinación de ambas. Mejores resultados en negrita.

Impacto normalización de intensidad con HM

De la misma manera que se hizo para la red de síntesis de T2, se realizó un experimento para comprobar el impacto de la normalización de intensidad con *histogram matching* para la red de síntesis de FLAIR. Se quiso comprobar el efecto de nuevo sobre todo por el hecho de haber normalizado las imágenes FLAIR con el atlas propio generado como referencia en lugar de con uno estandarizado como lo son los MNI T1 y T2.

En la Tabla 5.5 se muestran los resultados de test para la red entrenada y evaluada con las imágenes FLAIR sin normalización HM, con normalización HM con una imagen FLAIR como referencia, y con normalización HM con el atlas FLAIR generado como referencia. Como se puede extraer de la tabla, la normalización con el atlas proporciona los mejores resultados, superando, por una parte, como ya cabía esperar por los resultados obtenidos en la red de T2, a la no normalización y, por otra parte, a la normalización con un único sujeto como referencia. Esto se debe a que el atlas es un promedio de 52 sujetos frente a la imagen de un único sujeto, representando mejor así las distribuciones de intensidades poblacionales de las imágenes FLAIR y, por tanto, permitiendo un mapeo de intensidad vía *histogram matching* a un espacio más estándar y representativo.

Los resultados obtenidos, acordes a los obtenidos en el experimento análogo con la red de síntesis de T2, confirman de nuevo la utilidad de la normalización de intensidad mediante *histogram matching* para la mejora del desempeño de la red de síntesis de IRM.

	No HM		HM (1 sujeto)		HM (atlas 52 sujetos)	
	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>
CC	0.9424	0.9820	0.9547	0.9883	0.9634	0.9908
PSNR	25.1684	27.6016	23.3256	28.3165	22.3147	28.1550
SSIM	0.5796	0.8973	0.7895	0.9196	0.7203	0.9239

Tabla 5.5: Resultados de test de la comparación de la normalización con HM respecto a 1 sujeto, respecto al atlas de 52 sujetos y sin normalización. Mejores resultados en negrita.

Comparación red de síntesis T1/T2 - red de síntesis T1/FLAIR

En referencia a la red de síntesis de T2 frente a la de síntesis de FLAIR, tal y como podemos observar en la Tabla 5.6, esta primera ha obtenido mejores resultados para las métricas de evaluación, y la razón de esto probablemente reside en la distinta naturaleza de los conjuntos de datos. Mientras el *dataset* IXI de T1/T2 resulta ser bastante homogéneo presentando imágenes de sujetos únicamente sanos y adquiridas con únicamente dos escáneres distintos, el *dataset* de FLAIR es bastante más heterogéneo presentando imágenes de sujetos tanto sanos como con lesiones muy diversas, además de adquiridas en múltiples escáneres diferentes. Así, el caso de FLAIR muestra más fielmente ese problema de la heterogeneidad inherente de la IRM debido no solo a la presencia de patologías sino también a los procesos de adquisición, que dificulta el análisis automático. Así, a pesar de haber alcanzado resultados ligeramente inferiores a la red de síntesis de T2, al haber sido entrenada con un *dataset* más diverso, la red de síntesis de FLAIR presentará una mayor capacidad de generalización, característica de valor en los métodos de análisis automático.

	T1/T2		T1/FLAIR	
	<i>Full-head</i>	<i>Skull-stripped</i>	<i>Full-head</i>	<i>Skull-stripped</i>
CC	0.9740	0.9938	0.9634	0.9908
PSNR	26.0575	32.3327	22.3147	28.1550
SSIM	0.8445	0.9634	0.7203	0.9239

Tabla 5.6: Tabla comparativa mejores resultados obtenidos en la síntesis de T2 y en la síntesis de FLAIR.

5.3 Resultados validación. Segmentación

A continuación se presentan e interpretan los resultados obtenidos para los experimentos de validación del método de síntesis de imagen mediante su aplicación a segmentación explicados previamente en el apartado 4.5.

En la Tabla 5.7 podemos observar los resultados de test de los experimentos: los coeficientes de Dice para cada uno de los 7 tejidos cerebrales y su valor promedio en la columna final. Tal y como se puede apreciar, el índice de Dice aumenta a medida que aumentamos las modalidades de entrada a la red, siendo mejores los resultados de la segmentación trimodal que los de la bimodal y éstos últimos a su vez superiores a los de la segmentación monomodal. En la Figura 5.5 podemos observar un caso de segmentación con sus segmentaciones resultantes para entrada monomodal y trimodal. Así, podemos observar también visualmente la mayor precisión de la segmentación trimodal frente a la monomodal, sobre todo en zonas pequeñas de tejido, siendo la trimodal más detallada y cercana a la segmentación *ground-truth*.

La introducción de las imágenes T2 y FLAIR sintetizadas a partir del caso T1 a segmentar como entradas adicionales aportan al método de segmentación diferentes perspectivas de la misma realidad del sujeto, incrementando así la información disponible para llevar a cabo la tarea de

segmentación. Cada una de las modalidades presenta sus puntos fuertes y débiles en cuanto a la distinción de los distintos tejidos cerebrales, de manera que el conjunto multimodal proporciona complementariedad en la información de los tejidos subyacentes.

Como cabía esperar, el método de segmentación se ve beneficiado por la riqueza de la información complementaria de las distintas modalidades de imagen de RM sintetizadas, quedando así demostrada la utilidad del método de síntesis de imágenes de T2 y FLAIR a partir de imágenes T1 para la mejora de la precisión de la segmentación del volumen cerebral.

	Coeficiente de Dice							Promedio
	<i>CSF</i>	<i>CGM</i>	<i>WM</i>	<i>SCGM</i>	<i>CeGM</i>	<i>CeWM</i>	<i>BS</i>	
Monomodal (T1)	0.868	0.943	0.967	0.919	0.953	0.941	0.950	0.9348
Bimodal I (T1, T2)	0.872	0.946	0.970	0.922	0.954	0.946	0.951	0.9374
Trimodal (T1, T2, FLAIR)	0.883	0.949	0.968	0.921	0.956	0.947	0.953	0.9395

Tabla 5.7: Resultados de test de los experimentos con la red de segmentación monomodal, bimodal I y II, y trimodal. Mejores resultados en negrita.

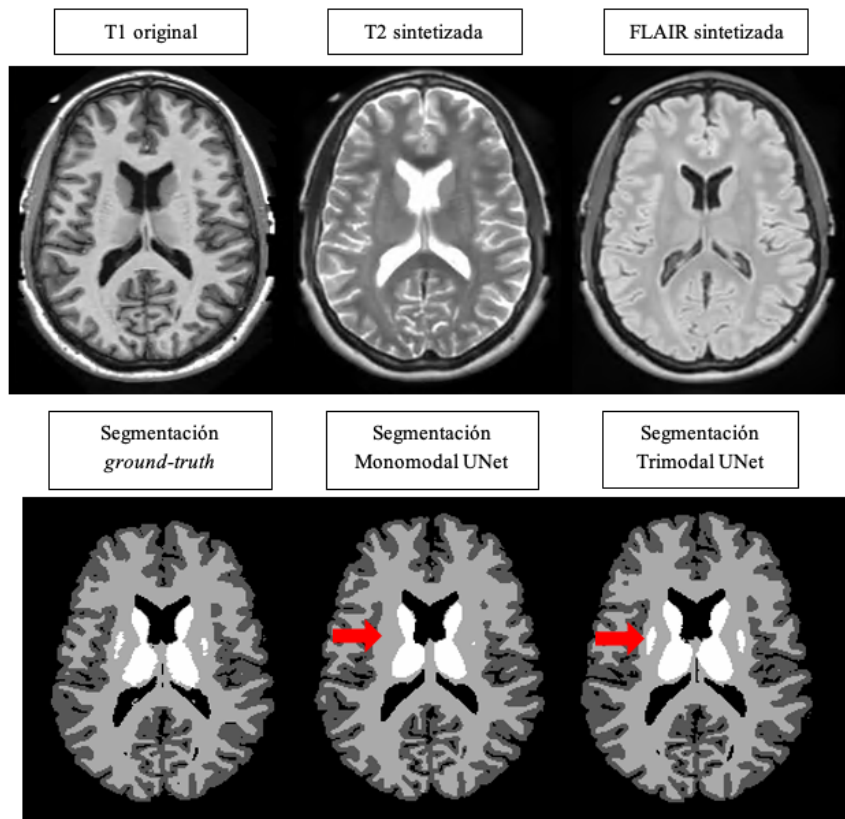


Figura 5.5: Imagen T1 original e imágenes T2 y FLAIR sintetizadas por el método (fila superior, de izquierda a derecha) e imágenes segmentadas *ground-truth*, segmentada con la UNet con entrada monomodal y con entrada trimodal (fila inferior, de izquierda a derecha). Nótese la mayor precisión de la segmentación trimodal en detalles como el indicado por las flechas rojas.

Capítulo 6

Conclusiones

En el presente Trabajo de Fin de Grado, se ha presentado una propuesta basada en *Deep Learning* para llevar a cabo la tarea de síntesis de IRM cerebral, en concreto de imágenes T1 a imágenes T2 y FLAIR. El método propuesto se basa en una variante 3D eficiente desde un punto de vista de memoria de la conocida arquitectura de red U-Net, que permite el procesamiento del volumen cerebral completo de forma simultánea.

El método propuesto ha demostrado superar a dos métodos relevantes del estado del arte basados en cortes 2D o *patches*, REPLICA y pGAN, en términos de precisión y eficiencia. La naturaleza tridimensional de volumen completo del método permite obtener predicciones consistentes sin presencia de artefactos típicos de un enfoque por cortes, proporcionando resultados mejorados gracias a una maximización de la información del contexto.

Por otra parte, se ha obtenido una mejora notable de la precisión en los resultados empleando una normalización de intensidad adicional vía *histogram matching*. Pensamos que el uso de esta normalización aumenta la capacidad de generalización de la red al mapear a un espacio de intensidad canónico y que puede reducir potencialmente los problemas de adaptación de dominios al usar imágenes de diferentes escáneres o secuencias de adquisición.

Finalmente, ha quedado demostrada la utilidad del método en una aplicación a segmentación, mejorando los resultados al incluir como imágenes de entrada extra a la T1 a segmentar, sus correspondientes imágenes T2 y FLAIR sintetizadas con la red propuesta, aportando así información complementaria que ha mostrado ser útil para la crucial tarea de la segmentación automática del cerebro.

Parte II

Presupuesto

Presupuesto

6.1 Introducción

En el presente documento del Trabajo de Fin de Grado se presenta una valoración económica del proyecto llevado a cabo.

En cuanto a la mano de obra para la realización del proyecto, ha sido necesario un ingeniero biomédico *junior* (todavía estudiante), que lleva a cabo la mayor parte del trabajo, y un ingeniero superior (el tutor), que se encarga de supervisar y dirigir el trabajo del primero. Se ha considerado un sueldo base mensual de 1.200 € para el ingeniero *junior* y de 2.500 € para el superior (según la información proporcionada por el servicio de recursos humanos de la Universitat Politècnica de València), con dos pagas extra anuales. Respecto al coste de seguridad social de los trabajadores, se han considerado los siguientes porcentajes acordes a la información del Ministerio de Empleo y Seguridad Social: 23,60 % en contingencias comunes, 5,50 % de seguro de desempleo, 0,20 % de FOGASA, 0,6 % de formación profesional y 1,65 % de accidentes de trabajo y enfermedades profesionales. Si se considera una jornada laboral de 8 horas y se descuentan 15 días festivos, fines de semana y 30 días de vacaciones, el número total de horas de trabajo asciende a 1.792 horas/año. Así, se deduce que el coste/hora es de 12,33 €/h para el ingeniero *junior* y 25,69 €/h para el superior.

Cabe mencionar que para los costes referentes al *hardware* y a las licencias de los programas de *software* empleados, se ha considerado la amortización correspondiente a la duración del proyecto en lugar del precio de venta completo (obtenidos de las páginas web oficiales de cada uno de los programas). Los factores de amortización se han calculado como el ratio entre la duración del trabajo (mas concretamente una aproximación del periodo de uso de cada recurso) y la vida útil estimada de cada uno de los elementos. Para el *hardware* (ordenador y GPU) se ha considerado una amortización a 5 años (60 meses), para el sistema operativo Windows 10 Enterprise se ha considerado como periodo de amortización 4 años (48 meses) hasta su obsolescencia, y con respecto al *software* Matlab R2019b, se trata de una licencia anual (12 meses). El resto de programas empleados son gratuitos, por lo que no han aumentado el coste del proyecto. Por último mencionar que el tiempo de utilización de hardware y software se ha agregado al completo en la

fase primera del proyecto por simplificar.

Por otra parte, mencionar que para el cálculo del presupuesto por contrata se ha aplicado un incremento del presupuesto de ejecución material en un 13 % en materia de gastos generales y en un 6 % de beneficio industrial. Sobre esta cifra, se ha aplicado finalmente el IVA correspondiente del 21 % (Agencia Tributaria) para obtener la cifra final de presupuesto por contrata.

6.2 Cuadro de Mano de Obra

Nº	Código	Denominación de la mano de obra	Precio	Horas	Total
1	MO.IBJ	Ingeniero biomédico <i>junior</i>	12,33€/h	429 h	5.289,57€
2	MO.IS	Ingeniero superior (tutor)	25,69€/h	49 h	1.258,81€
Total mano de obra:					6.548,38€

Tabla 6.1: Tabla de precios de mano de obra

6.3 Cuadro de materiales, equipos y *software*

Nº	Código	Denominación del material/equipo/software	Precio (€/u)	Cantidad	Factor Amortiz.	Total (€)
1	MES.Or	Ordenador de sobremesa	1000,00	1 u	9/60	150,00
2	MES.GPU	Tarjeta Gráfica	780,00	1 u	7/60	91,00
3	MES.W	Microsoft Windows 10 Enterprise	259,00	1 u	9/48	48,56
4	MES.Mat	Matlab R2019b	250,00	1 u	2/12	41,67
5	MES.ITK	ITK-SNAP 3.6	0,00	1 u	-	0,00
6	MES.MRI	MRIero	0,00	1 u	-	0,00
7	MES.ANT	Advanced Normalization Tools ANTs	0,00	1 u	-	0,00
8	MES.SPM	Statistical Parametric Mapping SPM	0,00	1 u	-	0,00
9	MES.An	Anaconda 3.7	0,00	1 u	-	0,00
10	MES.OL	OverLeaf (Editor de LaTeX)	0,00	1 u	-	0,00
Total materiales, equipos y software:						331,23€

Tabla 6.2: Tabla de precios de materiales, equipos y *software*.

6.4 Cuadro de presupuestos parciales

1. Planificación del proyecto y formación en conocimientos teóricos y estado del arte

Nº	Ud	Denominación	Cantidad	Precio (€)	Total(€)
1.1	h	Reunión inicial para presentar y planificar el proyecto.	1	38,02	38,02
1.2	h	Revisión de la literatura científica sobre síntesis de imagen en IRM y teoría de Deep Learning y CNNs.	15	12,33	184,95
1.3	u	Instalación y aprendizaje de uso de programas software.	1	567,56	567,56
Total presupuesto parcial nº1:					790,53€

Tabla 6.3: Tabla de presupuestos parciales del Capítulo 1.

2. Desarrollo del método de síntesis de IRM.

Nº	Ud	Denominación	Cantidad	Precio (€)	Total(€)
2.1	u	Diseño y aplicación del preprocesado de los datos.	1	236,33	236,33
2.2	u	Diseño, implementación y experimentación con la red.	1	4.110,10	4.110,10
Total presupuesto parcial nº2:					4.346,43€

Tabla 6.4: Tabla de presupuestos parciales del Capítulo 2.

3. Validación del método mediante su aplicación a segmentación.

Nº	Ud	Denominación	Cantidad	Precio (€)	Total(€)
3.1	u	Elección del método de segmentación y preparación de los datos.	1	62,68	62,68
3.2	h	Realización de los experimentos de segmentación.	10	12,33	123,30
Total presupuesto parcial nº3:					185,98€

Tabla 6.5: Tabla de presupuestos parciales del Capítulo 3.

4. Redacción y defensa del TFG

Nº	Ud	Denominación	Cantidad	Precio (€)	Total(€)
4.1	h	Redacción de los documentos del trabajo.	100	12,33	1.233,00
4.2	u	Revisión y corrección de errores.	1	380,20	380,20
4.3	h	Preparación de la defensa del trabajo.	10	12,33	123,30
Total presupuesto parcial nº4:					1.736,50€

Tabla 6.6: Tabla de presupuestos parciales del Capítulo 4.

6.5 Cuadro de precios unitarios

Nº	Denominación	Importe	
		En cifra (€)	En letra (€)
	1. PLANIFICACIÓN DEL PROYECTO Y FORMACIÓN EN CONOCIMIENTOS TEÓRICOS Y ESTADO DEL ARTE		
1.1	h Reunión inicial para presentar y planificar el proyecto.	38,02	Treinta y ocho euros con dos céntimos
1.2	h Revisión de la literatura científica sobre síntesis de imagen en IRM y teoría de Deep Learning y CNNs.	12,33	Doce euros con treinta y tres céntimos
1.3	u Instalación y aprendizaje de uso de programas software.	567,56	Quinientos sesenta y siete euros con cincuenta y seis céntimos
	2. DESARROLLO DEL MÉTODO DE SÍNTESIS DE IRM.		
2.1	u Diseño y aplicación del preprocesado de los datos	236,33	Doscientos treinta y seis euros con treinta y tres céntimos
2.2	u Diseño, implementación y experimentación con la red.	4110,10	Cuatro mil ciento diez euros con diez céntimos
	3. VALIDACIÓN DEL MÉTODO EN SEGMENTACIÓN.		
3.1	u Elección del método de segmentación y preparación de los datos.	62,68	Sesenta y dos euros con sesenta y ocho céntimos
3.2	h Realización de los experimentos de segmentación.	12,33	Doce euros con treinta y tres céntimos
	4. REDACCIÓN Y DEFENSA DEL TFG		
4.1	h Redacción de los documentos del trabajo.	12,33	Doce euros con treinta y tres céntimos
4.2	u Revisión y corrección de errores.	380,20	Trescientos ochenta euros con veinte céntimos
4.3	h Preparación de la defensa del trabajo.	123,30	Ciento veintitrés euros con treinta céntimos

Tabla 6.7: Tabla de precios unitarios.

6.6 Cuadro de precios descompuestos

Nº	Ud	Denominación		Total
1. PLANIFICACIÓN DEL PROYECTO Y FORMACIÓN EN CONOCIMIENTOS TEÓRICOS Y ESTADO DEL ARTE				
1.1	h	Reunión inicial para presentar y planificar el proyecto.		
	MO.IBJ	1 h	Ingeniero biomédico <i>junior</i>	12,33 €/h 12,33€
	MO.IS	1 h	Ingeniero superior	25,69€/h 25,69€
		Precio total por h:		38,02€
1.2	h	Revisión de la literatura científica sobre síntesis de imagen en IRM y teoría de Deep Learning y CNNs.		
	MO.IBJ	1 h	Ingeniero biomédico <i>junior</i>	12,33 €/h 12,33€
		Precio total por h:		12,33€
1.3	u	Instalación y aprendizaje de uso de programas software.		
	MO.IBJ	15 h	Ingeniero biomédico <i>junior</i>	12,33 €/h 184,95€
	MO.IS	2 h	Ingeniero superior	25,69€/h 51,38€
	MES.Or	1 u 9/60	Ordenador de sobremesa	1000,00€ 150€
	MES.GPU	1 u 7/60	Tarjeta Gráfica	780,00€ 91€
	MES.W	1 u 9/48	Microsoft Windows 10 Enterprise	259,00€ 48,56€
	MES.Mat	1 u 2/12	Matlab R2019b	250,00€ 41,67€
	MES.ITK	1 u -	ITK-SNAP 3.6	0,00€ 0,00€
	MES.MRI	1 u -	MRIcro	0,00€ 0,00€
	MES.ANT	1 u -	Advanced Normalization Tools (ANTs)	0,00€ 0,00€
	MES.SPM	1 u -	Statistical Parametric Mapping (SPM)	0,00€ 0,00€
	MES.An	1 u -	Anaconda 3.7	0,00€ 0,00€
	MES.OL	1 u -	OverLeaf (Editor de LaTeX)	0,00€ 0,00€
		Precio total por u:		567,56€
2. DESARROLLO DEL MÉTODO DE SÍNTESIS DE IRM.				
2.1	u	Diseño y aplicación del preprocesado de los datos		
	MO.IBJ	15 h	Ingeniero biomédico <i>junior</i>	12,33 €/h 12,33€
	MO.IS	2 h	Ingeniero superior	25,69€/h 25,69€
		Precio total por u:		236,33€
2.2	u	Diseño, implementación y experimentación con la red.		
	MO.IBJ	250 h	Ingeniero biomédico <i>junior</i>	12,33 €/h 12,33€
	MO.IS	40 h	Ingeniero superior	25,69€/h 1027,6€
		Precio total por u:		4110,10€
3. VALIDACIÓN DEL MÉTODO EN SEGMENTACIÓN.				
3.1	u	Elección del método de segmentación y preparación de los datos.		
	MO.IBJ	3 h	Ingeniero biomédico <i>junior</i>	12,33 €/h 36,99€
	MO.IS	1 h	Ingeniero superior	25,69€/h 25,69€
		Precio total por u:		62,68€
3.2	h	Realización de los experimentos de segmentación.		
	MO.IBJ	1 h	Ingeniero biomédico <i>junior</i>	12,33 €/h 12,33€
		Precio total por h:		12,33€

Nº	Ud	Denominación	Total
4. REDACCIÓN Y DEFENSA DEL TFG			
4.1	h	Redacción de los documentos del trabajo.	
		MO.IBJ 1 h Ingeniero biomédico <i>junior</i>	12,33 €/h 12,33€
		Precio total por h:	12,33€
4.2	u	Revisión y corrección de errores.	
		MO.IBJ 10 h Ingeniero biomédico <i>junior</i>	12,33 €/h 123,30€
		MO.IS 3 h Ingeniero superior	25,69€/h 256,90€
		Precio total por u:	380,20€
4.3	h	Preparación de la defensa del trabajo.	
		MO.IBJ 1 h Ingeniero biomédico <i>junior</i>	12,33 €/h 123,30€
		Precio total por h:	123,30€

Tabla 6.8: Tabla de precios descompuestos.

6.7 Presupuesto de ejecución por contrata

Capítulo	Importe (€)
Capítulo 1. Planificación del proyecto y formación en conocimientos teóricos y estado del arte.	790,53€
Capítulo 2. Desarrollo del método de síntesis de IRM.	4.346,43€
Capítulo 3. Validación del método en segmentación.	185,98€
Capítulo 4. Redacción y defensa del TFG.	1.736,50€
Presupuesto de ejecución material	7.059,44€
Gastos generales (13 %)	917,73€
Beneficio industrial (6 %)	423,57€
Suma	8.400,74€
IVA (21 %)	1.764,16€
Presupuesto de ejecución por contrata:	10.164,90€

Tabla 6.9: Tabla de presupuesto de ejecución por contrata.

El presupuesto de ejecución por contrata asciende a la cantidad de **diez mil ciento sesenta y cuatro euros con noventa céntimos**.

Bibliografía

- Agarap, Abien Fred (2018). “Deep learning using rectified linear units (relu)”. En: *arXiv preprint arXiv:1803.08375* (vid. pág. 22).
- Avants, Brian B y col. (2008). “Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain”. En: *Medical image analysis* 12.1, págs. 26-41 (vid. pág. 39).
- Ayachi, Riadh y col. (2018). “Strided convolution instead of max pooling for memory efficiency of convolutional neural networks”. En: *International conference on the Sciences of Electronics, Technologies of Information and Telecommunications*. Springer, págs. 234-243 (vid. pág. 49).
- Bottou, Léon (2010). “Large-scale machine learning with stochastic gradient descent”. En: *Proceedings of COMPSTAT'2010*. Springer, págs. 177-186 (vid. pág. 25).
- Breiman, Leo (2001). “Random forests”. En: *Machine learning* 45.1, págs. 5-32 (vid. págs. 16, 55).
- Brownlee, Jason (ago. de 2020). *Understand the Impact of Learning Rate on Neural Network Performance*. URL: <https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/>. [Accessed: 17.06.2020] (vid. pág. 25).
- Buades, Antoni, Bartomeu Coll y Jean-Michel Morel (2011). “Non-local means denoising”. En: *Image Processing On Line* 1, págs. 208-212 (vid. pág. 42).
- Burgos, Ninon y col. (2014). “Attenuation correction synthesis for hybrid PET-MR scanners: application to brain studies”. En: *IEEE transactions on medical imaging* 33.12, págs. 2332-2341 (vid. pág. 16).
- Chartsias, Agisilaos y col. (2017). “Multimodal MR synthesis via modality-invariant latent representation”. En: *IEEE transactions on medical imaging* 37.3, págs. 803-814 (vid. pág. 15).

- Chris Rorden's Neuropsychology Lab, University of South Carolina (s.f.). *MRICro Software*. URL: <https://people.cas.sc.edu/rorden/mricro/mricro.html> (vid. pág. 39).
- Convolutional Neural Networks (LeNet)* (jun. de 2018). URL: <http://deeplearning.net/tutorial/lenet.html>. [Accessed: 25.06.2020] (vid. pág. 29).
- Coupé, Pierrick y col. (2011). "Patch-based segmentation using expert priors: Application to hippocampus and ventricle segmentation". En: *NeuroImage* 54.2, págs. 940-954 (vid. pág. 18).
- Craig, John A, James Perkins y Frank H Netter (2002). *Atlas of neuroanatomy and neurophysiology: selections from the Netter Collection of medical illustrations*. Icon Custom Communications (vid. pág. 14).
- Dar, Salman UH y col. (2019). "Image synthesis in multi-contrast MRI with conditional generative adversarial networks". En: *IEEE transactions on medical imaging* 38.10, págs. 2375-2388 (vid. págs. 15, 16, 55).
- Deshpande, Mohit (2017). *Perceptrons: The First Neural Networks*. URL: <https://pythonmachinelearning.pro/perceptrons-the-first-neural-networks/>. [Accessed: 05.06.2020] (vid. pág. 21).
- Despotović, Ivana, Bart Goossens y Wilfried Philips (2015). "MRI segmentation of the human brain: challenges, methods, and applications". En: *Computational and mathematical methods in medicine* 2015 (vid. pág. 18).
- Dou, Qi y col. (2017). "3D deeply supervised network for automated segmentation of volumetric medical images". En: *Medical image analysis* 41, págs. 40-54 (vid. págs. 18, 32).
- Fonov, Vladimir S y col. (2009). "Unbiased nonlinear average age-appropriate brain templates from birth to adulthood". En: *NeuroImage* 47, S102 (vid. pág. 37).
- Gili, Jaume (1993). "Introducción biofísica a la resonancia magnética". En: *Centre Diagnòstic Pedralbes* (vid. págs. 6-11).
- Gonzales, RC y BA Fittes (1977). "Gray-level transformations for interactive image enhancement". En: *Mechanism and Machine Theory* 12.1, págs. 111-122 (vid. pág. 46).
- Gonzalez, Rafael C, Richard Eugene Woods y Steven L Eddins (2004). *Digital image processing using MATLAB*. Pearson Education India (vid. pág. 46).
- Hochreiter, Sepp y col. (2001). *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies* (vid. pág. 22).
- Hore, Alain y Djemel Ziou (2010). "Image quality metrics: PSNR vs. SSIM". En: *2010 20th international conference on pattern recognition*. IEEE, págs. 2366-2369 (vid. pág. 52).

- Huang, Yawen, Ling Shao y Alejandro F Frangi (2017). “Cross-modality image synthesis via weakly coupled and geometry co-regularized joint dictionary learning”. En: *IEEE transactions on medical imaging* 37.3, págs. 815-827 (vid. págs. 15, 16).
- Hussain, Altaf y col. (dic. de 2018). *Keras Conv2D and Convolutional Layers*. URL: <https://www.pyimagesearch.com/2018/12/31/keras-conv2d-and-convolutional-layers/>. [Accessed: 25.06.2020] (vid. pág. 28).
- Hutson, Matthew (2018). “Has artificial intelligence become alchemy?” En: *Science* 360.6388, págs. 478-478. ISSN: 0036-8075. DOI: [10.1126/science.360.6388.478](https://doi.org/10.1126/science.360.6388.478). eprint: <https://science.sciencemag.org/content/360/6388/478.full.pdf>. URL: <https://science.sciencemag.org/content/360/6388/478> (vid. pág. 25).
- Iglesias, Juan Eugenio, Ender Konukoglu y col. (2013). “Is synthesizing MRI contrast useful for inter-modality analysis?” En: págs. 631-638 (vid. pág. 15).
- Iglesias, Juan Eugenio y Mert R Sabuncu (2015). “Multi-atlas segmentation of biomedical images: a survey”. En: *Medical image analysis* 24.1, págs. 205-219 (vid. págs. 17, 18).
- Intuitive understanding of 1D, 2D, and 3D convolutions in convolutional neural networks* (ago. de 1966). URL: <https://stackoverflow.com/questions/42883547/intuitive-understanding-of-1d-2d-and-3d-convolutions-in-convolutional-neural-n/44628011> (vid. pág. 33).
- IXI Dataset* (s.f.). URL: <https://brain-development.org/ixi-dataset/> (vid. pág. 37).
- Jog, Amod, Aaron Carass y col. (2015). “MR image synthesis by contrast learning on neighborhood ensembles”. En: *Medical image analysis* 24.1, págs. 63-76 (vid. págs. 15, 16).
- (2017). “Random forest regression for magnetic resonance image synthesis”. En: *Medical image analysis* 35, págs. 475-488 (vid. págs. 15, 16, 39, 55).
- Jog, Amod, Snehashis Roy y col. (2013). “Magnetic resonance image synthesis through patch regression”. En: *2013 IEEE 10th International Symposium on Biomedical Imaging*. IEEE, págs. 350-353 (vid. págs. 15, 16).
- Johns Hopkins Medicine (s.f.). *Anatomy of the Brain*. URL: <https://www.hopkinsmedicine.org/health/conditions-and-diseases/anatomy-of-the-brain>. [Accessed: 10.05.2020] (vid. pág. 13).
- Joyce, Thomas, Agisilaos Chertsias y Sotirios A Tsaftaris (2017). “Robust multi-modal MR image synthesis”. En: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, págs. 347-355 (vid. pág. 15).
- Kingma, Diederik P y Jimmy Ba (2014). “Adam: A method for stochastic optimization”. En: *arXiv preprint arXiv:1412.6980* (vid. págs. 25, 50).

- Las redes neuronales* (2006). URL: <https://worldbenfergroup.wordpress.com/2016/04/03/las-redes-neuronales/>. [Accessed: 15.06.2020] (vid. pág. 21).
- LeCun, Yann, Yoshua Bengio y Geoffrey Hinton (2015). “Deep learning”. En: *nature* 521.7553, págs. 436-444 (vid. pág. 20).
- Lee, Chen-Yu y col. (2015). “Deeply-supervised nets”. En: *Artificial intelligence and statistics*, págs. 562-570 (vid. pág. 32).
- Lladó, Xavier y col. (2012). “Segmentation of multiple sclerosis lesions in brain MRI: a review of automated approaches”. En: *Information Sciences* 186.1, págs. 164-185 (vid. pág. 12).
- Manjón, José V y Pierrick Coupé (2016). “volBrain: an online MRI brain volumetry system”. En: *Frontiers in neuroinformatics* 10, pág. 30 (vid. pág. 38).
- Manjón, José V, Pierrick Coupé y col. (2010). “Adaptive non-local means denoising of MR images with spatially varying noise levels”. En: *Journal of Magnetic Resonance Imaging* 31.1, págs. 192-203 (vid. pág. 42).
- Mayo Clinic (s.f.). *Brain lobes*. URL: <https://www.mayoclinic.org/brain-lobes/img-20008887>. [Accessed: 10.05.2020] (vid. pág. 13).
- Miller, Michael I y col. (1993). “Mathematical textbook of deformable neuroanatomies”. En: *Proceedings of the National Academy of Sciences* 90.24, págs. 11944-11948 (vid. pág. 16).
- Nie, Dong y col. (2018). “Medical image synthesis with deep convolutional adversarial networks”. En: *IEEE Transactions on Biomedical Engineering* 65.12, págs. 2720-2730 (vid. págs. 15, 16).
- NIfTI: - Neuroimaging Informatics Technology Initiative* (s.f.). URL: <https://nifti.nih.gov/> (vid. pág. 37).
- OpenNN, Open Neural Networks (s.f.). URL: <https://www.opennn.net/>. [Accessed: 17.06.2020] (vid. pág. 23).
- Parmar, Ravindra (sep. de 2018). *Training Deep Neural Networks*. URL: <https://towardsdatascience.com/training-deep-neural-networks-9fdb1964b964>. [Accessed: 26.06.2020] (vid. pág. 31).
- Rinck, Peter A (2019). *Magnetic resonance in medicine: a critical introduction*. BoD—Books on Demand (vid. pág. 3).
- Ronneberger, Olaf, Philipp Fischer y Thomas Brox (2015). “U-net: Convolutional networks for biomedical image segmentation”. En: págs. 234-241 (vid. págs. 18, 30, 31, 34).

- Roy, Snehashis, Aaron Carass y Jerry Prince (2011). “A compressed sensing approach for MR tissue contrast synthesis”. En: *Biennial International Conference on Information Processing in Medical Imaging*. Springer, págs. 371-383 (vid. págs. 15, 16).
- Saha, Sumit (dic. de 2018). *A Comprehensive Guide to Convolutional Neural Networks-the ELI5 way*. URL: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. [Accessed: 25.06.2020] (vid. pág. 29).
- School of Medicine, Case Western Reserve University (2006). *Magnetic Resonance Imaging (MRI) of the Brain and Spine: Basics*. URL: <https://casemed.case.edu/clerkships/neurology/Web%5C%20Neurorad/MRI%5C%20Basics.htm>. [Accessed: 08.05.2020] (vid. pág. 12).
- SPM - Statistical Parametric Mapping (s.f.). URL: <https://www.fil.ion.ucl.ac.uk/spm/> (vid. pág. 39).
- TensorFlow Conv2D Layers: A Practical Guide (s.f.). URL: <https://missinglink.ai/guides/tensorflow/tensorflow-conv2d-layers-practical-guide/>. [Accessed: 25.06.2020] (vid. pág. 28).
- Tustison, Nicholas J y col. (2010). “N4ITK: improved N3 bias correction”. En: *IEEE transactions on medical imaging* 29.6, págs. 1310-1320 (vid. pág. 39).
- Van Nguyen, Hien, Kevin Zhou y Raviteja Vemulapalli (2015). “Cross-domain synthesis of medical images using efficient location-sensitive deep network”. En: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, págs. 677-684 (vid. pág. 15).
- Wang, Zhou y col. (2004). “Image quality assessment: from error visibility to structural similarity”. En: *IEEE transactions on image processing* 13.4, págs. 600-612 (vid. pág. 52).
- Yushkevich, Paul A. y col. (2006). “User-Guided 3D Active Contour Segmentation of Anatomical Structures: Significantly Improved Efficiency and Reliability”. En: *Neuroimage* 31.3, págs. 1116-1128. URL: www.itksnap.org (vid. pág. 39).