

Desarrollo e Implementación de Software para la Reconstrucción 3D Mediante Superficies y Análisis de Fracturas Vertebrales en Tomografía Computarizada (CT)

TOMÁS CASTILLO¹

¹Pontificia Universidad Católica de Chile (e-mail: toms.castilloe@uc.cl)

• **ABSTRACT** En el área de imágenes médicas, la segmentación es una herramienta clave para el análisis de regiones de interés (ROI), como la evaluación de fracturas de compresión vertebral (VCFs). Este artículo presenta un avance en el desarrollo de un software que permite la segmentación y reconstrucción tridimensional de vértebras a partir de imágenes de tomografía computarizada (CT). La actualización del programa permite seleccionar puntos a través de diferentes cortes coronales y generar superficies utilizando interpolación lineal, optimizando la visualización y análisis de las vértebras. Además, se implementa la medición de alturas utilizando análisis de componentes principales (PCA) para evaluar la estructura tridimensional de las vértebras. Este desarrollo tiene aplicaciones potenciales en el diagnóstico y manejo de VCFs, facilitando el trabajo del personal médico en la evaluación de pacientes y en la docencia médica.

• **INDEX TERMS** Imágenes médicas, Segmentación, Fracturas de compresión vertebral, Tomografía computarizada (CT), Interpolación lineal, Análisis de componentes principales (PCA), Reconstrucción tridimensional, Software médico.

I. INTRODUCCIÓN

EN el área de imágenes médicas es posible encontrarse con distintos tipos de procesamiento para analizar órganos o región de interés (ROI), una de las más importantes es la segmentación. La segmentación tiene que ver con como recortamos la región de interés para una posterior interpretación y/o procesamiento. Esta herramienta nos permite identificar aislar áreas de una imagen para, en relación al proyecto que se está realizando, aplicar criterios diagnósticos que se ocupen para identificar fracturas de compresión de cuerpos vertebrales. Al menos dos tercios de pacientes con fracturas de compresión vertebral (VCFs) son diagnosticados incidentalmente [4]. Los pacientes sintomáticos pueden presentar dolor de espalda y fractura tal que se demuestra en radiografías, y ocurre más comúnmente entre T8 y L4. [4]. Los VCF sintomáticos a menudo conducen a lesiones espinales graves, dolor, deformidad de la columna, disminución de la movilidad y disminución de la función pulmonar, y puede aumentar el riesgo de mortalidad en relación a la edad. Para disminuir esta carga, La prevención y el manejo basados en

evidencia son esenciales [5].

Para ello, el presente artículo da cuentas del avance en el proyecto enfocado en reconocer las fracturas de vertebras de imágenes de tomografías computarizadas (CT). En una edición anterior, se comentó sobre la segmentación manual de las imágenes a través de distintos cortes en el eje coronal para luego hacer una interpolación mediante B-Splines para crear la curva que se escoge con puntos de forma manual. Ahora, se presenta una actualización de esto, pudiendo el programa deslizarse a través de distintos corte, seleccionando puntos en cada uno de ellos y pudiendo crear superficies que finalmente crearán figuras tridimensionales de las vertebras para su posterior análisis.

Este tipo de softwares es importante en la industria pues permite al personal médico poder realizar inspecciones y sacar diagnóstico médicos en la evaluación de alguna ROI segmentando esta zona para aplicar otros algoritmos para la evaluación de parámetros clínicos importantes para dar declaraciones concluyentes en el proceso de evaluación de

pacientes [2]. Incluso permite ser de ayuda para la docencia médica misma [1]. Además de que se podría aportar a los softwares de libre desarrollo debido a que la implementación se realizó en un lenguaje de gratis uso como lo es Python [3]. A continuación se detalla el proceso de desarrollo y utilización del programa.

II. MÉTODOS

Los datos utilizados fueron entregados por el curso, en donde se nos entregaron dos carpetas con archivos de imágenes tipo DICOM de dos pacientes, el primero tenía alrededor de 611 archivos mientras que el segundo 711.

A. SELECCIÓN DE CORTE EN VISTA CORONAL

Este punto se abordó en la versión anterior. La única modificación es que se puede no congelar el corte que se analiza, ya que es necesario seleccionar puntos en distintos cortes. Para ello, se puede mover el selector de barrido y elegir los contornos a analizar.

B. SELECCIÓN DE PUNTOS ARBITRARIOS SOBRE CONTORNO DE VERTEBRAS

Con respecto a la edición anterior, para la elección de puntos debemos presionar la tecla "t" y podremos seleccionar con click izquierdo la región de interés en el corte en el que se presiona la tecla. Ahora también podemos mover el seleccionador para escoger el corte en que se quiere trabajar y posteriormente podremos seguir añadiendo puntos en este corte si mantenemos el seleccionador en el mismo mientras se realiza la acción. De esta manera se irán guardando los puntos en función del corte en que se hizo y en la región XY del espacio de la imagen para la posterior construcción de superficie.

Para una visualización inmediata se implemento unos puntos de prueba determinados, que recorren 50 cortes de forma aproximada en la vista coronal, para cierta vertebra de interés tanto en el paciente 1 como en el paciente 2. Para implementarlos el software preguntará si se quieren utilizar los puntos predeterminados para generar las superficies sobre una cantidad de puntos que fue seleccionada previamente como forma de presentación del programa.

C. CONSTRUCCIÓN DE SUPERFICIES MEDIANTE INTERPOLACIÓN LINEAL

Para una construcción robusta del código, se optó por el uso de interpolación lineal y la función de *griddata()* de la librería de *scipy*. Esto pues en un principio se trató de utilizar la definición de superficies B-Spline de producto tensorial pero estas fallaban cuando se encontraba datos irregulares. En este contexto, estos datos son los que el usuario elegía, y para que funcionara este método se debían elegir puntos regulares entre cada corte de la vista coronal, basandome completamente en lo pasado en clases, pues *griddata()* de todas formas tiene una modalidad de interpolación de B-

Splines que deja curvas suaves en la superficie.

griddata() funciona tal que toma un conjunto de puntos en 2D o 3D (o en dimensiones superiores) con sus respectivos valores y los interpola sobre una cuadrícula definida por el usuario. La función esencialmente construye una superficie que pasa a través o cerca de los puntos dados, y luego evalúa los valores de esa superficie en los puntos de la cuadrícula. Es especialmente útil cuando se tiene un conjunto disperso de puntos en un espacio y se desea estimar los valores en puntos de una cuadrícula regular. Así, la función se utiliza bastante en la visualización de superficies en donde los datos no están dispuestos de manera estructurada en la mayoría de los casos.

Luego, para una mejor visualización de la vertebra elegida, se optó por una interpolación lineal. Esto pues si se utilizaban las aproximaciones por B-Splines las curvas suaves no permitían reflejar de buena manera la estructura de la vertebra elegida. El método de interpolación lineal es el más común y el predeterminado en *griddata()*. La interpolación lineal divide el espacio en simplex (triángulos en 2D o tetraedros en 3D) y estima el valor en un punto x dentro de un simplex usando una combinación lineal de los valores en los vértices del simplex.

Dado un conjunto de puntos P_1, P_2, \dots, P_n y sus valores asociados $f(P_1), f(P_2), \dots, f(P_n)$, se construyen simplex que cubren el dominio de los datos [6]. Para un punto x que cae dentro de un simplex específico $[P_i, P_j, P_k]$ (en 2D, por ejemplo), el valor $f(x)$ se calcula como:

$$f(x) = \lambda_1 f(P_i) + \lambda_2 f(P_j) + \lambda_3 f(P_k)$$

donde $\lambda_1, \lambda_2, \lambda_3$ son las coordenadas baricéntricas de x respecto a los vértices P_i, P_j, P_k . Estas coordenadas se calculan basándose en las distancias relativas de x a cada vértice, asegurando que la suma de las coordenadas baricéntricas sea 1, ($\lambda_1 + \lambda_2 + \lambda_3 = 1$) [6].

D. MEDICIÓN DE ALTURA: EJES IZQUIERDA-DERECHA Y ANTERIO-POSTERIOR DE LOS CUERPOS VERTEBRALES

Para la implementación de la medición de alturas en estos ejes se ocupó la función *contourf()* que permite generar un mapa de contorno de una superficie bidimensional basada en la densidad de puntos. Esta función toma una malla de puntos que representa las coordenadas (x, y, z) y genera contornos de niveles en la superficie.

Con el mapa de contorno generado, se puede medir la variabilidad en los ejes izquierda-derecha y antero-posterior, observando cómo cambian los niveles de contorno a lo largo de estas direcciones. Esto es útil para cuantificar la altura de la vértebra en distintas regiones y evaluar deformidades o variaciones morfológicas en la estructura vertebral. En otras palabras, la función nos permite visualizar la vertebra como

si sacáramos una fotografía en la visualización tridimensional en las direcciones de los ejes mencionados.

E. MEDICIÓN DE ALTURA: ANÁLISIS DE COMPONENTES PRINCIPALES (PCA)

Para esta medición de alturas, la PCA se utiliza para medir las alturas de las superficies generadas anteriormente. Primero, las coordenadas de la superficie (x, y, z) se apilan en una matriz de forma $(N, 3)$, donde cada fila es un punto tridimensional de la superficie vertebral. Luego, la PCA se aplica solo a las coordenadas en el plano XY. El objetivo aquí es encontrar las direcciones principales de variabilidad en ese plano, que corresponden a las dos componentes principales.

El procedimiento sería el siguiente. Primero, se calcula el eje que maximiza la varianza en los puntos proyectados en el plano XY. Este eje es considerado la "longitud" de la superficie en ese plano. Posterior a esto, se construye una matriz de rotación para alinear el eje principal con el eje X para facilitar la medición de la altura. Una vez alineados los puntos, se toma la coordenada Z de los puntos rotados y se reestructura para que coincida con la forma original de la superficie. Esto permite evaluar la variabilidad en Z en la superficie alineada. La altura de la vértebra se calcula como la diferencia entre el valor máximo y mínimo de Z en la superficie rotada. Luego, se convierte a milímetros multiplicando por el espaciado de píxeles en la dirección Z. Es importante mencionar que el eje ortogonal al eje principal es fundamental para orientar la estructura de la vértebra, este se puede considerar como la dirección que define el ancho o la estructura secundaria de la vértebra en ese plano.

III. RESULTADOS

En este apartado se mostrarán imágenes de la paciente - 2. Lo que se observará serán los puntos predeterminados que se mencionaban con anterioridad, para poder dar más importancia a lo nuevo de esta edición del software sobre la creación de superficies y cálculo de alturas.

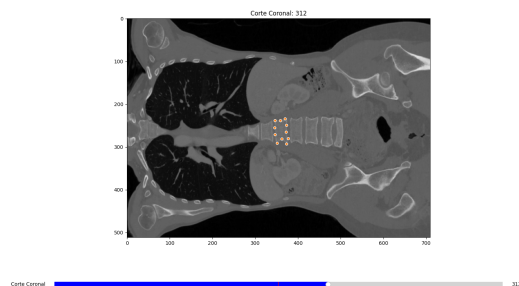


FIGURA 1: Imagen referencial de puntos predeterminados que seleccionan una vértebra de interés en paciente 2

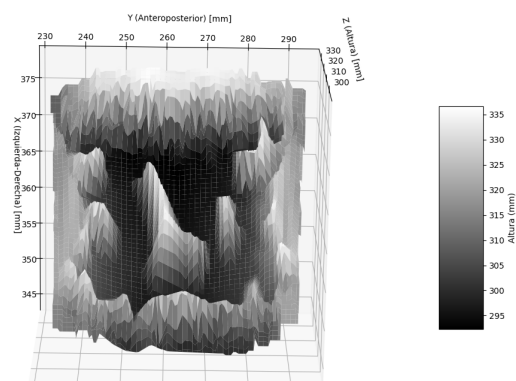


FIGURA 2: Vértebra reconstruida en 3D con interpolación lineal de paciente 2

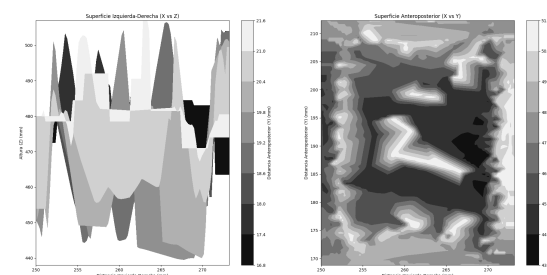


FIGURA 3: Gráficos de alturas de izquierda-derecha y antero-posterior para vértebra de interés del paciente 2 respectivamente

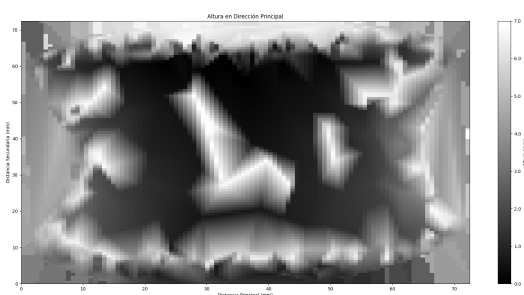


FIGURA 4: Gráficos de alturas utilizando el análisis de componentes principales (PCA) para vértebra de interés del paciente 2

IV. DISCUSIÓN

Si bien no se logró una implementación totalmente “manual” pues se hizo uso de librerías externas, el tratar de utilizar la modelación matemática de todas las formas ayudó a identificar fallencias de la formulación propia difíciles de manejar computacionalmente, al menos en esta fase de desarrollo, y que afectaban la robustez del software. Aun así, fue posible crear superficies que permitieron reconstruir en 3D una vértebra, evidenciando los surcos característicos de los cuerpos vertebrales. Limitaciones de esto claramente

es la cantidad de puntos por corte que elige el usuario, pues entre mayor cantidad de puntos tanto por corte como abarcando una mayor profundidad (es decir más cortes), será posible caracterizar de mejor forma la vertebra escogida. Además, la “calidad” de la vertebra dependerá de la forma de interpolación, pues como se comentaba antes en esta ocasión se eligió la más simple pues la que utilizaba las curvas suave de B-Splines no permitía distinguir de buena manera la figura de la vertebra.

Con respecto a las alturas, la imagen de alturas izquierda-derecha de la Figura 3, se ve las alturas que le otorga tridimensionalidad a la Figura 2. La diferencia de alturas nos dice que hasta esta elección de puntos tenemos 5mm de espesor de vertebra seleccionados, que tiene sentido con la cantidad de cortes trabajados. Si se hubieran escogido más, tendríamos una variación mayor de valores. Luego, en la altura calculada en sentido anteroposterior, podemos analizar que hay componentes que sobresalen de la vertebra que se registran con alturas mayores y se ven como las curvas de nivel de la vertebra, que están asociadas a las características de esta. Y en la Figura 4 se hacen mas notorios los contornos que sobresalen de las alturas de la imagen anteroposterior, y la dimensionalidad es la misma si se calcula con la barra de grises que acompaña a las Figuras.

Esta nueva implementación nos aporta información relevante para poder hacer diagnósticos claros sobre posibles fracturas en los pacientes que se analizan, pudiendo ver las irregularidades que nacen en la generación de superficies de las vertebras, en donde posiblemente aparecerán parámetros fisiológicos que nos indicarán si hay o no problemas en la región que se analiza.

V. CONCLUSIÓN

El desarrollo de este software ha permitido avances significativos en la segmentación y análisis de imágenes médicas, específicamente en la reconstrucción tridimensional de vértebras. La implementación de interpolación lineal para la generación de superficies, así como el uso de PCA para la medición de alturas, ofrece una herramienta robusta para la evaluación de fracturas de compresión vertebral. Las pruebas realizadas con imágenes de pacientes demuestran que el software puede identificar irregularidades en la estructura vertebral que son clave para el diagnóstico de VCFs. No obstante, se identificaron limitaciones relacionadas con la cantidad y calidad de puntos seleccionados, además de la instrucción propia del usuario que utilice este programa, lo que impacta la precisión de las superficies generadas. A pesar de estas limitaciones, este software representa un paso importante en la integración de herramientas de libre uso, como Python, para el desarrollo de soluciones médicas accesibles y de alta utilidad en la práctica clínica y en la educación médica.

VI. REFERENCIAS

- [1] I. Calvo Bragado *et al.*, “Estudio comparativo de software de segmentación y postprocesado de imágenes médicas para impresión 3d en docencia médica,” 2023.
- [2] A. A. G. Flores, A. M. Buesaquillo, D. I. A. Rios, *et al.*, “La importancia crucial de la imageneologia en el diagnostico de patología craneales avances tecnológico y contribuciones clínicas,” *Tesla Revista Científica*, vol. 4, no. 1, e323–e323, 2024.
- [3] F. J. Marín Marín *et al.*, “Segmentación de imágenes médicas en software libre,” 2015.
- [4] J. McCARTHY and A. Davis, “Diagnosis and management of vertebral compression fractures,” *American family physician*, vol. 94, no. 1, pp. 44–50, 2016.
- [5] P. C. Parreira, C. G. Maher, R. Z. Megale, L. March, and M. L. Ferreira, “An overview of clinical guidelines for the management of vertebral compression fracture: A systematic review,” *The Spine Journal*, vol. 17, no. 12, pp. 1932–1938, 2017.
- [6] T. Warburton, “An explicit construction of interpolation nodes on the simplex,” *Journal of engineering mathematics*, vol. 56, pp. 247–262, 2006.

VII. ANEXO

El siguiente repositorio tiene el código del software hecho.

...