



ugr

Universidad
de Granada

TRABAJO FIN DE GRADO
INGENIERÍA INFORMÁTICA

Estimación de la calidad de imágenes médicas 3D

Aprendizaje automático y Aprendizaje profundo

Autor

Brian Sena Simons

Directores

Pablo Mesejo Santiago

Enrique Bermejo Nievas



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE
TELECOMUNICACIÓN

Granada, mes de 201

Anotaciones y Cambios

Índice general

1. Introducción	9
1.1. Definición del Problema	9
1.2. Motivación	11
1.3. Objetivos	13
1.4. Planificación del proyecto	13
2. Fundamentos Teóricos	17
2.0.1. Subproblemas de la IQA	17
2.1. Aprendizaje Automático	20
2.2. Aprendizaje Profundo	21
2.2.1. Redes Convolucionales	23
2.2.2. Capas convolucionales	23
2.2.3. Capa de pooling	24
2.2.4. Capas totalmente conectadas	25
2.2.5. Aplicadas a Videos	25
2.2.6. Aplicadas a nubes de puntos	26
2.3. Ensemble o Conjunto <i>Deep Learning</i>	27
2.4. Imágenes médicas y distorsiones	28
3. Estado del Arte	31
4. Materiales, Métodos y Distorsiones	33
4.1. Materiales	33
4.2. Métodos	33
4.3. Distorsiones	33
5. Implementación y Experimentos	35
5.1. Detalles Técnicos de Implementación	35
5.2. Experimentos	35
6. Conclusiones y Trabajos Futuros	37
Bibliografía	44

Índice de figuras

1.1.	Ejemplo de visualización de un directorio <i>DICOM</i> . Se pueden observar las proyecciones axial, coronal y sagital. Además de una renderización volumétrica de los huesos. Para ello se ha utilizado <i>Slicer3D</i> [1]	12
2.1.	En este ejemplo, extraído de [2], vemos que sumar una constante positiva a una imagen de referencia (a) produce la imagen (b) que contiene la misma distancia <i>Minkowski</i> que (c), imagen fabricada por la misma constante pero permutando signo de forma aleatoria. Siendo que la percepción final es que la imagen (c) es peor que la imagen (b).	19
2.2.	En este ejemplo, extraído de [3], la misma imagen distorsionada de distintas maneras resulta en el mismo valor <i>MSE</i> =181. Siendo evidente que algunas distorsiones producen efectos visuales más marcados que otras.	19
2.3.	Ejemplo de aprendizaje supervisado, vemos como a partir de un conjunto de clases etiquetadas aprendemos un hiper plano que las separa.	20
2.4.	Ejemplo de aprendizaje no supervisado, dado un conjunto de puntos aprendemos un conjunto de clases a partir de los patrones.	21
2.5.	Ejemplo gráfico de una red neuronal [4–6]. En (a) vemos una neurona con su representación artificial simplificada (b). En (c) vemos una conexión entre neuronas y en (d) la representación de diferentes profundidades de conexiones artificiales.	22
2.6.	Ejemplo extraída de [7] del proceso de convolución sobre una radiografía pulmonar para la detección de enfermedades.	23
2.7.	Representación visual de la operación de convolución sobre una imagen, extraída de [8].	24
2.8.	Ejemplo de operación de <i>max-pooling</i> con <i>stride</i> a 2.	25
2.9.	Ejemplo extraído de [9] para ilustrar la distinción entre la extracción espacial del camino “ <i>slow path</i> ” y de movimiento con el “ <i>fast path</i> ”.	26

2.10. Ejemplo extraído de [10] que demuestra el resultado de transformar una nube de puntos sin estructura en una cuadrícula voxelizada.	26
2.11. Representación de métodos de <i>ensemble</i> extraído de [11] . . .	27
2.12. Ejemplo de tomografía computarizada y modelo 3D [12] . . .	28
2.13. Ejemplo de artefactos sobre imágenes Dicom [13]: (a) un artefacto de anillo, (b) un difuminado provocado por movimiento, (c) artefactos producidos por interpolación helicoidal, (d) artefactos de iluminación y dispersión del haz.	29
2.14. Ejemplo de distorsiones generadas. Leyendo de izquierda a derecha tenemos: la imagen original, compresión <i>octree</i> , reducción de puntos por <i>uniform downsampling</i> , movimiento local, rotación local y ruido gaussiano.	30

Índice de tablas

1.1. Planificación temporal inicial del proyecto	14
1.2. Planificación resultante del proyecto	14
1.3. Total de horas y días trabajados	15
1.4. Estimación final de coste del proyecto	15

Capítulo 1

Introducción

1.1. Definición del Problema

Con la demanda incremental de aplicaciones, tanto para el entretenimiento como para el estudio biomédico, la información visual cada vez tiene un rol más importante. Sin embargo, la calidad de la información puede sufrir drásticamente con las etapas de adquisición, procesado, compresión, transmisión y reproducción. Es por ello que poder evaluar la calidad de la información se ha vuelto un tema cada vez más importante [14].

La rama sobre la cuál nos centraremos es la evaluación de la calidad de la imagen, en inglés *Image Quality assessment (IQA)*. Es un problema fundamental en el procesamiento de imágenes y de visión por computador. Se refiere a la tarea de medir y cuantificar la calidad perceptual de una imagen, teniendo en cuenta factores como el contenido, la resolución, el contraste, las distorsiones visuales y la percepción humana. La mejora de las técnicas suele estar altamente conectado con el avance de los estudio del sistema de visión humano [3].

El problema de la evaluación de la calidad de la imagen se aborda mediante enfoques subjetivos y objetivos. Los enfoques subjetivos implican realizar experimentos perceptuales en los que se recopilan las opiniones y evaluaciones de los observadores humanos. Estos observadores pueden calificar las imágenes en términos de su calidad visual o realizar comparaciones entre diferentes versiones de una misma imagen. Con base a las respuestas recopiladas, se pueden establecer modelos y métricas que reflejen la calidad percibida por los humanos, también conocida como *mean opinion score*, *MOS*¹.

Alternativamente, los enfoques objetivos buscan desarrollar algoritmos y

¹ *Mean Opinion Score* o Valor medio de opinión, consiste en la media de la opinión de diversas personas para establecer un valor de referencia.

métricas automáticas que puedan estimar la calidad de la imagen sin intervención humana. Estos enfoques se basan en características y propiedades visuales extraídas de la imagen, que se utilizan para calcular una puntuación de calidad. Estas características pueden incluir medidas de nitidez, contraste, estructura, color, distribución de texturas y otros aspectos relevantes para la percepción visual.

La elección entre enfoques subjetivos u objetivos depende del contexto y los recursos disponibles. Los enfoques subjetivos son considerados como la referencia estándar para la evaluación de la calidad de la imagen, ya que capturan la apreciación humana. Sin embargo, estos enfoques pueden ser costosos y requieren de un número significativo de participantes. Mientras que los enfoques objetivos se pueden llegar a automatizar, haciendo que sea muy prácticos para grandes cantidades de datos y diversas aplicaciones.

No obstante, el objetivo es desarrollar algoritmos y métricas que puedan proporcionar una estimación precisa y consistente de la calidad de la imagen, teniendo en cuenta tanto aspectos subjetivos como objetivos respecto a las distorsiones. Y, de esta forma, poder evaluar y comparar diferentes métodos de adquisición, compresión, restauración o manipulación de imágenes teniendo en cuenta que el receptor final es el humano.

Para abordar el problema de la IQA, se emplean diversas técnicas y enfoques. Entre ellos se incluyen métodos basados en características, modelos de percepción visual, aprendizaje automático y técnicas de procesamiento de señales. Uno de los enfoques comunes es utilizar características básicas de la imagen. Las características elementales de la imagen son por ejemplo el contraste, la nitidez, la exposición y la uniformidad del color [3]. Estas características pueden ser cuantificadas mediante algoritmos de procesamiento de imágenes y proporcionar una estimación inicial de la calidad.

Por otro lado, los modelos de percepción visual intentan simular cómo el sistema visual humano percibe y evalúa la calidad de la imagen. Estos modelos se basan en el entendimiento de los mecanismos y procesos perceptuales del cerebro humano, y utilizan características visuales y estadísticas para calcular la calidad percibida [2, 15]. Buscan emular la forma en que los humanos interpretan y responden a las imágenes en términos de su calidad visual [16, 17].

Habitualmente, se suelen emplear algoritmos de aprendizaje automático para tratar de resolver el problema. Se utilizan técnicas supervisadas o no supervisadas para intentar aproximar una función que a partir del conjunto de características extraídas pueda determinar la calidad de la imagen en una escala específica, generalmente en el rango de 0 a 10.

Las aplicaciones comunes de los algoritmos de estimación de calidad suele ser la comparativa entre algoritmos de compresión, ya que permite elegir aque-

llos con menor pérdida de información, la generación de mapas de calidad² permitiendo el estudio de métodos de reducción de ruido, la determinación de la calidad del servicio de transmisión o *quality-of-service* (*QoS*), ya que permiten evaluar los errores de transmisión [18,19]. Se podría incluso extender al pre-procesamiento de datos de entrenamiento o estimar la precisión de un modelo de predicción basado en la calidad de los datos [20].

El uso de algoritmos IQA se encuentra ampliamente difundido en el ámbito general de las imágenes 2D. Sin embargo, en este trabajo nos enfocaremos en el ámbito de las aplicaciones médicas y analizaremos los mecanismos existentes a la hora de evaluar la calidad de las imágenes. Lamentablemente, la naturaleza de estas imágenes y las distorsiones que pueden presentar, implican una disminución en la precisión de los modelos cuando se aplican directamente sobre ellas [14]. A parte, el número de métodos propuestos decrece al desplazarnos a tres dimensiones debido al aumento de complejidad, con ninguno, al momento de escritura, aplicado a imágenes médicas 3D. Es por ello que se propone investigar específicamente el uso de métodos tridimensionales para el ámbito biomédico, aplicado a las reconstrucciones y visualizaciones volumétricas que se suelen emplear en medicina.

1.2. Motivación

En el caso del ámbito biomédico, dado los rápidos avances en las últimas década de las técnicas no invasivas de imágenes y la gran cantidad de fabricantes de equipamientos, nació el estándar *DICOM* [21] en 1995 con objetivo de hacer que el intercambio de imágenes médicas se realice de forma fácil, segura y con alta calidad. Permitiendo integración con diversos sistemas e incluso almacenar información extra en forma de metadatos y anotaciones, así como segmentaciones que permiten la reconstrucción 3D de diferentes regiones anatómicas.

Cada vez más frecuentemente se emplean volúmenes tridimensionales, como tomografías computarizadas (TC) o resonancias magnéticas (RM) en lugar de radiografías convencionales, porque proporcionan una visión más completa y detallada de la anatomía y las estructuras internas del cuerpo, ver (1.1). Esto brinda a los médicos una comprensión más completa de la anatomía, les ayuda a identificar con mayor precisión lesiones, enfermedades o anomalías y a la planificación quirúrgica y otros [22–24].

A pesar de ello, las distorsiones, que son una ocurrencia común en las imágenes cotidianas, están muy presentes en las imágenes médicas [25]. Prevalecen

²Calidad perceptual en diferentes regiones de la imagen. Resaltan áreas que pueden tener degradación visual o distorsiones.

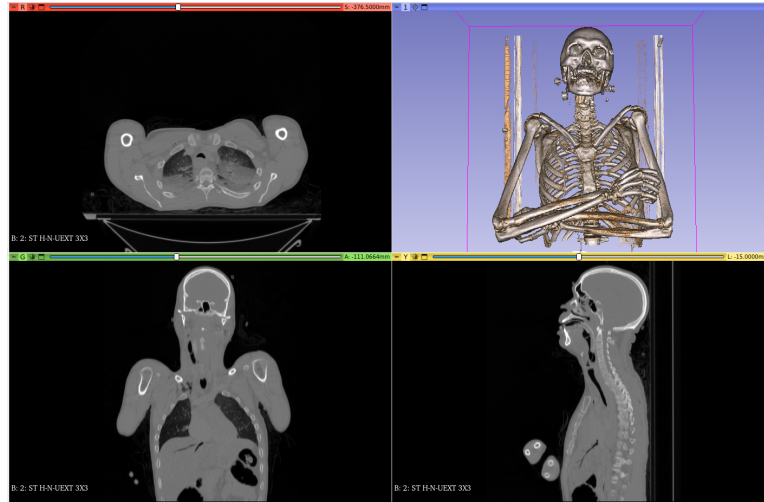


Figura 1.1: Ejemplo de visualización de un directorio *DICOM*. Se pueden observar las proyecciones axial, coronal y sagital. Además de una renderización volumétrica de los huesos. Para ello se ha utilizado *Slicer3D* [1]

las distorsiones de contraste, ruido y difuminado³, que se detallarán en la sección 2.4. Estas a su vez, podrían afectar al volumen 3D que se puede generar a partir las imágenes médicas.

En [26] se estudió las razones por las que se suelen rechazar las radiografías, la relación con la calidad de la imagen y el valor del diagnóstico final. Reveló que la mayoría de los rechazos se producen por errores de posicionamiento, valores inadecuados de exposición, artefactos y los problemas de cooperación del paciente. Además, no es difícil imaginar que una alta calidad de imagen médica tiene implicaciones significativas sobre el cuidado del paciente. Ya que la mala calidad de imagen puede provocar diagnósticos erróneos o falsos negativos. Sin mencionar los elevados costes que supone realizar nuevas pruebas para conciliar las anteriores.

Resolver este problema, o dar pasos hacia adelante, formulando una medida de calidad puede conllevar a la mejora de los dispositivos médicos, de los algoritmos de comprensión, almacenado y transmisión de información (tanto a nivel médico como a cualquier aplicación con datos tridimensionales). Resultando en reducción de costes (menos pruebas), de tiempo de consultas y mejorando la calidad del diagnóstico médico.

³Con ruido nos referimos a pequeñas fluctuaciones no deseadas en los colores de los píxeles debido a interferencias de todo tipo. Difuminado se refiere a la pérdida de detalles en los bordes, como una pérdida de enfoque.

1.3. Objetivos

Para el desarrollo de este documento se plantean una serie de objetivos. El objetivo principal consiste en encontrar un método adecuado para abordar al problema de la estimación de la calidad de imágenes médicas tridimensionales. Este objetivo se puede descomponer en una serie de metas parciales:

1. Realizar una revisión del estado del arte para la estimación de calidad de imágenes $3D$, así como de la calidad de imágenes médicas en particular.
2. Estudiar las distorsiones comunes en el ámbito de la imagen. Analizar los patrones de distorsión que afectan la calidad de las imágenes biomédicas.
3. Analizar los enfoques más prometedores que permitan abordar el problema planteado.
4. Generar un conjunto de datos sintético que permita validar los métodos analizados. Para ello, será necesario estudiar diferentes estrategias y métricas de evaluación objetivas [27–29].
5. Realizar un estudio experimental que permita validar los enfoques propuestos y extraer conclusiones sobre su aplicabilidad al problema.

1.4. Planificación del proyecto

Al planificar el proyecto, es fundamental tener en cuenta que la asignatura del Trabajo Fin de Grado (TFG) tiene una carga de 12 créditos ECTS, donde cada crédito representa aproximadamente 25 horas de trabajo. En total, se estima que se necesitarán alrededor de 300 horas para llevar a cabo el proyecto. Considerando que el segundo cuatrimestre tiene aproximadamente 20 semanas, se requerirá dedicar al TFG unas 15 horas por semana, lo cual equivaldría a unas 3 horas diarias durante 5 días a la semana.

La naturaleza del proyecto no presenta una complejidad significativa en términos de su alcance y requisitos, lo cual permite abordar su desarrollo a través de un enfoque de ciclo de vida en cascada [30]. No obstante, bajo este enfoque se evita retroceder en cualquiera de las fases del ciclo, y aunque se espera que el diseño y los requisitos del sistema sean estables, existe la posibilidad de realizar ajustes menores conforme se obtenga más información sobre el problema y los métodos. Es por ello que utilizamos una pequeña variante, la versión con retroalimentación.

Las fases del ciclo de vida son:

- **Análisis de requisitos:** Consiste en reuniones iniciales con los clientes, en este caso sería los directores del TFG. Se organiza el análisis bibliográfico del problema *IQA* y *PCQA*⁴, teniendo en cuenta un estudio previo de las distorsiones médicas.
- **Diseño:** Consistió en la investigación y selección de métodos conforme al análisis anterior, tanto para la resolución como la validación de la solución. Así como pruebas preliminares y diseño del software de experimentación.
- **Implementación:** Consiste en la adaptación de las técnicas encontradas, implementación de nuevas funcionalidades y generación de un conjunto de datos médicos nuevos.
- **Pruebas:** Realización de diversos experimentos de validación, tanto al la generación de las distorsiones como a los modelos y resultados.

Tarea	Semanas - Horas	Febrero			Marzo			Abril				Mayo					Junio				Julio			
Análisis de Requisitos	4 - 60	21	28	07	14	21	28	04	11	18	25	02	09	16	23	30	06	13	20	27	04	11	18	25
Diseño	4 - 60																							
Implementación	6 - 90																							
Pruebas	6 - 90																							

Tabla 1.1: Planificación temporal inicial del proyecto

La planificación (1.1) se tomó como referencia, pero no de forma estricta. Ya que se tuvo en cuenta que el autor estaba realizando prácticas de empresa, tenía una asignatura y participaba de un curso de *Google* ofrecido por la universidad. Además, se esperaba que ocurriera retrasos sobre todo en la implementación, como se puede ver en (1.2), dado la novedad de la propuesta y dificultad del problema. Un ejemplo fue a la hora de simular las distorsiones médicas, caso que fue algo iterativo y manual.

Tarea	Semanas - Horas	Febrero			Marzo			Abril				Mayo					Junio				Julio			
Análisis de Requisitos	5 - 75	21	28	07	14	21	28	04	11	18	25	02	09	16	23	30	06	13	20	27	04	11	18	25
Diseño	4 - 60																							
Implementación	8 - 120																							
Pruebas	6 - 90																							

Tabla 1.2: Planificación resultante del proyecto

Para realizar este proyecto se tuvo en cuenta los siguientes materiales: Suscripción a *Google Colab Pro*, un portátil personal de gama media, *Google Drive 100GB* y otros gastos. Además, para el coste estimado, se asume un salario de 25€/hora, como para un investigador *senior* o responsable I+D de una empresa tecnológica en España.

⁴ *Point cloud quality assessment* o estimación de calidad de nubes de puntos

Respecto al servidor GPU, con las especificaciones actuales de *Google*, se estima un coste aproximado de 10.000€. Se asume una amortización de 2 años, lo que implica un pago diario de 13.70€. El desglose total de los costes se puede ver en la siguiente tabla (1.4)

Fecha inicio	21/02/2022
Fecha fin	25/07/2022
Duración	154 días, 110 laborables

Tabla 1.3: Total de horas y días trabajados

Item	Costo
Salario	8 250.00€
Portátil de Gama Media	700.00€
Google Colab Pro	55.50€
Servidor GPU	2 109.8€
Google Drive 100GB	10.00€
Otros	300.00€
Total	11 425.3 €

Tabla 1.4: Estimación final de coste del proyecto

Capítulo 2

Fundamentos Teóricos

2.0.1. Subproblemas de la IQA

Existen tres subproblemas presentes en el ámbito de *IQA*. Los primeros, son problemas donde tenemos acceso a la imagen original, que suponemos exenta de desperfectos, en la cual se pueden aplicar métodos basados en diferencia de características entre ambas, como puede ser al nivel del color de píxel posición a posición, y se denomina “*Full Reference*” (*FR*). La tarea, aparentemente sencilla, en realidad presenta una complejidad alta dada por la necesidad de codificar la percepción humana a la hora de calificar la calidad de una imagen [31]. Ya que métricas que miden distancias no suelen ser suficientes al no haber buena correlación entre la calidad percibida y el resultado de la métrica.

La mayoría de las veces no se menciona, pero al optar métodos de sensibilidad al error (distancias) se imponen un conjunto de suposiciones cuestionables. Primeramente se asume la misma importancia para todas las señales de la imagen, que la magnitud del error es lo único que determina la calidad, que el contenido de la imagen no afecta al resultado final tras aplicar una distorsión y que si cambiamos el orden de las señales la medida de distorsión no es afectada. Lamentablemente, ninguna de estas suposiciones se cumplen [3], ver imágenes (2.1,2.2).

El siguiente subproblema es aquel donde tenemos algún tipo de información adicional respecto a la imagen original en el momento de análisis de la calidad de la imagen final, denominados “*Reduced Reference*” (*RR*). La información extra puede incluir características estadísticas, metadatos, parámetros de compresión o características extraídas de una región de interés específica.

Y por último, tenemos aquellos problemas donde desconocemos el origen y cualquier información respecto a la imagen inicial, denominados problemas “*No reference*” (*NR*). Estas métricas están exentas de cualquier información

de referencia y se centran en capturar características generales de calidad.

La evaluación de calidad de imagen sin referencia es, quizás, el problema más difícil en el análisis de imágenes. De cierto modo, el modelo debe ser capaz de evaluar la calidad de cualquier imagen sin saber nada de la imagen “real”, original. Superficialmente parece “misión imposible”. No obstante, esa es una tarea sorprendentemente sencilla para el ser humano [3].

Para resolver problemas NR, debemos disponer de conocimientos de la naturaleza de las imágenes de las que tratamos y los efectos de las distorsiones. Lo que se denomina estadísticas naturales de escena (NSS, por sus siglas en inglés). Un ejemplo sería JPEG, un algoritmo de compresión que se codifica por bloques 8x8. Los efectos negativos de la compresión se representan por el difuminado entre bloques y los artefactos que generan. Entender estos efectos permite diseñar métricas específicas [32, 33]

As veces resulta difícil describir las características de la imagen y los efectos de la distorsión. Es por ello que los métodos de aprendizaje profundo son cada vez más frecuente y dan mejores resultados. Permitimos que sea la máquina la que aprenda las propiedades de la distorsión, su relación con el contenido y efecto sobre la percepción visual [34–36].

La complejidad del problema crece conforme nos desplazamos a las tres dimensiones. El analizar la calidad de los modelos 3D implica mayor nivel de dificultad dado que nos enfrentamos a dos grandes retos: La complejidad computacional de las operaciones y la escasez de bases de datos etiquetadas sobre objetos tridimensionales para entrenar y evaluar modelos.

Para las nubes de puntos, que representan una colección de puntos en un espacio tridimensional (x, y, z) cada uno con un color asociado RGB^1 , se pueden emplear métricas y algoritmos basándose en criterios como la densidad de puntos, la uniformidad, la precisión geométrica y la detección de artefactos. También se pueden considerar aspectos relacionados con la coherencia de los colores o texturas asociadas a los puntos [37–39]. Un enfoque común es la evaluación de calidad de una nube de puntos tridimensional mediante proyecciones 2D desde diferentes perspectivas [40–42]. De esta forma podemos tratar el problema como uno de IQA 2D reduciendo la complejidad computacional, pudiendo implementar métodos y soluciones ya existentes.

Teniendo en cuenta todas estas consideraciones, el presente TFG aborda la estimación, sin referencia, de calidad de imágenes médicas en espacio tridimensional.

²La distancia de *Minkowski* es una métrica vectorial que puede considerarse como una generalización tanto de la distancia euclídea como de la distancia de Manhattan .

³*Mean squared error* o error cuadrático medio es una métrica de distancia que se calcula como la media de la suma de las diferencias al cuadrado.

¹RGB son las siglas en inglés para rojo, verde y azul. Los colores se representan por tripletas de valores en escala 0-255 ó 0-1 que significan la cantidad que aporta cada color.

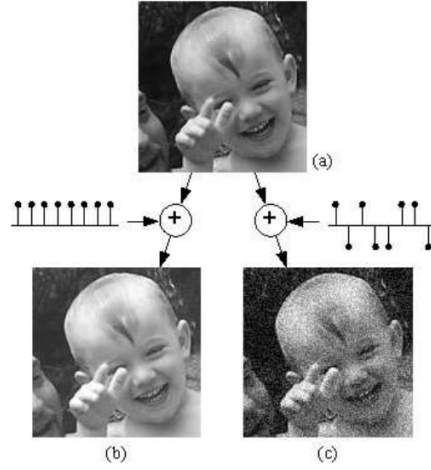


Figura 2.1: En este ejemplo, extraído de [2], vemos que sumar una constante positiva a una imagen de referencia (a) produce la imagen (b) que contiene la misma distancia $Minkowski^2$ que (c), imagen fabricada por la misma constante pero permutando signo de forma aleatoria. Siendo que la percepción final es que la imagen (c) es peor que la imagen (b).

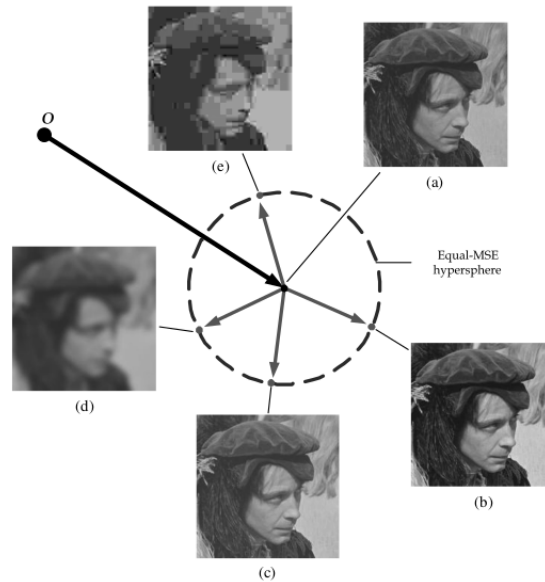


Figura 2.2: En este ejemplo, extraído de [3], la misma imagen distorsionada de distintas maneras resulta en el mismo valor $MSE^3=181$. Siendo evidente que algunas distorsiones producen efectos visuales más marcados que otras.

2.1. Aprendizaje Automático

El aprendizaje automático [43] o *MachineLearning* (*ML*) es una de las ramas que compone lo que definimos como la inteligencia artificial (*IA*). Esta última se puede definir, según [44], como “La ciencia e ingeniería de crear máquinas inteligentes, especialmente programas de computadora inteligentes. Está relacionada con la tarea similar de utilizar computadoras para comprender la inteligencia humana...”

En este caso hablamos de dar soluciones a problemas complejos sin solución analítica (o que resulta muy costoso hallarla), es decir, necesitamos que la computadora sea la que identifique los patrones en los datos y realice predicciones sobre ellos [45]. Se puede definir más formalmente que un programa aprende de la experiencia E con respecto a alguna clase de tareas T y una métrica de rendimiento P si su rendimiento en las tareas T , medido con P , mejora con la experiencia E [46].

Dependiendo de factores como las necesidades del problema, la naturaleza de los datos a utilizar o el objetivo a alcanzar, podemos encontrar distintos tipos de algoritmos de aprendizaje. En este documento se recogerán dos grandes grupos: aprendizaje supervisado y aprendizaje no supervisado. En el primero disponemos de un conjunto de datos anotados, es decir, con las salidas deseadas para cada ejemplo y en el segundo se espera que sea la máquina la que determine los patrones (2.3, 2.4).

En general se suelen aplicar las técnicas de *ML* sobre grandes conjuntos de datos sobre los cuales deseamos detectar los patrones subyacentes [47].

Puede observarse que dada estas descripciones, al problema presente podemos aplicarle alguna técnica de *ML*: Tenemos datos de entrada (características extraídas de nubes de puntos distorsionadas) y una salida (valor de calidad). Además, existen conjuntos de datos públicos etiquetadas para distintos tipos de distorsiones. Esto significa que estamos entre un problema de aprendizaje supervisado.

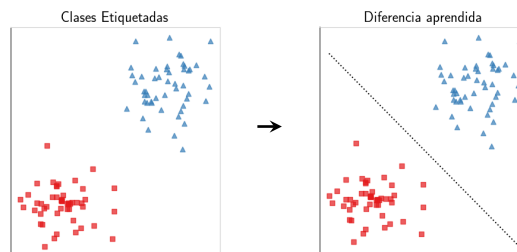


Figura 2.3: Ejemplo de aprendizaje supervisado, vemos como a partir de un conjunto de clases etiquetadas aprendemos un hiper plano que las separa.

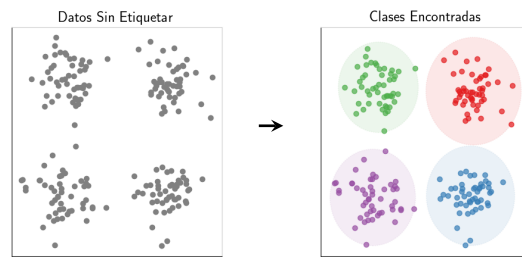


Figura 2.4: Ejemplo de aprendizaje no supervisado, dado un conjunto de puntos aprendemos un conjunto de clases a partir de los patrones.

2.2. Aprendizaje Profundo

En el aprendizaje profundo o *Deep Learning* (*DL*), a diferencia de los modelos anteriores donde tenemos un conjunto de variables extraídas por un humano experto, las características sobre la cual inferimos son obtenidas por el propio modelo automáticamente [48–50]. En términos generales, la extracción automática de características suele desempeñar mejores resultados en contra de las características manuales.

La mayoría de los modelos de DL son basados en múltiples capas jerárquicas de procesamiento de datos. Las más conocidas son las redes neuronales (ANN, por sus siglas en inglés), modelo bioinspirado que simula el funcionamiento de las neuronas del cerebro humano (abstracción simplificada) [51, 52].

A alto nivel, el funcionamiento de una red neuronal implica tres etapas principales: entrada, procesamiento y salida. En la etapa de entrada, se proporciona a la red neuronal un conjunto de datos o características que representan la información que se desea analizar o procesar. Estos datos de entrada se propagan a través de la red neuronal (*feedforward*). En la etapa de procesamiento, las neuronas reciben las entradas y realizan cálculos utilizando pesos y funciones de activación. Los pesos representan la importancia relativa de las diferentes entradas en el cálculo, y las funciones de activación determinan la salida de una neurona en función de su entrada. A medida que los datos se propagan a través de la red neuronal, las capas intermedias procesan y combinan las entradas, extrayendo características relevantes y creando representaciones internas cada vez más abstractas. Esto permite que la red neuronal aprenda y descubra patrones en los datos. Finalmente, en la etapa de salida, la red neuronal produce una respuesta o predicción basada en las características extraídas. Esto puede ser la clasificación de una imagen, la predicción de un valor numérico o cualquier otro resultado deseado. En esta última etapa se calcula el error de predicción respecto a la salida deseada con la función de pérdida y se ajusta los pesos respectivamente.

El aprendizaje de una red neuronal se logra mediante un proceso llamado entrenamiento. Donde de forma iterativa repetimos el proceso descrito anteriormente varias veces con distintos ejemplos. El conjunto de datos es muy relevante para el correcto aprendizaje. Debe de ser representativo, extenso y limpio de anomalías ya que estaremos extrayendo características y relevancias a partir de ellos.

En definitiva, una red neuronal es en esencia una serie de ajustes de parámetros para lograr el resultado deseado. Estos incluyen ajuste de los pesos y sesgos iniciales, selección de las funciones de activación, como las más utilizadas sigmoide o ReLU, de una función de pérdida y un optimizador, encargado de determinar como ajustar los pesos según el error obtenido en cada fase del entrenamiento. No obstante, existe un fenómeno denominado sobreentrenamiento o *overfitting*. Ocurre cuando hay un sobreajuste de los parámetros hacia los datos de entrenamiento, disminuyendo la capacidad de generalización del modelo. Informalmente, es como decir que el modelo ha memorizado los resultados y, por ello, con datos nunca vistos posee errores substancialmente altos. Para lidiar con estos problemas se deben elegir también formas de regularización del modelo, es decir, restricciones sobre el entrenamiento para evitar el sobre ajuste.

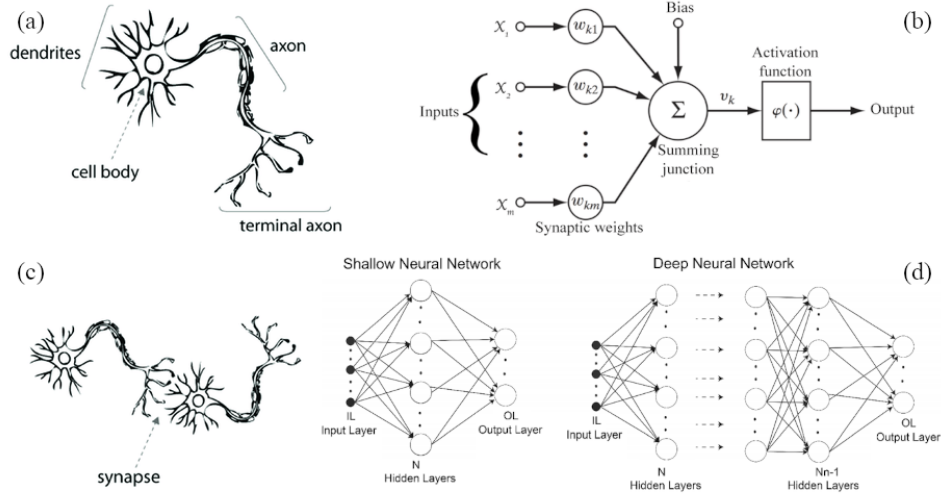


Figura 2.5: Ejemplo gráfico de una red neuronal [4–6]. En (a) vemos una neurona con su representación artificial simplificada (b). En (c) vemos una conexión entre neuronas y en (d) la representación de diferentes profundidades de conexiones artificiales.

2.2.1. Redes Convolucionales

Las redes convolucionales o *convolutional neural network* (CNN) [53,54] son un tipo de arquitectura de redes neuronales diseñadas específicamente para el procesamiento de datos estructurados en forma de matrices, como imágenes. Se ha descubierto que son aplicables para el procesamiento de texto, sonidos y, recientemente, a superficies tridimensionales. Utilizan capas convolucionales que aplican filtros a regiones locales de la entrada para extraer características relevantes. En la figura (2.6) podemos ver un ejemplo de esquema jerárquico de extracción de características para el diagnóstico médico a partir de una radiografía. Con ese ejemplo se puede observar las diferencias con respecto a las ANN, posee dos capas adicionales: capas convolucionales y capas de *pooling*.

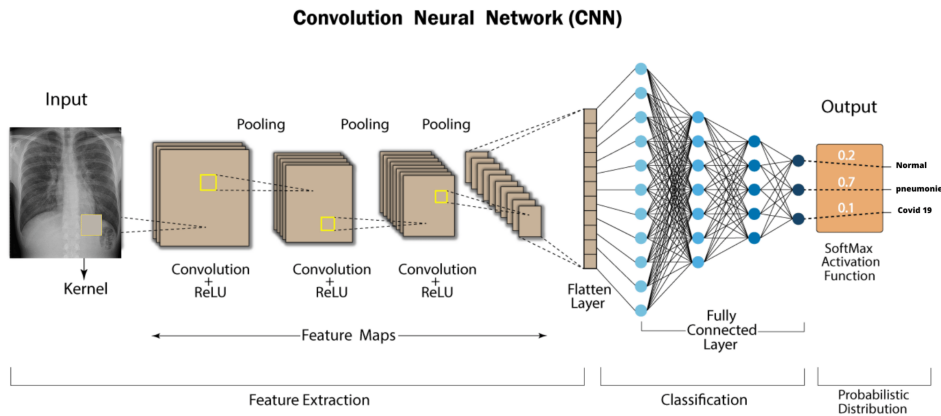


Figura 2.6: Ejemplo extraída de [7] del proceso de convolución sobre una radiografía pulmonar para la detección de enfermedades.

2.2.2. Capas convolucionales

Para simplificar la explicación, la realizaremos sobre imágenes 2D. Una capa convolucional es encargada de realizar la operación de convolución sobre los datos de entrada. La convolución se refiere a una operación matemática que combina dos funciones para crear una tercera función. En este caso, se aplica una operación de convolución entre una matriz de entrada (como una imagen) y un filtro (*kernel*). La operación de convolución implica deslizar el filtro sobre la matriz de entrada, multiplicando los elementos coincidentes y sumándolos para obtener un único valor en la matriz de salida, conocida como mapa de características. Este proceso se repite en diferentes ubicaciones de la matriz de entrada para generar el mapa de características completo. En la figura (2.7) vemos una operación sobre la ubicación inicial de la imagen, esquina superior izquierda. La elección del siguiente trozo o *patch* de

la imagen suele venir determinado por el paso o *stride*. Habitualmente se utiliza un *stride* de 1. Es decir, elegimos la matriz adyacente con distancia horizontal igual a 1 hasta llegar al final de esa fila y luego nos desplazamos 1 hacia abajo. Por medio de este proceso, la red es capaz de capturar dependencias temporales y espaciales en los datos con la aplicación de los filtros correspondientes.

Podemos observar en (2.7) que aplicar directamente el operador de convolución a una imagen resulta en una reducción del tamaño del mapa de activación debido a la naturaleza del operador. Sin embargo, esto no siempre es deseable. Para abordar este problema, se puede agregar relleno o *padding* a la imagen de entrada utilizando información existente en la misma. Esto garantiza que el mapa de activación tenga la misma dimensionalidad que la imagen original. Además, es posible reducir aún más la salida ajustando los saltos o *strides* del filtro de convolución mientras se recorre la imagen.

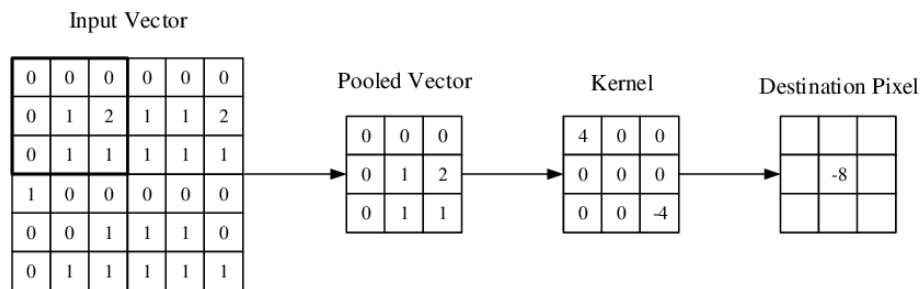


Figura 2.7: Representación visual de la operación de convolución sobre una imagen, extraída de [8].

2.2.3. Capa de pooling

El propósito principal de las capas de *pooling* es reducir la cantidad de parámetros y la complejidad computacional de la red, al tiempo que conservan las características más relevantes. Además, el *pooling* puede ayudar a hacer que la representación sea invariante a pequeñas variaciones en la posición o el tamaño de los objetos en la imagen, lo que mejora la capacidad de generalización del modelo.

En la figura (2.8) vemos un operador de *pooling* común, el operador de valor máximo. También es habitual el uso del operador de valor medio y valor mínimo. El pooling, al igual que la convolución, posee un filtro o ventana que recorre los datos dado un salto o *stride* al moverse por los mismos,

Las capas convolucionales y de pooling trabajan en conjunto para procesar y extraer características. Dependiendo de la complejidad del problema, se puede ajustar el número de estas.

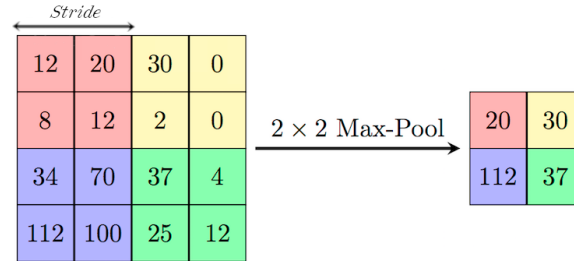


Figura 2.8: Ejemplo de operación de *max-pooling* con *stride* a 2.

2.2.4. Capas totalmente conectadas

Las capas totalmente conectadas o *fully connected*, también llamadas capas densas o *dense*, son aquellas en las que todas sus neuronas están conectadas con todas las neuronas de la capa anterior y de la siguiente. Si bien existen modelos totalmente convolucionales, resulta común que las CNN's incluyan capas totalmente conectadas al final de la arquitectura. Estas capas forman una ANN clásica. La salida de la última capa densa, siendo la salida de la red entera, es donde se evaluará la función de pérdida elegida, y al igual que en una red neuronal clásica se utilizará este valor para ajustar los pesos.

2.2.5. Aplicadas a Videos

Las redes convolucionales se pueden llegar a aplicar incluso a videos. Para ello, se puede utilizar una variante de las redes convolucionales llamada redes convolucionales 3D(3D CNNs) o redes convolucionales espaciotemporales. Estas redes están diseñadas específicamente para capturar tanto las características espaciales como las temporales presentes en los videos.

La principal diferencia entre una red convolucional tradicional y una 3D CNN es la adición de una dimensión temporal en las operaciones de convolución. En lugar de considerar solo imágenes individuales, se toman secuencias de imágenes (*frames*) para capturar la información temporal.

En este TFG se explora el uso de una 3D CNN capaz de analizar videos que pertenece a la familia que se conoce como *SlowFast networks* [9]. Están basadas en dos caminos de entrada de datos. Un conjunto de *frames* espaciados en el tiempo, *slow path*, para obtener información espacial y otro con todos ellos, *fast path*, para obtener información de movimiento, ver (2.9).

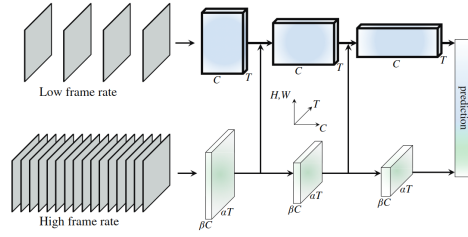


Figura 2.9: Ejemplo extraído de [9] para ilustrar la distinción entre la extracción espacial del camino “*slow path*” y de movimiento con el “*fast path*”.

2.2.6. Aplicadas a nubes de puntos

De forma similar al caso de los videos, actualmente existen modelos de 3D CNN’s capaces de procesar nubes de puntos al añadir una dimensión más que representa la profundidad de los puntos. Sin embargo, la complejidad de diseño y tiempo de cómputo para estos modelos 3D crece enormemente. Ya que habitualmente las nubes de puntos están formadas por puntos dispersos en el espacio, en lo que denominamos datos sin estructura ni orden propio, y debemos mapear una operación de convolución que está basada en operaciones sobre datos ordenados y estructurados.

Aunque cada vez hay más métodos que se aplican directamente sobre la nube de puntos desde la publicación de *PointNet* [55], habitualmente se intenta estructurar la información de las nubes de puntos mediante lo que denominamos vóxeles [10]. La voxelización es el proceso de transformar una nube de puntos u otra representación tridimensional en una estructura discreta conocida como volumen voxelizado. Esto implica dividir el espacio tridimensional en una cuadrícula de voxels y asignar valores a cada voxel según la información contenida en los datos originales. La voxelización proporciona una representación estructurada y discreta que permite el uso de técnicas específicas para volúmenes y facilita el procesamiento y análisis de datos 3D.

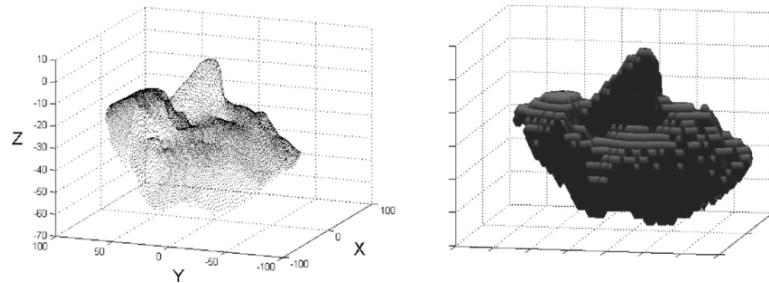


Figura 2.10: Ejemplo extraído de [10] que demuestra el resultado de transformar una nube de puntos sin estructura en una cuadrícula voxelizada.

2.3. Ensemble o Conjunto *Deep Learning*

Un *ensemble*, en el contexto del aprendizaje automático, es una técnica que combina múltiples modelos de aprendizaje para mejorar la precisión y el rendimiento general de las predicciones. En lugar de depender de un único modelo, se crean múltiples modelos y se combinan sus predicciones para obtener un resultado final más robusto y preciso [11, 43, 47].

La idea fundamental detrás de los *ensembles* es que los diferentes modelos pueden tener fortalezas y debilidades diferentes, y al combinar sus predicciones, se puede obtener una mejor generalización y un mayor rendimiento en una variedad de situaciones. Para construir un *ensemble* se suele utilizar un conjunto de técnicas que se describirán a continuación.

El *bagging* consiste en generar múltiples conjuntos de datos de entrenamiento mediante muestreo con reemplazo, entrenando un modelo en cada conjunto y promediando o ponderando sus predicciones. En el *boosting*, los modelos se construyen secuencialmente, corrigiendo los errores del modelo anterior, y se combinan para formar un modelo más fuerte. La aumentación de datos consiste en ampliar el conjunto de entrenamiento para mejorar la capacidad de generalización del modelo y reducir el sobreajuste por medio de transformaciones sobre los datos como la rotación y ampliación, se podría usar como paso en el proceso de *bagging*. Los *random forests* combinan *bagging* y árboles de decisión, generando múltiples árboles utilizando diferentes subconjuntos de datos y características. Las predicciones de los árboles individuales se combinan para obtener la predicción final. Por último, el *stacking* entrena múltiples modelos base y utiliza un meta-modelo para combinar sus predicciones. Cada estrategia tiene sus beneficios y consecuencias.

El presente TFG evalúa el uso de un meta-modelo para la estimación de calidad de las imágenes médicas 3D sin referencia.

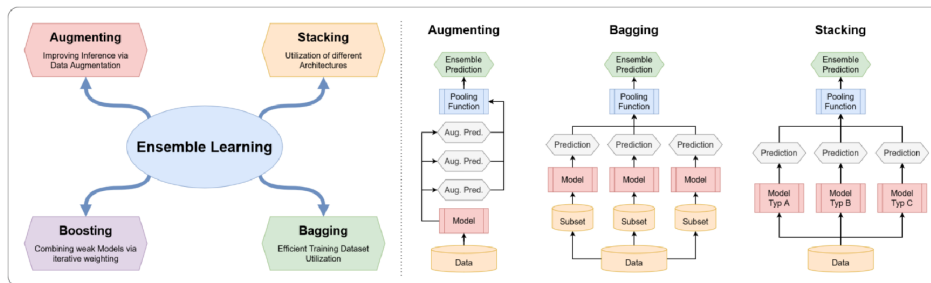


Figura 2.11: Representación de métodos de *ensemble* extraído de [11]

2.4. Imágenes médicas y distorsiones

Las tomografías computarizadas (CT) son un tipo de técnica de imagen médica que utiliza rayos X para obtener imágenes detalladas del interior del cuerpo. Durante una tomografía computarizada, el paciente se coloca en una mesa que se mueve a través de un anillo en forma de donut llamado *gantry*. Dentro del *gantry*, se encuentra un tubo de rayos X que gira alrededor del paciente, emitiendo haces de rayos X en forma de abanico. Los detectores ubicados en el lado opuesto del *gantry* registran la cantidad de rayos X que atraviesan el cuerpo del paciente. Estos datos se recopilan en múltiples ángulos y se utilizan para reconstruir imágenes transversales del cuerpo.

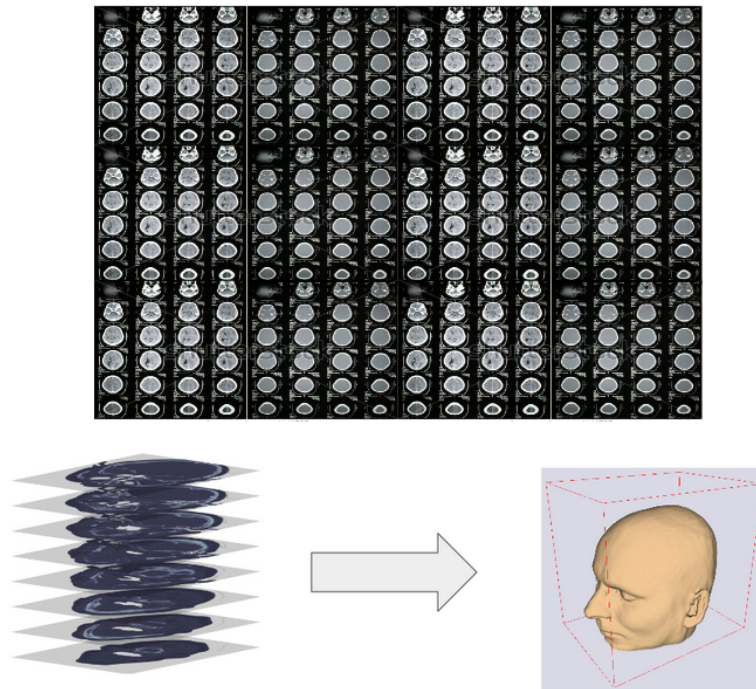


Figura 2.12: Ejemplo de tomografía computarizada y modelo 3D [12]

El número de imágenes en una tomografía computarizada se selecciona en función de varios factores, como el área del cuerpo que se está examinando, el propósito clínico de la exploración y las preferencias del radiólogo o médico que interpreta las imágenes. Ajustar el número de imágenes puede influir en el tiempo de adquisición, la cantidad de radiación utilizada y la cantidad de información detallada que se obtiene de la exploración. Afecta directamente a la calidad del modelo 3D generado al final ya que el número de cortes es la tercera dimensión que relaciona las imágenes (profundidad).

Sobre todo nos centraremos en las distorsiones geométricas que ocurren en

la generación volumétrica de la imagen. La generación del volumen consiste en disponer de un conjunto segmentado² en todas las capas de las imágenes. A continuación usando el conjunto segmentado unificamos las coordenadas de los puntos por medio de intersecciones entre rayos proyectados sobre las imágenes (*ray casting*), ver (2.12).

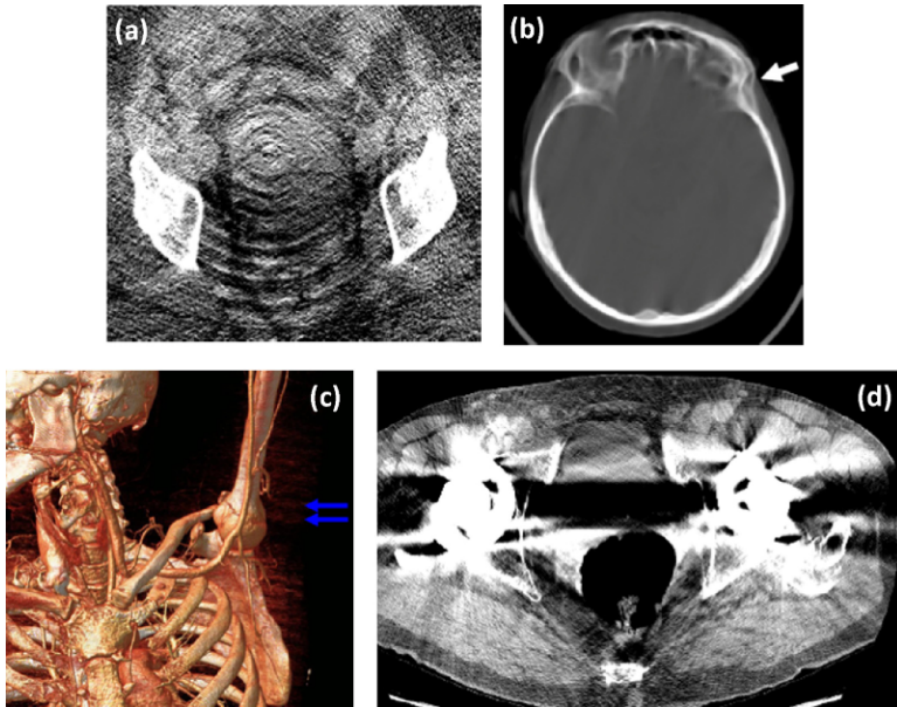


Figura 2.13: Ejemplo de artefactos sobre imágenes Dicom [13]: (a) un artefacto de anillo, (b) un difuminado provocado por movimiento, (c) artefactos producidos por interpolación helicoidal, (d) artefactos de iluminación y dispersión del haz.

Dentro de las causas de las distorsiones geométricas sobre los volúmenes 3D están el difuminado por movimiento, errores de contraste (dificultad al segmentar), artefactos luminosos y problemas de interpolación o ruido al generar la proyección (2.13).

Para simularlo, partiremos de ejemplos de imágenes médicas consideradas exentas de distorsiones, con su segmentación correspondiente realizada por profesionales. A continuación, generaremos la representación volumétrica como una nube de puntos y sobre ella simularemos las distorsiones.

²La segmentación se refiere al proceso de dividir una imagen o conjunto de datos en regiones o componentes más pequeños. Ejemplo, en una foto del bosque, segmentamos los árboles para distinguirlos del suelo.

Dos de ellas serán para simular los resultados de varios algoritmos de compresión, como puede ser *octree compression* [56] y la reducción de número de puntos por medio de mostreo uniforme (*uniform downsampling*), que suele ser el paso previo a cualquier transmisión de información. Un *octree* es una representación más eficiente que los vóxeles, se trata de la descomposición de forma recursiva de la escena en 8 partes hasta la profundidad máxima, donde cada nodo representa un cubo tridimensional llamado octante. En la compresión se analiza los octantes y se elimina los que aportan menos información. En el segundo, realizamos la reducción de la cantidad de puntos manteniendo una distribución uniforme en el espacio tridimensional.

Las demás representarán efectos que podrían ocurrir por desplazamiento de los puntos, como el movimiento del paciente y aquellos provocados por ruido en la transmisión de datos.

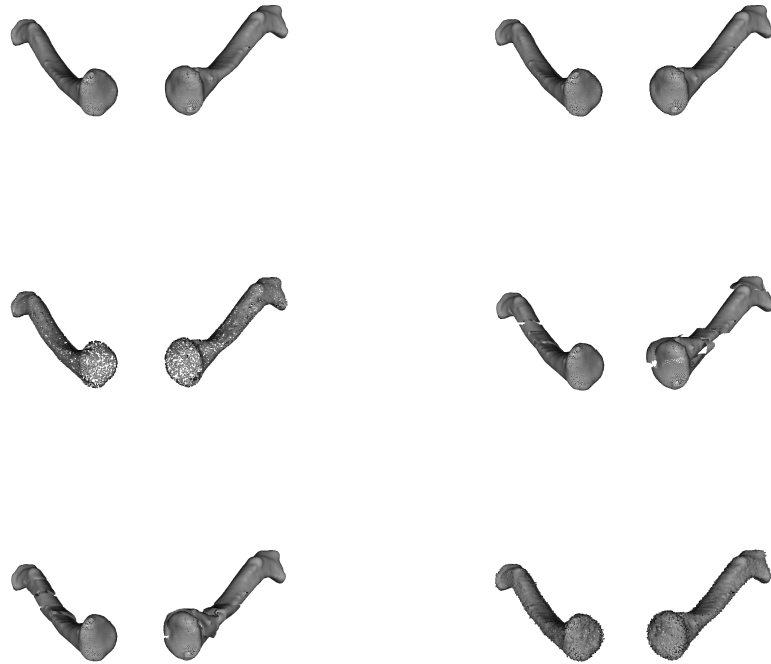


Figura 2.14: Ejemplo de distorsiones generadas. Leyendo de izquierda a derecha tenemos: la imagen original, compresión *octree*, reducción de puntos por *uniform downsampling*, movimiento local, rotación local y ruido gaussiano.

Capítulo 3

Estado del Arte

Capítulo 4

Materiales, Métodos y Distorsiones

4.1. Materiales

4.2. Métodos

4.3. Distorsiones

Capítulo 5

Implementación y Experimentos

5.1. Detalles Técnicos de Implementación

5.2. Experimentos

Capítulo 6

Conclusiones y Trabajos Futuros

Bibliografía

- [1] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, John Buatti, Stephen Aylward, James V. Miller, Steve Pieper, and Ron Kikinis. 3d slicer as an image computing platform for the quantitative imaging network. *Magnetic Resonance Imaging*, 30(9):1323–1341, 2012. Quantitative Imaging in Cancer. URL: <https://www.sciencedirect.com/science/article/pii/S0730725X12001816>, doi:<https://doi.org/10.1016/j.mri.2012.05.001>.
- [2] Kalpana Seshadrinathan, Thrasyvoulos Pappas, Robert Safranek, Junqing Chen, Zhou Wang, Hamid Sheikh, and Alan Bovik. Image quality assessment. *The Essential Guide to Image Processing*, 12 2009. doi:10.1016/B978-0-12-374457-9.00021-4.
- [3] Zhou Wang and Alan Conrad Bovik. Modern image quality assessment. In *Modern Image Quality Assessment*, 2006.
- [4] Zhenzhu Meng, Yating Hu, and Christophe Ancey. Using a data driven approach to predict waves generated by gravity driven mass flows. *Water*, 12, 02 2020. doi:10.3390/w12020600.
- [5] Ergün Akgün and Metin Demir. Modeling course achievements of elementary education teacher candidates with artificial neural networks. *International Journal of Assessment Tools in Education*, 5, 01 2018. doi:10.21449/ijate.444073.
- [6] A. Bakiya, K. Kamalanand, V. Rajinikanth, Ramesh Sunder Nayak, and Seifedine Kadry. Deep neural network assisted diagnosis of time-frequency transformed electromyograms. *Multimedia Tools and Applications*, 79(15):11051–11068, Apr 2020. doi:10.1007/s11042-018-6561-9.
- [7] Zakaria Rguibi, Abdelmajid Hajami, Dya Zitouni, Amine Elqaraoui, and Anas Bedraoui. Cxai: Explaining convolutional neural networks for medical imaging diagnostic. *Electronics*, 11(11),

-
2022. URL: <https://www.mdpi.com/2079-9292/11/11/1775>, doi: 10.3390/electronics11111775.
- [8] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *ArXiv e-prints*, 11 2015.
 - [9] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition, 2019. [arXiv:1812.03982](https://arxiv.org/abs/1812.03982).
 - [10] Berk Gokberk, Helin Dutağacı, Aydın Ulaş, Lale Akarun, and Bulent Sankur. Representation plurality and fusion for 3-d face recognition. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 38:155 – 173, 03 2008. doi:10.1109/TSMCB.2007.908865.
 - [11] Dominik Müller, Iñaki Soto-Rey, and Frank Kramer. An analysis on ensemble learning optimized medical image classification with deep convolutional neural networks, 2022. [arXiv:2201.11440](https://arxiv.org/abs/2201.11440).
 - [12] Cairo University. Medical visualization and volume rendering. <https://sbme-tutorials.github.io/2019/CG/notes/7-week7.html>, 2023.
 - [13] Li Sze Chow and Raveendran Paramesran. Review of medical image quality assessment. *Biomedical Signal Processing and Control*, 27:145–154, 2016. URL: <https://www.sciencedirect.com/science/article/pii/S1746809416300180>, doi:<https://doi.org/10.1016/j.bspc.2016.02.006>.
 - [14] Yong Ding. *Visual Quality Assessment for Natural and Medical Image*. 03 2018. doi:10.1007/978-3-662-56497-4.
 - [15] Illya Bakurov, Marco Buzzelli, Raimondo Schettini, Mauro Castelli, and Leonardo Vanneschi. Structural similarity index (ssim) revisited: A data-driven approach. *Expert Systems with Applications*, 189:116087, 2022. URL: <https://www.sciencedirect.com/science/article/pii/S0957417421014238>, doi:<https://doi.org/10.1016/j.eswa.2021.116087>.
 - [16] Edoardo Ardizzone and Alessandro Bruno. Image quality assessment by saliency maps. volume 1, 02 2012.
 - [17] Jinjian Wu, Jupo Ma, Fuhu Liang, Weisheng Dong, Guangming Shi, and Weisi Lin. End-to-end blind image quality prediction with cascaded deep neural network. *IEEE Transactions on Image Processing*, 29:7414–7426, 2020. doi:10.1109/TIP.2020.3002478.
 - [18] Netflix. Video multi-method assessment fusion. <https://github.com/charlespwd/project-title>, 2016.

-
- [19] Reza Rassool. Vmaf reproducibility: Validating a perceptual practical video quality metric. In *2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, pages 1–2, 2017. doi:10.1109/BMSB.2017.7986143.
 - [20] Zhou Wang. Applications of objective image quality assessment methods [applications corner]. *IEEE Signal Processing Magazine*, 28(6):137–142, 2011. doi:10.1109/MSP.2011.942295.
 - [21] Charles Parisot. The dicom standard. *The International Journal of Cardiac Imaging*, 11(3):171–177, Sep 1995. doi:10.1007/BF01143137.
 - [22] Karl Heinz Höhne, Henry Fuchs, and Stephen M. Pizer. 3d imaging in medicine : algorithms, systems, applications. 1990.
 - [23] Orhan Hakki Karatas and Ebubekir Toy. Three-dimensional imaging techniques: A literature review. *European Journal of Dentistry*, 8:132 – 140, 2014.
 - [24] Luuk H G A Hopman, Pranav Bhagirath, Mark J Mulder, Iris N Eggink, Albert C van Rossum, Cornelis P Allaart, and Marco J W Götte. Quantification of left atrial fibrosis by 3D late gadolinium-enhanced cardiac magnetic resonance imaging in patients with atrial fibrillation: impact of different analysis methods . *European Heart Journal - Cardiovascular Imaging*, 23(9):1182–1190, 11 2021. arXiv:<https://academic.oup.com/ehjci/advance-article-abstract/doi/10.1093/ehjci/jeab245/6316536>, doi:10.1093/ehjci/jeab245.
 - [25] Yuhao Sun and Gabriela Mogos. Impact of visual distortion on medical images. *IAENG International Journal of Computer Science*, 49:36–45, 03 2022.
 - [26] Elin Kjelle and Catherine Chilanga. The assessment of image quality and diagnostic value in x-ray images: a survey on radiographers’ reasons for rejecting images. *Insights into Imaging*, 13, 03 2022. doi:10.1186/s13244-022-01169-9.
 - [27] ITU-R. Methodology for the subjective assessment of the quality of television pictures. ITU-R Recommendation BT.500-13, International Telecommunication Union - Radiocommunication Sector (ITU-R), 2012.
 - [28] Junming Zhou, Gangyi Jiang, Xiangying Mao, Mei Yu, Feng Shao, Zongju Peng, and Yun Zhang. Subjective quality analyses of stereoscopic images in 3dtv system. 11 2011. doi:10.1109/VCIP.2011.6115913.
 - [29] Yipeng Liu, Qi Yang, Yiling Xu, and Le Yang. Point cloud quality assessment: Dataset construction and learning-based no-reference metric. 12 2020. URL: <http://arxiv.org/abs/2012.11895>.

-
- [30] R. S. Pressman. *Software Engineering: A Practitioner's Approach*. Palgrave Macmillan, 2005.
- [31] Zhou Wang, Alan C. Bovik, and Ligang Lu. Why is image quality assessment so difficult? In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages IV–3313–IV–3316, 2002. doi:10.1109/ICASSP.2002.5745362.
- [32] Zhou Wang, H.R. Sheikh, and A.C. Bovik. No-reference perceptual quality assessment of jpeg compressed images. In *Proceedings. International Conference on Image Processing*, volume 1, pages I–I, 2002. doi:10.1109/ICIP.2002.1038064.
- [33] Zhou Wang, A.C. Bovik, and B.L. Evan. Blind measurement of blocking artifacts in images. In *Proceedings 2000 International Conference on Image Processing (Cat. No.00CH37101)*, volume 3, pages 981–984 vol.3, 2000. doi:10.1109/ICIP.2000.899622.
- [34] Kwan-Yee Lin and Guanxiang Wang. Hallucinated-iqa: No-reference image quality assessment via adversarial learning. URL: <https://kwanyeelin.github.io>.
- [35] Weixia Zhang, Kede Ma, Jia Yan, Dexiang Deng, and Zhou Wang. Blind image quality assessment using a deep bilinear convolutional neural network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30:36–47, 1 2020. doi:10.1109/TCSVT.2018.2886771.
- [36] Kede Ma, Wentao Liu, Tongliang Liu, Zhou Wang, and Dacheng Tao. Dipiq: Blind image quality assessment by learning-to-rank discriminable image pairs. *IEEE Transactions on Image Processing*, 26:3951–3964, 8 2017. doi:10.1109/TIP.2017.2708503.
- [37] Zicheng Zhang, Wei Sun, Xiongkuo Min, Tao Wang, Wei Lu, and Guangtao Zhai. No-reference quality assessment for 3d colored point cloud and mesh models. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(11):7618–7631, 2022. doi:10.1109/TCSVT.2022.3186894.
- [38] Wei Zhou, Qi Yang, Qiuping Jiang, Guangtao Zhai, and Weisi Lin. Blind quality assessment of 3d dense point clouds with structure guided resampling, 2022. arXiv:2208.14603.
- [39] Ziyu Shan, Qi Yang, Rui Ye, Yujie Zhang, Yiling Xu, Xiaozhong Xu, and Shan Liu. Gpa-net:no-reference point cloud quality assessment with multi-task graph convolutional network, 2023. arXiv:2210.16478.

-
- [40] Qi Yang, Yipeng Liu, Siheng Chen, Yiling Xu, and Jun Sun. No-reference point cloud quality assessment via domain adaptation, 2022. [arXiv:2112.02851](#).
 - [41] Zicheng Zhang, Wei Sun, Yucheng Zhu, Xiongkuo Min, Wei Wu, Ying Chen, and Guangtao Zhai. Treating point cloud as moving camera videos: A no-reference quality assessment metric, 2022. [arXiv:2208.14085](#).
 - [42] Zicheng Zhang, Wei Sun, Xiongkuo Min, Quan Zhou, Jun He, Qiyuan Wang, and Guangtao Zhai. Mm-pcqa: Multi-modal learning for no-reference point cloud quality assessment. 09 2022. doi:10.48550/arXiv.2209.00244.
 - [43] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 edition, 2010.
 - [44] John McCarthy. What is artificial intelligence? 01 2004.
 - [45] Yaser S. Abu-Mostafa, Malik Magdon-Ismael, and Hsuan-Tien Lin. *Learning From Data*. AMLBook, 2012.
 - [46] Tom M. Mitchell. Machine learning. In *Machine Learning*, chapter 1, page 2. McGraw-Hill, 1997.
 - [47] Oded Maimon and Lior Rokach, editors. *Data mining and knowledge discovery handbook*. Springer Science+Business Media, LLC, Boston, MA, 2 edition, 2010. URL: <https://doi.org/10.1007/978-0-387-09823-4>.
 - [48] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
 - [49] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015. doi:10.1038/nature14539.
 - [50] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117, jan 2015. URL: <https://doi.org/10.1016%2Fj.neunet.2014.09.003>, doi:10.1016/j.neunet.2014.09.003.
 - [51] C.M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, USA, 1995.
 - [52] Brian D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996. doi:11.1017/CB09780511812651.

-
- [53] Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989.
- [54] Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, and Kaori Togashi. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9(4):611–629, Aug 2018. doi:10.1007/s13244-018-0639-9.
- [55] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017. doi:10.1109/CVPR.2017.16.
- [56] Ruwen Schnabel and Reinhard Klein. Octree-based point-cloud compression. pages 111–120, 01 2006. doi:10.2312/SPBG/SPBG06/111-120.
- [57] 3D Slicer image computing plataform. <https://www.slicer.org/>. Visualizada en: 2023-06-01.