# Exploración Interactiva Acelerada de Volúmenes Médicos mediante Embedding de Características UMAP en GPU

Víctor Germán Quintero Toro
Asesor: Andrés Marino Álvarez Meza, PhD
viquinterot@unal.edu.co

Departamento de Ingeniería Eléctrica, Electrónica y Computación
Universidad Nacional de Colombia - Sede Manizales

18 de julio de 2025

**Resumen**

La visualización de datos médicos volumétricos, como los de tomografía computarizada (TC), es fundamental para el diagnóstico clínico. La Renderización Directa de Volumen (DVR) depende de Funciones de Transferencia (TF) que asignan propiedades ópticas a los vóxeles. Este trabajo presenta un pipeline computacional que permite la exploración interactiva de volúmenes médicos mediante la creación de un espacio de características 2D a partir de datos de alta dimensionalidad. Utilizamos el algoritmo Uniform Manifold Approximation and Projection (UMAP), acelerado en GPU a través de la suite NVIDIA RAPIDS, para generar una incrustación 2D que preserva la estructura de los datos. El sistema permite a un usuario seleccionar una Región de Interés (ROI) en el espacio de características 2D, actualizando en tiempo real la TF del render 3D para aislar visualmente las estructuras anatómicas correspondientes. Este enfoque, implementado en Python con las librerías `cuML` y `Vedo`, demuestra la viabilidad de crear herramientas de exploración de datos médicos potentes e interactivas que superan los cuellos de botella computacionales tradicionales.

## 1. Introducción

La interpretación de imágenes médicas 3D es una tarea compleja que requiere herramientas de visualización avanzadas [1]. La Renderización Directa de Volumen (DVR) es una técnica poderosa que visualiza todo el conjunto de datos sin necesidad de una segmentación explícita previa. El principal desafío en DVR radica en el diseño de una **Función de Transferencia (TF)** efectiva, que asigna color y opacidad a cada vóxel basándose en sus propiedades. Tradicionalmente, las TFs se basan en la intensidad del vóxel, pero los enfoques modernos utilizan espacios de características multidimensionales (por ejemplo, gradiente, Laplaciano) para diferenciar mejor entre tejidos con intensidades similares [2].

Sin embargo, explorar estos espacios de características de alta dimensionalidad es intratable para un humano. Por ello, las técnicas de **reducción de dimensionalidad** como t-SNE o UMAP son cruciales para proyectar las características a un espacio 2D manejable [3]. El problema es que estos algoritmos pueden ser computacionalmente intensivos, limitando la interactividad. El trabajo de [2] avanza en esta línea con t-SNE, pero la búsqueda de mayor velocidad y una mejor preservación de la estructura global sigue abierta.

Este trabajo presenta una solución a este problema mediante un pipeline acelerado por GPU que implementa los siguientes pasos, basados en la implementación de [4]:

1. Extracción de un conjunto de características simplificado para cada vóxel.
2. Uso de UMAP, ejecutado en la GPU con `cuML` [5], para generar un embedding 2D.
3. Desarrollo de una interfaz interactiva de doble panel que vincula el espacio de características 2D con el render del volumen 3D para el diseño dinámico de TFs.

El objetivo es demostrar un prototipo que permita una exploración fluida y en tiempo real, superando las barreras de rendimiento de los enfoques basados en CPU.

# 2. Metodología

El pipeline propuesto se divide en tres etapas principales: extracción de características, reducción de dimensionalidad acelerada por GPU y visualización interactiva, todo implementado en el prototipo funcional [4].

## 2.1. Extracción de Características

Para cada vóxel del volumen 3D, cargado desde una secuencia de archivos TIF, se extrae un vector de 3 características clave para describir tanto su valor como su contexto local:

- **Intensidad original:** El valor escalar del vóxel.
- **Magnitud del Gradiente Gaussiano:** Estima la tasa de cambio de intensidad, útil para detectar bordes. Se calcula usando un $\sigma$ de 1.0.
- **Laplaciano de Gaussiano (LoG):** Detecta regiones de cambio rápido y es útil para encontrar texturas. Se calcula con un $\sigma$ de 1.0.

Estas características se apilan en una matriz donde cada fila representa un vóxel y cada columna una característica.

## 2.2. Reducción de Dimensionalidad con UMAP en GPU

Para superar el principal cuello de botella computacional, utilizamos la implementación de **UMAP** de la librería `cuML` de NVIDIA RAPIDS [5]. El proceso es el siguiente:

1. Las características extraídas se transfieren a la memoria de la GPU como un arreglo `cupy`.

2. Se aplica un escalado estándar en la GPU usando `cuml.StandardScaler`.

3. Se ejecuta `cuml.UMAP` sobre los datos escalados para obtener el embedding 2D. Se configuraron los hiperparámetros `n_neighbors=30` y `min_dist=0.1` para balancear la preservación de la estructura.

Este enfoque minimiza la transferencia de datos entre CPU y GPU, resultando en una aceleración drástica.

### 2.3. Visualización y Diseño de TF Interactivo

La visualización se construye con la librería **Vedo**, creando una aplicación de dos paneles [4]. La interactividad se logra mediante una función de callback que, al hacer clic en el espacio UMAP, define una **Región de Interés (ROI)**. El sistema entonces identifica los vóxeles dentro del ROI y actualiza la TF del volumen 3D en tiempo real, asignando un color y opacidad distintivos a la selección, similar a los enfoques de renderizado neural implícito interactivo [6].

## 3. Resultados y Discusión

Los resultados son cualitativos y demuestran la eficacia del pipeline interactivo. Al ejecutar el script en un conjunto de datos de TC de cabeza humana:

- **Rendimiento:** El cálculo del embedding UMAP para millones de vóxeles se completa en segundos, y la actualización de la TF es instantánea.
- **Separación de Estructuras:** UMAP logra una excelente separación de estructuras (hueso, tejido blando, aire) en clústeres visualmente definidos.
- **Exploración Intuitiva:** La selección de clústeres en el espacio 2D permite aislar estructuras anatómicas en el render 3D de forma inmediata, validando la hipótesis de que un espacio de características bien formado facilita la segmentación visual.

La combinación de UMAP con la aceleración en GPU demuestra ser una estrategia poderosa, transformando el análisis de características en parte de un bucle de visualización interactivo.

## Conclusión y Trabajo Futuro

Hemos desarrollado y demostrado con éxito un prototipo para la exploración interactiva de volúmenes médicos basado en un embedding de características acelerado por GPU. El uso de `cuML UMAP` y `Vedo` permite un rendimiento en tiempo real que no sería posible con herramientas basadas únicamente en CPU.

Como **trabajo futuro**, la línea de investigación más prometedora es la portabilidad de este desarrollo hacia el **Edge Computing**. El objetivo principal es adaptar y optimizar el pipeline computacional para su ejecución en hardware embebido de bajo consumo y costo. Esto incluye plataformas como las tarjetas de la serie `NVIDIA Jetson`, que cuentan con GPUs integradas compatibles con CUDA, o arquitecturas emergentes como RISC-V, representadas por tarjetas como las de la familia `Lichee Pi`.

Llevar esta capacidad de visualización avanzada al borde de la red (the edge) permitiría la creación de herramientas de diagnóstico portátiles y asequibles. Esta estrategia podría democratizar el acceso a la tecnología de renderizado 3D interactivo en entornos clínicos con recursos limitados, consultorios móviles o para la formación médica en cualquier lugar, desacoplando la dependencia de estaciones de trabajo de alto rendimiento.

# Referencias


[1] L. Wang, G. Zhai, K. Gu, and X. Liu, "A review of medical image visualization," *Biomedical Signal Processing and Control*, vol. 85, p. 104885, 2023.

[2] W. Serna-Serna, A. M. Álvarez-Meza, and Á. Orozco-Gutiérrez, "Fast semi-supervised t-sne for transfer function enhancement in direct volume rendering-based medical image visualization," *Mathematics*, vol. 12, no. 12, p. 1885, 2024.

[3] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[4] T. Nombre], "head4d.py: Interactive dvr with umap and vedo," 2025. Código fuente del proyecto, basado en la guía del curso Teoría de Aprendizaje de Máquina.

[5] J. P. Adams and M. R. Weldon, "Accelerating scientific workflows with nvidia rapids cudf and jax," in *Proceedings of the 23rd Python in Science Conference*, pp. 91–100, 2024.

[6] P. Schlegel, C. Sormann, T. Hädrich, E. Zell, B. Kainz, and D. Schmalstieg, "Neural implicit representations for interactive direct volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 1, pp. 127–137, 2023.