

Visión por Computador en Imágenes Médicas 3D

Carmen Azorín Martí, María Cribillés Pérez, Laura Lázaro
Soraluce, José Manuel Román Rodríguez

Universidad de Granada



**UNIVERSIDAD
DE GRANADA**

Tabla de Contenido

- 1 Preprocesamiento
- 2 Loss Functions
- 3 Evaluación
- 4 Modelos 3D
- 5 Modelos 2D
- 6 Experimentos
- 7 Conclusiones
- 8 Referencias

Conjunto de Datos: KiTS23

- Dataset: **KiTS23** (Kidney Tumor Segmentation Challenge 2023).
- Propósito:
 - Fomentar la investigación avanzada en segmentación de tumores renales.
 - Segmentar tumores renales y tejidos circundantes en tomografías computarizadas (CT).
- Características:
 - Incluye imágenes 3D preprocesadas.
 - Categorías en las máscaras:
 - 0: Fondo.
 - 1: Riñón.
 - 2: Cáncer.
 - 3: Quiste.

Preprocesamiento

- **Técnicas aplicadas:**

- Redimensionamiento de imágenes a $128 \times 128 \times 128$.
- Normalización: Z-Score.
- Codificación de máscaras como enteros.
- División de los datos en entrenamiento, validación y test.

- **3D:**

- Las imágenes se procesan en lotes adaptables.

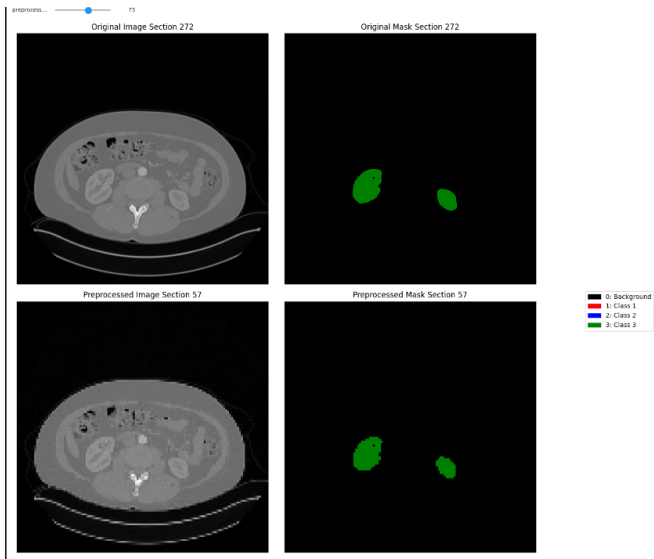
- **2D:**

- Cortes axiales equidistantes de imágenes 3D.
- Almacenados en formato '.npy' para manejar datos más fácilmente.

- **Ventaja de 2D:** Compatible con modelos bidimensionales.

- **Objetivo:** Optimizar memoria y mejorar el rendimiento del modelo.

Preprocesado 2D



Sobremuestreo

- **Motivación:**

- El dataset presenta un **desequilibrio extremo** entre clases.
- Riesgo: El modelo puede favorecer predicciones del fondo, afectando la precisión en clases relevantes.

- **Técnica:**

- Asignar pesos de muestreo a cada clase:

$$w_c = \frac{1}{n_c}$$

Donde n_c es el número de píxeles de la clase c .

- Peso para cada máscara:

$$w_{ejemplo} = \sum_{c \in \text{Clases presentes}} w_c \cdot n_c$$

- **Resultados:**

- Mejora significativa en la precisión durante el entrenamiento.
- **Problema:** Sobreajuste, ya que en validación y test no mejora notablemente.

Combined Loss: Dice Loss + Cross-Entropy

- **Combined Loss:** Dice Loss + Cross-Entropy
- **Focal Loss**
 - Focal Loss ponderada por clases
- **Class-Balanced Loss**

Focal Loss

- **Ventajas:**

- Diseñada para manejar el desbalance de clases.
- Se enfoca en ejemplos difíciles y reduce el peso de los bien clasificados.
- Penaliza más los errores en clases minoritarias y menos los ejemplos bien clasificados.

- **Fórmula:**

$$L_{Focal} = -\frac{1}{N} \sum_{t=1}^N \alpha_t (1 - p_t)^\gamma \log(p_t)$$

- α_t : Peso para manejar el desbalance entre clases.
- γ : Ajusta el enfoque en ejemplos difíciles ($\gamma > 0$).

Focal Loss ponderada por clases

- **Mejora de Focal Loss:**
 - α_t : Vector de pesos específico para cada clase.
 - Se adapta mejor a la frecuencia de las clases.
- **Ventaja:** Ajusta la importancia de las clases críticas (como tumor y quiste) sin ignorar la información global.
- **Aplicación:**
 - Implementada en SegResNet.
 - Comparada con CB Loss y Combined Loss.

Class-Balanced Loss (CB Loss)

- **Motivación:**

- Diseñada para mitigar el impacto del desbalance de clases.
- Usa la **frecuencia efectiva** para ajustar los pesos de cada clase:

$$\text{Frecuencia efectiva} = 1 - \beta^n$$

- **Peso por clase:**

$$w_c = \frac{1 - \beta}{1 - \beta^{n_c}}$$

Donde n_c es el número de ejemplos de la clase c .

- **Ventaja:** Mejora la segmentación de clases pequeñas como tumores y quistes al penalizar menos los píxeles de fondo.

Intersection over Union (IoU) y Coeficiente Dice

- Las métricas cuantifican la calidad de las segmentaciones realizadas por los modelos.
- Definición IoU:** Mide la superposición entre la región predicha (P_c) y la región verdadera (T_c) para una clase c .

- Fórmula:**

$$IoU_c = \frac{|P_c \cap T_c|}{|P_c \cup T_c|}$$

- Definición Dice:** Evalúa la similitud entre las regiones predichas y las reales, asignando más peso a la intersección.

- Fórmula:**

$$Dice_c = \frac{2|P_c \cap T_c|}{|P_c| + |T_c|}$$

- Rango:**

- $IoU, Dice = 1$: Coincidencia perfecta.
- $IoU, Dice = 0$: Sin superposición.

Modelos 3D

- **Objetivo:** Evaluar arquitecturas de redes neuronales convolucionales (CNN) para la segmentación de imágenes volumétricas 3D.
- Se implementaron y optimizaron tres modelos:
 - **U-Net 3D.**
 - **V-Net.**
 - **SegResNet.**
- **Ventaja de los modelos 3D:**
 - Capturan información espacial completa de los volúmenes.
 - Mejoran la precisión en tareas de segmentación médica compleja.

U-Net 3D vs V-Net

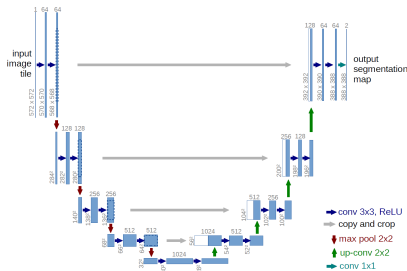
● U-Net 3D:

- Estructura en U, con ruta de contracción (encoder) y expansión (decoder).
- Conexiones salto preservan detalles finos durante la reconstrucción.
- Combina información global y local, ideal para bordes anatómicos.

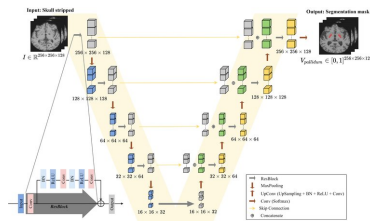
● V-Net:

- Diseñada para segmentación 3D médica.
- Usa convoluciones residuales que facilitan el entrenamiento profundo.
- Bloques residuales y activaciones paramétricas (*Parametric ReLU*) mejoran el aprendizaje.
- Robusta para clases pequeñas y segmentaciones complejas.

Comparación: U-Net vs V-Net



U-Net



V-Net

Figura: Comparación visual entre U-Net y V-Net.

SegResNet

- **Modelo especializado en datos volumétricos 3D.**
- Basado en convoluciones residuales con bloques de codificador y decodificador.
- **Características principales:**
 - **Fine-tuning:**
 - Las primeras capas del encoder se congelan durante el entrenamiento.
 - Solo se actualizan las capas del decoder.
 - Inicialización de pesos con *Xavier*, similar a U-Net 3D.
- **Ventaja:**
 - Equilibrio entre complejidad computacional y rendimiento.
 - Ideal para escenarios con recursos de cómputo limitados.

Modelos 2D

- **Motivación:**

- A pesar de la superioridad de los modelos 3D, los modelos 2D son útiles debido a:
 - Menor necesidad de datos.
 - Reducción de recursos computacionales.
 - Mayor cantidad de modelos preentrenados disponibles.

- **Estrategia:**

- Cortes axiales equidistantes de las imágenes 3D preprocesadas.
- Uso de arquitecturas 2D para segmentación.

U-Net con ResNet50 preentrenado

- **Configuración del modelo:**
 - Codificador basado en ResNet50, preentrenado en *ImageNet*.
 - Implementación con *Segmentation Models PyTorch*.
- **Optimización:**
 - Función de pérdida: **Focal Loss**.
 - Optimizador: Adam (learning rate = 1×10^{-4}).
- **Resultados:**
 - Buen desempeño en la segmentación del fondo.
 - Bajo rendimiento en clases minoritarias como tumor y quiste.
- **Limitación:** El desbalance de clases afecta el aprendizaje.

U-Net con oversampling

- **Estrategia aplicada:**

- Sobremuestreo de las clases minoritarias durante el entrenamiento.
- Creación de ejemplos adicionales de tumores y quistes.

- **Resultados esperados:**

- Mejora en la segmentación de las clases minoritarias.
- Reducción del impacto del desbalance de clases.

- **Resultados observados:**

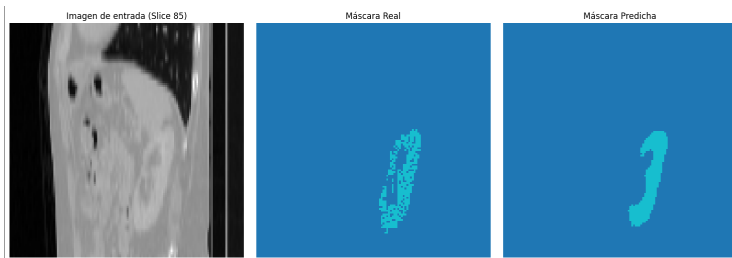
- Mejora significativa en el entrenamiento.
- Sin mejoras consistentes en validación y test (**sobreamoldamiento**).

U-Net con subsampling

- **Motivación:**
 - Reducir el peso de la clase dominante (*fondo*).
- **Técnica aplicada:**
 - Eliminación de imágenes con más del 99 % de píxeles pertenecientes al fondo.
- **Resultados:**
 - Reducción del sesgo hacia el fondo.
 - **Limitación:** Pérdida de datos valiosos, lo que afecta negativamente el rendimiento global.
- **Conclusión:** Técnica poco recomendable debido a la eliminación de información relevante.

Resultados con U-Net

- Se empleó U-Net con sobremuestreo y Combined Loss.
- Resultados principales:
 - **Clase 0 (fondo):** $\text{IoU} = 0.9936$, $\text{Dice} = 0.9968$.
 - **Clase 1 (riñón):** $\text{IoU} = 0.4639$, $\text{Dice} = 0.6274$.
 - **Clase 2 (tumor):** $\text{IoU} = 0.0245$, $\text{Dice} = 0.0454$.
 - **Clase 3 (quiste):** $\text{IoU} = 0.0004$, $\text{Dice} = 0.0001$.



Resultados con V-Net

- **Combined Loss:**

- Clase 0: IoU = 0.9931, Dice = 0.9966.
- Clase 1: IoU = 0.3743, Dice = 0.5395.
- Clase 2: IoU = 0.0442, Dice = 0.0713.
- Clase 3: IoU = 0.0000, Dice = 0.0001.

- **Focal Loss:**

- Clase 0: IoU = 0.9931, Dice = 0.9966.
- Clase 1: IoU = 0.3743, Dice = 0.5395.
- Clase 2: IoU = 0.0442, Dice = 0.0713.
- Clase 3: IoU = 0.0000, Dice = 0.0001.

- **CB Loss:**

- Clase 0: IoU = 0.9938, Dice = 0.9969.
- Clase 1: IoU = 0.4118, Dice = 0.5773.
- Clase 2: IoU = 0.0934, Dice = 0.1546.
- Clase 3: IoU = 0.0000, Dice = 0.0000.

Resultados con V-Net



Figura: V-Net CBLoss

Resultados con SegResNet: Parte 1

● Combined Loss:

- Clase 0: IoU = 0.9927, Dice = 0.9963.
- Clase 1: IoU = 0.4130, Dice = 0.5728.
- Clase 2: IoU = 0.0107, Dice = 0.0175.
- Clase 3: IoU = 0.0000, Dice = 0.0000.

● Focal Loss:

- Clase 0: IoU = 0.9933, Dice = 0.9967.
- Clase 1: IoU = 0.4220, Dice = 0.5778.
- Clase 2: IoU = 0.0928, Dice = 0.1462.
- Clase 3: IoU = 0.0000, Dice = 0.0000.

Resultados con SegResNet: Parte 2

● CB Loss:

- Clase 0: IoU = 0.9936, Dice = 0.9968.
- Clase 1: IoU = 0.4396, Dice = 0.5938.
- Clase 2: IoU = 0.0861, Dice = 0.1339.
- Clase 3: IoU = 0.0000, Dice = 0.0000.

● Focal Loss Weighted:

- Clase 0: IoU = 0.9883, Dice = 0.9941.
- Clase 1: IoU = 0.3480, Dice = 0.5142.
- Clase 2: IoU = 0.1433, Dice = 0.2344.
- Clase 3: IoU = 0.0033, Dice = 0.0065.

Comparación de Modelos

- **SegResNet con CB Loss** destaca por su rendimiento consistente en validación y test.
- Ventajas:
 - Uso de fine-tuning para mejorar la generalización.
 - Balance entre complejidad computacional y desempeño.
- Limitaciones:
 - Clases minoritarias (tumores y quistes) aún presentan bajos resultados.

Conclusiones

- Las arquitecturas 3D (SegResNet, U-Net 3D, V-Net) muestran mejor desempeño en segmentación médica al aprovechar información espacial completa.
- SegResNet con CB Loss destaca por sus resultados consistentes tras el fine-tuning.
- Persisten desafíos en el manejo de clases minoritarias (tumores y quistes), afectando las métricas como Dice e IoU.
- Las arquitecturas 2D, aunque más eficientes computacionalmente, pierden información crucial y muestran menor rendimiento.
- Futuras investigaciones deben centrarse en estrategias para mitigar el desbalance de clases y mejorar la generalización.

Referencias

- Joseph Chen and Benson Jin. *A 2D U-Net for Automated Kidney and Renal Tumor Segmentation*, 2019.
- Nicholas Heller et al. *The KiTS21 Challenge: Automatic Segmentation of Kidneys, Renal Tumors, and Renal Cysts*, 2023.
- Xin Wang et al. *2.75D: Boosting Learning by Representing 3D Medical Imaging to 2D Features for Small Data*, 2023.
- Michael Yeung et al. *Unified Focal Loss: Generalising Dice and Cross Entropy-Based Losses to Handle Class Imbalanced Medical Image Segmentation*, 2022.