

# Modelado tridimensional de estructuras anatómicas heterogéneas



# Modelado tridimensional de estructuras anatómicas heterogéneas

Miller Gómez Mora





UNIVERSIDAD DISTRITAL  
FRANCISCO JOSÉ DE CALDAS

**UD**  
Editorial

**E2**  
ESPACIOS

© Universidad Distrital Francisco José de Caldas  
© Centro de Investigaciones y Desarrollo Científico  
© Miller Gómez Mora  
Primera edición, abril de 2020  
ISBN: 978-958-787-173-9

Dirección Sección de Publicaciones

Rubén Eliécer Carvajalino C.

Coordinación editorial

Nathalie De la Cuadra N.

Corrección de estilo

John Fredy Guzmán

Montaje de cubierta

Astrid Pietro

Imagen de cubierta

Miller Gómez Mora

Editorial UD

Universidad Distrital Francisco José de Caldas

Carrera 24 No. 34-37

Teléfono: 3239300 ext. 6202

Correo electrónico: [publicaciones@udistrital.edu.co](mailto:publicaciones@udistrital.edu.co)

Gómez Mora, Miller

Modelado tridimensional de estructuras anatómicas heterogéneas / Miller Gómez Mora. -- Bogotá: Universidad Distrital Francisco José de Caldas, 2020.

142 páginas; 24 cm.

Incluye referencias bibliográficas.

ISBN 978-958-787-173-9

1. Sistemas de imágenes en medicina -- Modelos geométricos 2. Medición de superficies -- Modelos geométricos 3. Modelado tridimensional I. Tít. 516.0228 cd 22 ed.

A1660083

CEP-Banco de la República-Biblioteca Luis Ángel

Todos los derechos reservados.

Esta obra no puede ser reproducida sin el permiso previo escrito de la Sección de Publicaciones de la Universidad Distrital.

Hecho en Colombia

# Contenido

---

<b>1.</b>	<b>Introducción</b>	<b>15</b>
<b>2.</b>	<b>Imágenes, métodos y herramientas</b>	<b>21</b>
<b>2.1.</b>	<b>Imágenes médicas</b>	<b>21</b>
2.1.1.	Imágenes TAC	23
2.1.2.	Segmentación de imágenes TAC	24
2.1.2.1.	<i>Umbralización</i>	25
2.1.2.2.	<i>Crecimiento de regiones</i>	26
<b>2.2.</b>	<b>Reconstrucción geométrica de superficies</b>	<b>28</b>
2.2.1.	Métodos paramétricos	29
2.2.2.	Métodos de geometría computacional	30
2.2.3.	Métodos implícitos	31
2.2.4.	Aproximación por mínimos cuadrados	36
2.2.5.	Estimación de la normal mediante PCA	38
<b>2.3.</b>	<b>Extracción de la geometría de las regiones</b>	<b>40</b>
2.3.1.	Marching Cubes	40
2.3.2.	Superficies paramétricas	42
2.3.3.	Superficies triangulares	43
2.3.4.	Superficies implícitas	43
<b>2.4.</b>	<b>Extracción de la geometría de las estructuras heterogéneas</b>	<b>44</b>
2.4.1.	Extensión del algoritmo Marching Cubes	44
2.4.2.	Renamiento de Delaunay	44
2.4.3.	Métodos de ensamble	46

<b>3. Método de modelado RAM</b>	<b>49</b>
<b>3.1. Modelado matemático</b>	<b>49</b>
<b>3.2. Proceso de reconstrucción implícita</b>	<b>51</b>
3.2.1. Extracción de regiones	52
3.2.2. Generación de la nube de puntos	52
3.2.3. Cálculo de la función implícita	56
3.2.4. Creación de los vectores VFI y VFC	57
<b>3.3. Proceso de extracción de la geométrica</b>	<b>60</b>
3.3.1. Malla de superficie	61
3.3.2. Malla volumétrica	63
<b>4. Experimentos</b>	<b>69</b>
<b>4.1. Entorno experimental</b>	<b>69</b>
<b>4.2. Modelado de regiones anatómicas</b>	<b>70</b>
4.2.1. Regiones anatómicas simples	71
4.2.2. Regiones anatómicas complejas	74
<b>4.3. Modelado de estructuras anatómicas heterogéneas</b>	<b>77</b>
4.3.1. Estructuras heterogéneas con regiones colindantes	78
4.3.2. Estructuras heterogéneas con regiones anidadas	81
4.3.3. Estructuras heterogéneas combinadas	83
<b>5. Análisis de resultados</b>	<b>85</b>
<b>5.1. Tiempo computacional</b>	<b>85</b>
<b>5.2. Precisión de los modelos</b>	<b>92</b>
<b>5.3. Calidad geométrica</b>	<b>98</b>
5.3.1. Coeficiente de Dice	98
5.3.2. Distancia de Hausdorff	102
<b>6. Conclusiones</b>	<b>105</b>
<b>7. Apéndice A. Conceptos relevantes</b>	<b>109</b>
<b>7.1. Conceptos topológicos</b>	<b>109</b>
7.1.1. Espacio topológico	109
7.1.2. Variedad	110

7.1.3. Partición	111
7.1.4. Vecindad	112
7.1.5. Conectividad	113
<b>7.2. Curvas y supercicies</b>	<b>113</b>
7.2.1. Curvas implícitas en 2D	113
7.2.2. Gradiente	115
7.2.3. Súpercicies en 3D	116
7.2.3.1. <i>Súperficie paramétrica</i>	116
7.2.3.2. <i>Súperficie implícita</i>	116
7.2.4. Vector normal a la súperficie	117
<b>7.3. Reconstrucción tomográfica</b>	<b>117</b>
7.3.1. La transformada de Radon	117
7.3.2. La transformada de Fourier	119
7.3.3. Coeficiente de atenuación	123
7.3.4. Adquisición de las proyecciones tomográficas	124
<b>7.4. Estructuras de datos espaciales</b>	<b>125</b>
7.4.1. Arreglo de etiquetas	125
7.4.2. Estructuras jerárquicas	126
7.4.3. Grafo de adyacencia de regiones	127
<b>Bibliografía</b>	<b>131</b>

# Lista de figuras

---

1.1.	Modelado de un objeto heterogéneo compuesto de tres partes	15
1.2.	Escáneres para capturar los puntos de supercie de un objeto	17
1.3.	Captura y segmentación de una imagen TAC	18
1.4.	Creación manual de una malla de supercie a partir de una imagen TAC	19
1.5.	Generación automática de la malla de supercie	20
2.1.	Imágenes de dos dimensiones	22
2.2.	Proceso para la captura de una imagen TAC	23
2.3.	Método de umbralización	26
2.4.	Aplicación del método de crecimiento de regiones	27
2.5.	El vecindario en 3D de un punto p (rojo)	28
2.6.	Partición de la unidad multinivel en 1D	33
2.7.	Método de reconstrucción de supercies de Poisson	35
2.8.	Algoritmo Marching Cubes	41
2.9.	Triangulación de Delaunay en 2D y 3D	45
2.10.	Modelado de un objeto sólido a partir de un árbol booleano	46
2.11.	Diferentes resultados de aplicar la operación booleana de unión	47
3.1.	Regiones segmentadas a partir de una imagen TAC	52
3.2.	Generación de la nube de puntos a partir de la segmentación de un hígado	53

<b>3.3.</b>	<b>Efectos negativos de normales mal orientadas</b>	<b>55</b>
<b>3.4.</b>	<b>VFI en el modelado de una rodilla con cuatro regiones</b>	<b>59</b>
<b>3.5.</b>	<b>Malla de supercie generada por el algoritmo Marching Cubes</b>	<b>62</b>
<b>3.6.</b>	<b>Malla de supercie generada por el m<o>odo RAM</o></b>	<b>63</b>
<b>3.7.</b>	<b>Generaci<o>n de mallas volumétricas</o></b>	<b>64</b>
<b>3.8.</b>	<b>Malla volumétrica de una estructura anatómica heterogénea</b>	<b>66</b>
<b>4.1.</b>	<b>Segmentación de regiones simples</b>	<b>71</b>
<b>4.2.</b>	<b>Segmentación de regiones complejas</b>	<b>71</b>
<b>4.3.</b>	<b>Mallas de supercie ajustadas a las funciones implícitas</b>	<b>72</b>
<b>4.4.</b>	<b>Corte vertical translúcido de las mallas volumétricas</b>	<b>73</b>
<b>4.5.</b>	<b>Diferencia entre un modelo de supercie y un modelo de volumen</b>	<b>74</b>
<b>4.6.</b>	<b>Mallas de supercie de regiones complejas</b>	<b>75</b>
<b>4.7.</b>	<b>Mallas volumétricas de regiones complejas</b>	<b>76</b>
<b>4.8.</b>	<b>Estructuras heterogéneas modeladas a partir de regiones de la columna</b>	<b>78</b>
<b>4.9.</b>	<b>Modelado de la estructura heterogénea Pelvis</b>	<b>79</b>
<b>4.10.</b>	<b>Estructuras heterogéneas modeladas a partir de regiones de la rodilla</b>	<b>80</b>
<b>4.11.</b>	<b>Modelado de estructuras heterogéneas con regiones anidadas</b>	<b>82</b>
<b>4.12.</b>	<b>Estructuras heterogéneas modeladas a partir de regiones de cerebro</b>	<b>83</b>
<b>4.13.</b>	<b>Estructuras heterogéneas modeladas a partir de regiones del abdomen</b>	<b>84</b>
<b>5.1.</b>	<b>Tiempo para el modelado de las regiones del cerebro</b>	<b>86</b>
<b>5.2.</b>	<b>Tiempo para el modelado de las regiones del abdomen</b>	<b>87</b>
<b>5.3.</b>	<b>Tiempo para el modelado de las regiones de la rodilla</b>	<b>89</b>
<b>5.4.</b>	<b>Tiempo para el modelado de las regiones de la zona torácica</b>	<b>90</b>
<b>5.5.</b>	<b>Tiempo para el modelado de las regiones de la zona lumbar</b>	<b>91</b>
<b>5.6.</b>	<b>Estadísticas de la reconstrucción de las regiones del cerebro</b>	<b>93</b>

5.7.	Estadísticas de la reconstrucción de las regiones del abdomen	94
5.8.	Estadísticas de la reconstrucción de las regiones de la rodilla	95
5.9.	Estadísticas de la reconstrucción de las regiones de la zona torácica	97
5.10.	Estadísticas de la reconstrucción de las regiones de la zona lumbar	97
5.11.	Coeficiente de Dice vs. conservación del volumen	100
5.12.	Coeficiente de Dice de las regiones de la zona torácica	101
5.13.	Distancia de Hausdorff vs. conservación del área y del volumen	104
A.1.	Vecindades	112
A.2.	Gráfico de una función implícita	114
A.3.	Transformada de Radon de 1 y 2	119
A.4.	Teorema del corte central	121
A.5.	Adquisición de las proyecciones tomográficas	123
A.6.	Definición de una región en 2D con un camino 4-conectado	126
A.7.	Generación de un árbol cuádruple (quadtree)	127
A.8.	Generación de un grafo de adyacencia de regiones	128
A.9.	Fusión de regiones en un grafo de adyacencia	128

# Lista de tablas

---

<b>4.1. Vista axial, sagital y coronal de las imágenes TAC</b>	<b>70</b>
<b>5.1. Tiempo para el modelado de las regiones segmentadas del cerebro</b>	<b>85</b>
<b>5.2. Tiempo para el modelado de las regiones segmentadas del abdomen</b>	<b>87</b>
<b>5.3. Tiempo para el modelado de las regiones segmentadas de la rodilla</b>	<b>88</b>
<b>5.4. Tiempo para el modelado de las regiones de la zona torácica</b>	<b>89</b>
<b>5.5. Tiempo para el modelado de las regiones de la zona lumbar</b>	<b>90</b>
<b>5.6. Tiempo para el modelado de las estructuras anatómicas heterogéneas</b>	<b>91</b>
<b>5.7. Datos de los modelos de las regiones del cerebro</b>	<b>92</b>
<b>5.8. Datos de los modelos de las regiones del abdomen</b>	<b>92</b>
<b>5.9. Datos de los modelos de las regiones de la rodilla</b>	<b>95</b>
<b>5.10. Datos de los modelos de las regiones de la zona torácica</b>	<b>96</b>
<b>5.11. Datos de los modelos de las regiones de la zona lumbar</b>	<b>96</b>
<b>5.12. Datos de los modelos de las estructuras anatómicas heterogéneas</b>	<b>98</b>
<b>5.13. Coeciente de Dice de los modelos de las regiones</b>	<b>99</b>
<b>5.14. Coeciente de Dice de las estructuras heterogéneas modeladas</b>	<b>101</b>
<b>5.15. Distancia de Hausdorff de los modelos de las regiones</b>	<b>103</b>
<b>5.16. Distancia de Hausdorff de las estructuras heterogéneas modeladas</b>	<b>104</b>



# Introducción

---

Una estructura anatómica heterogénea es un objeto que consta de una o más partes con un tipo de material asociado y una interfaz que las separa (Wu *et al.*, 2008) (figura 1.1). Los métodos para el estudio de las estructuras heterogéneas han avanzado radicalmente, desde la disección de animales y de cadáveres frescos o preservados, hasta técnicas computacionales mediante la creación de modelos tridimensionales (3D). El objetivo de estos métodos ha sido proporcionar un medio a través del cual se puedan estudiar más fácilmente las propiedades de los órganos y las estructuras heterogéneas.

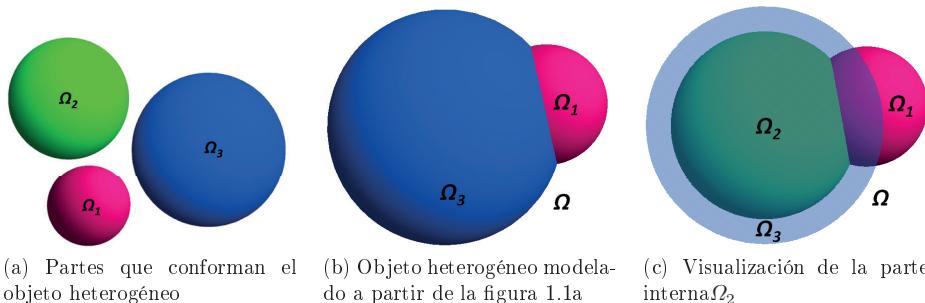


Figura 1.1: Modelado de un objeto heterogéneo artificial  $\Omega$  compuesto de tres partes,  $\Omega_1$ ,  $\Omega_2$  y  $\Omega_3$

Los ejemplos más frecuentes se encuentran en la bioingeniería, la medicina asistida por computador y la simulación mediante elementos finitos, donde los investigadores descomponen las estructuras anatómicas en regiones individuales para luego combinarlas en diferentes configuraciones y comprender mejor su funcionamiento (De *et al.*, 2010).

Una estructura heterogénea  $\Omega$  y, en general, los objetos heterogéneos se pueden describir como la unión de todas las partes  $\Omega_i$  que componen la estructura (Wang y Wang, 2005), es decir:

$$\Omega = \bigcup_{i=1}^n \Omega_i \quad (1.1)$$

El índice  $i$  ( $i = 1, 2, \dots, n$ ) es el identificador *id* de cada parte  $\Omega_i$ .

Se puede observar en la figura 1.1 cómo un objeto heterogéneo es modelado mediante la descomposición espacial del objeto en sus partes constituyentes. La figura 1.1a muestra 3 esferas identificadas como  $\Omega_1$ ,  $\Omega_2$  y  $\Omega_3$ , que forman parte del objeto heterogéneo  $\Omega$  que aparece en la figura 1.1b. La figura 1.1c muestra la parte  $\Omega_3$  de forma translúcida, lo que permite observar la parte  $\Omega_2$  que se encuentra en contacto con la parte  $\Omega_1$ .

Con base en la idea de particiones espaciales, algunos investigadores se centran en el modelado de objetos heterogéneos principalmente para procesos de fabricación con base en *software* de diseño asistido por computador (CAD). El propósito es diseñar y fabricar objetos sólidos con diferentes propiedades de materiales (Schroeder *et al.*, 2005; Wang *et al.*, 2011; Yuan *et al.*, 2012) (figura 1.1c). Sin embargo, la creación de modelos 3D con base en *software CAD* es un proceso completamente manual, lo que hace que el modelado de estructuras anatómicas heterogéneas sea extremadamente difícil, debido principalmente a las geometrías irregulares tanto de las estructuras heterogéneas que se están modelando como de sus partes constituyentes.

Otras técnicas de modelado 3D de objetos heterogéneos, tanto naturales como fabricados por el hombre, incluyen la reconstrucción 3D con base en el escaneado 3D del objeto heterogéneo (Wang, 2014). En este caso se realiza el escaneo con un láser constituido por dos componentes básicos: por un lado, el dispositivo medidor de distancias; por otro, el medidor de barrido, que cuenta con espejos que miden las direcciones en vertical y en horizontal y obtiene así datos de una gran cantidad de puntos en los tres ejes cartesianos, X, Y y Z (figura 1.2).

La figura 1.2a muestra un escáner Go!SCAN 3D de luz blanca, el cual se hace girar alrededor del objeto para obtener una nube de puntos y generar la malla de superficie. Por otro lado, el escáner EinScan-SE de la figura 1.2a tiene una bandeja sobre la cual se posiciona el objeto escaneado. El escáner permanece estático capturando los datos del objeto mientras este gira sobre la bandeja.

La reconstrucción 3D con base en escáneres láser es rápida y precisa; sin embargo, estos dispositivos solo permiten modelar la superficie de los objetos sólidos y no sus partes o

regiones internas. Para modelar estas regiones, los investigadores usan imágenes médicas de tomografía axial computarizada (TAC) y técnicas de ingeniería inversa para crear modelos de objetos heterogéneos (d'Otreppe *et al.*, 2012; Yoo, 2013; Otake y Suzuki, 2013). El concepto de generar modelos 3D a partir de imágenes médicas fue publicado por primera vez por Jerry Fram (Fram y Deutsch, 1975) y desde entonces ha generado un gran interés de investigación (Stytz *et al.*, 1991).



Figura 1.2: Escáneres para capturar los puntos de superficie de un objeto

Fuente: [www.creaform3d.com](http://www.creaform3d.com) y [www.einscan.com](http://www.einscan.com)

Las imágenes TAC permiten analizar la condición interna de un objeto sin el uso de procedimientos invasivos o destructivos (figura 1.3). Para este fin se toman imágenes 2D a través de varios cortes paralelos que son luego apilados en datos volumétricos (figura 1.3a). A partir de esos datos, se pueden segmentar las regiones de interés que servirán de base para la creación de modelos 3D que permitan analizar el objeto (figura 1.3b).

La segmentación obtenida a partir de una imagen TAC es la representación más precisa de una estructura heterogénea; sin embargo, esta descripción de volumen necesita gran espacio en memoria computacional debido a sus tres dimensiones y al gran número de véxeles apilados. Además, las segmentaciones carecen de una representación geométrica, lo que impide utilizarlas directamente en aplicaciones de ingeniería como análisis y simulación. Por lo tanto, se necesita un medio para reconstruir la geometría de la estructura a partir de las segmentaciones.

La creación de bordes mediante mallas poligonales suele ser la forma ideal para representar con precisión los resultados de la reconstrucción geométrica, al proporcionar una buena descripción de la superficie de las estructuras (Braid, 1974). Adicionalmente, las mallas poligonales son la representación gráfica comúnmente aceptada debido a su compatibilidad con el *software* y el *hardware* existentes.

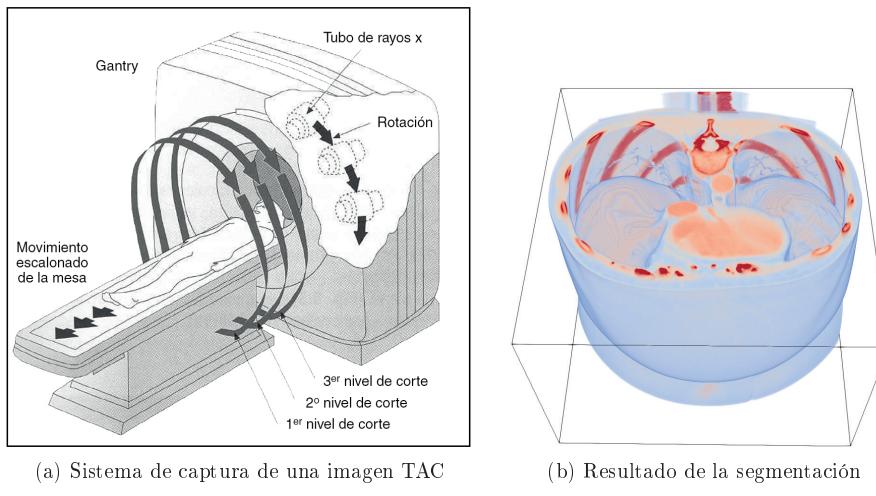


Figura 1.3: Captura y segmentación de una imagen TAC de la región abdominal de una persona

Actualmente hay dos técnicas para generar mallas poligonales a partir de imágenes médicas (Moustakides *et al.*, 2000), que dependen de si la segmentación se hace de forma manual o automática.

La primera técnica construye la superficie a partir de una nube de puntos obtenida de forma manual de una serie de contornos paralelos extraídos secuencialmente de las imágenes TAC (Knopf y Al-Naji, 2001) (figura 1.4). El proceso comienza con el marcado de puntos sobre el contorno de la estructura que aparece en cada corte de la imagen TAC (figura 1.4a). Luego estos puntos se almacenan y organizan en una nube de puntos (figura 1.4b). Finalmente, se conectan los puntos  $i$  y  $j$  de las capas adyacentes  $S$  y  $T$  para construir la triangulación de la superficie de la estructura (figura 1.4c).

La precisión de la malla de superficie depende en gran medida de los puntos de borde extraídos y de la función utilizada para ajustar las curvas a esos contornos. Los resultados de la segmentación manual se utilizan, entre otros aspectos, para evaluar los resultados de los algoritmos automáticos y semiautomáticos (Heimann *et al.*, 2009).

La segunda opción es convertir la segmentación en una malla poligonal mediante un proceso semiautomático o automático utilizando, por ejemplo, el algoritmo Marching Cubes (Lorensen y Cline, 1987). Sin embargo, este proceso no es una tarea fácil y plantea dos problemas relacionados con las inexactitudes que surgen de y son inherentes al proceso de adquisición y segmentación de las imágenes TAC (figura 1.5).

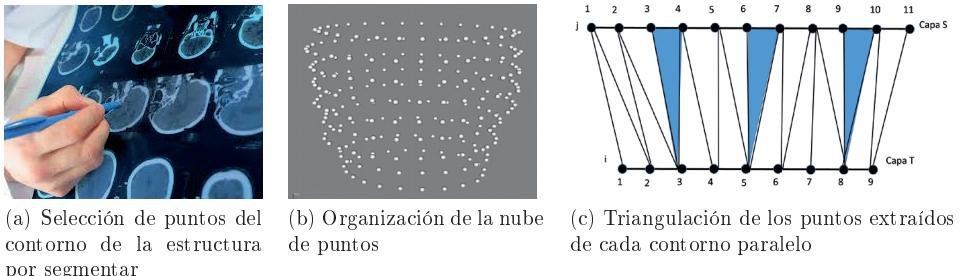


Figura 1.4: Creación manual de una malla de superficie a partir de una imagen TAC

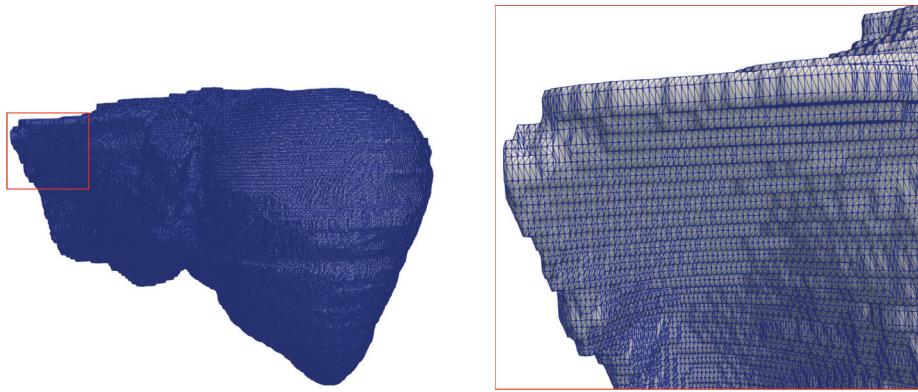
El primer problema está relacionado con los miles o millones de primitivas poligonales (triángulos) necesarias en las mallas para capturar los detalles de alta complejidad geométrica que tienen las estructuras heterogéneas. Esto no es práctico en términos computacionales ya que el tiempo y la memoria requeridos para el despliegue de los modelos 3D son proporcionales al número de triángulos. La figura 1.5a muestra la malla de superficie de un hígado humano con 221.402 vértices y 442.788 triángulos, lo cual impide incluso una buena visualización.

El segundo problema está relacionado con el efecto de escalón que presentan las mallas generadas de las segmentaciones. El proceso de segmentación extrae de la imagen TAC aquellos vóxeles que pertenecen a la región de interés y los apila en bloques. Las mallas siguen los contornos de estos bloques y el resultado final no refleja la suavidad que tienen las estructuras anatómicas. En la figura 1.5b se puede observar el zoom de la figura 1.5a, donde se evidencian de mejor manera los dos problemas en mención.

Como consecuencia, para que los modelos 3D generados a partir de las segmentaciones puedan ser utilizados, se debe reducir el número de triángulos que conforman la malla y garantizar la calidad de los modelos reduciendo el efecto escalón, ya que estos deben ser suaves y ajustarse a los límites de las regiones descritas en la imagen TAC y que conforman la estructura heterogénea.

Para reducir el número de triángulos de una malla se necesita eliminar información redundante (Kobbelt *et al.*, 1998). En general, un vértice es redundante si se puede cambiar su vecindad por otra con un número menor de triángulos. Así mismo, para reducir el efecto de escalón y lograr modelos con superficies suaves, se puede aplicar un procedimiento de suavizado como el laplaciano (Field, 1988). No obstante, estas técnicas pueden reducir considerablemente el área de la superficie (Taubin, 1995) y el volumen de la estructura anatómica ((Zhou *et al.*, 2005); (Zhang *et al.*, 2009)). La preservación

del volumen es una característica importante de los modelos 3D requerida en diferentes campos, desde la computación gráfica para hacer que las deformaciones parezcan más naturales (Roussellet, 2018), hasta sistemas dinámicos para lograr simulaciones más estables (Ozbolat y Koc, 2011).



(a) Malla de superficie generada con el algoritmo Marching Cubes (Lorensen y Cline, 1987)

(b) Zoom de la región demarcada en la figura 1.5a por el recuadro rojo

Figura 1.5: Generación automática de la malla de superficie a partir de la segmentación de un hígado humano

Este libro está dividido en cinco capítulos, además de la Introducción, y un anexo. En el capítulo 2 se describen las imágenes, métodos y herramientas utilizados en el modelado 3D de estas estructuras. El capítulo 3 se centra en el diseño de un método de modelado a partir de imágenes TAC; este método, denominado RAM (Region Aware Modeling), se estructura con base en un enfoque de *dividir y conquistar*. En el capítulo 4 se presentan diferentes experimentos que permiten comprobar la versatilidad del método RAM en el modelado de una gran variedad de geometrías que tienen generalmente tanto las regiones como las estructuras anatómicas heterogéneas. En el capítulo 5 se analizan los resultados de aplicar el método RAM. En el capítulo 6 se plasman las conclusiones más significativas del trabajo realizado. Finalmente, en el anexo A se revisan múltiples conceptos que se consideran relevantes para el modelado de las estructuras anatómicas heterogéneas con base en imágenes médicas.

# Imágenes, métodos y herramientas

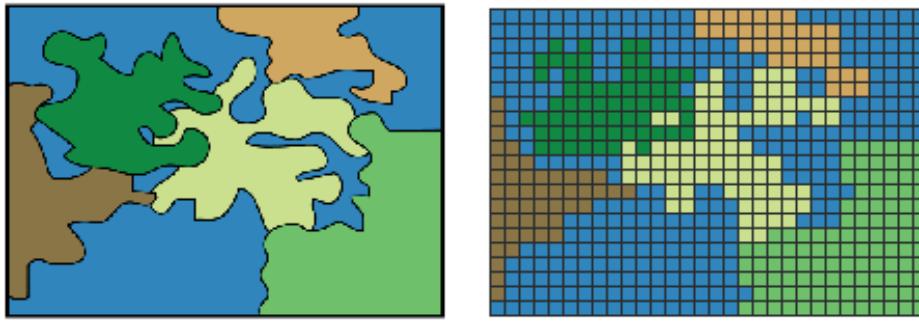
En este capítulo se hace un estudio de las imágenes, los métodos y las herramientas que se pueden utilizar en el modelado 3D de estructuras anatómicas heterogéneas. Este estudio comienza con las imágenes médicas utilizadas en la adquisición de datos volumétricos (sección 2.1). De estas imágenes se pueden segmentar las diferentes partes que componen las estructuras heterogéneas. Luego se analizan los métodos y herramientas para realizar la reconstrucción de las superficies a partir de los datos segmentados (sección 2.2). Finalmente, se estudian los algoritmos y métodos para extraer la geometría tanto de las regiones (sección 2.3) como de las estructuras anatómicas heterogéneas (sección 2.4).

## 2.1. Imágenes médicas

Las imágenes médicas pueden estar en dos o tres dimensiones. Las imágenes en dos dimensiones se dividen en *ráster* y *vectorial* (figura 2.1). Una imagen ráster es una estructura o archivo de datos que representa una rejilla o grilla rectangular de dos dimensiones, donde se almacena el valor de cada punto o píxel  $p_{x,y}$  de la imagen (figura 2.1b). Por otro lado, las imágenes vectoriales son descritas mediante el almacenamiento de descriptores de las formas, que generalmente son regiones o áreas de color limitadas por líneas y curvas, pero que no tienen ninguna referencia a alguna grilla de píxeles en particular (figura 2.1a). Una imagen ráster se puede abstraer como una función:

$$I(x, y) : R \rightarrow V \quad (2.1)$$

Donde  $R \subset \mathbb{R}^2$  es un área rectangular y  $V$  es el conjunto de posibles valores del píxel  $p_{x,y}$ . Este valor es un promedio local de las características medibles del objeto o escena y es llamado *punto de muestra de la imagen*. Así, cuando se encuentra el valor *val* en un píxel  $p_{x,y}$ , significa que el valor de la imagen en la vecindad del punto de muestra  $p_{x,y}$  es *val*. Las imágenes médicas de radiografía son un claro ejemplo de imágenes ráster.



(a) Imagen vectorial compuesta por un conjunto de polígonos

(b) Resultado ráster del proceso de conversión de la imagen de la figura 2.1a

Figura 2.1: Imágenes de dos dimensiones

Una de las formas de representar un objeto sólido es almacenando el volumen ocupado por él mediante un conjunto de muestras llamadas *vóxeles* (Kaufman *et al.*, 1993). De acuerdo con Kou y Tan (2007), la representación basada en vóxeles se puede generalizar como:

$$O = \{v_i\} = \{(x_i, y_i, z_i, m_i), \quad 1 \leq i \leq n\} \quad (2.2)$$

Donde  $v_i$  es un elemento vóxel en la colección de  $n$  vóxeles que constituyen el objeto sólido  $O$ ,  $(x_i, y_i, z_i)$  representa la ubicación geométrica del vóxel  $v_i$  y  $m_i$  representa la composición del material dentro del volumen.

Las imágenes médicas como la tomografía axial computarizada (TAC) se utilizan ampliamente para la visualización y el diagnóstico médico, pero también son esenciales para la construcción de modelos matemáticos y computacionales de objetos volumétricos con fines de análisis y simulación. Sin embargo, los datos digitales que contiene la TAC no tienen descripciones estructurales más allá de los valores de intensidad (subsección 2.1.1), lo que obliga a separarlos en regiones homogéneas  $\Omega_i$  semánticamente significativas mediante un proceso conocido como *segmentación* (subsección 2.1.2).

### 2.1.1. Imágenes TAC

La tomografía axial computarizada fue creada por Godfrey Hounsfield en 1972, en conjunto con Allan Cormack, y es esencialmente una extrapolación del concepto de radiografía en dos dimensiones (figura 2.2). En lugar de tomar una radiografía desde una sola perspectiva, una TAC hace girar una fuente puntual de rayos X alrededor de un objeto para ser fotografiado (figura 1.3a). Esto expone el detector en el lado opuesto del objeto. Los datos obtenidos al irradiar un objeto por medio de rayos X desde distintas direcciones suelen llamarse *proyecciones del objeto* (Epstein, 2007; Faridani, 2003).

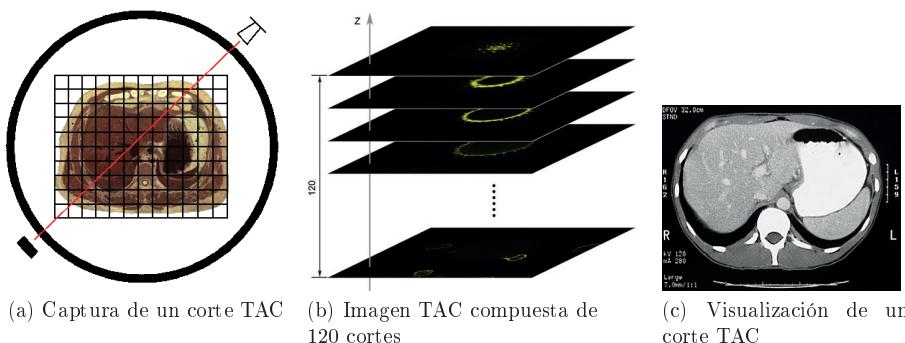


Figura 2.2: Proceso para la captura de una imagen TAC

Tomando los cálculos de una rotación completa, es posible reconstruir una imagen ráster del objeto (figura 2.2a), que será luego apilada en una cuadrícula rectangular 3D (figura 2.2b). El valor *val* del píxel  $p_i$  en una imagen TAC se basa en la absorción de rayos X del tejido en ese punto. De igual forma, los valores de intensidad, que son continuos, se tienen que dividir en una cantidad de niveles discretos  $V$ , generalmente 256. La imagen resultante puede visualizarse en un monitor o imprimirse en una película de rayos X (figura 2.2c).

La salida del escáner TAC es una cuadrícula 3D cuyos valores  $V$  son las densidades de las diferentes partes del objeto. El escáner TAC convierte la densidad del objeto  $\mathbb{R}^3 \rightarrow \mathbb{R}$  en un arreglo 3D de números reales  $\mathbb{Z}^3 \rightarrow \mathbb{R}$ . Así, una imagen TAC contienen información muestrada en 3D sobre una región específica de interés, de un objeto o persona, expresada mediante una función trivariada como:

$$f : D \subset \mathbb{R}^3 \rightarrow \mathbb{R} \\ (x_i, y_j, z_k) \rightarrow f(x_i, y_j, z_k), i, j, k = 1, \dots, n \quad (2.3)$$

El significado de cada punto o vóxel  $v_i$  en el volumen representa los coeficientes lineales de atenuación de los rayos X del material que compone el objeto en ese punto (John y McCloy, 2004). Los valores de los vóxeles se escalan de tal manera que el coeficiente de atenuación lineal de los rayos X del aire sea igual a  $-1024$  y el del agua sea igual a cero. Las unidades Hounsfield (HU) son coeficientes de atenuación de un material, normalizados al coeficiente de atenuación del agua, las cuales se definen por:

$$HU_{material} = \frac{\mu_{material} - \mu_{agua}}{\mu_{agua}} \times 1000 \quad (2.4)$$

El rango de brillo de un píxel depende de los valores de absorción asignados a los diferentes materiales. Por convención, el agua tiene un valor de 0 HU. Las estructuras más densas tienen un número más alto y el aire tiene el número más bajo, que se fija en  $-1000$  HU. Por lo tanto, el hueso es blanco brillante y el aire es negro. Esta atenuación puede ser alterada para hacer las estructuras visualizadas más oscuras o más brillantes, dependiendo de la vista deseada.

Los puntos  $p_{x,y,z}$  de los datos en la cuadrícula 3D actúan como los datos de parametrización 3D, donde la geometría está definida por los componentes cartesianos  $x$ ,  $y$ ,  $z$ , y la composición del material por valores de HU, es decir:

$$p_{x_i,y_j,z_k} = \{x_{i,j,k}, y_{i,j,k}, z_{i,j,k}, m_{i,j,k}\}^T \quad (2.5)$$

Donde  $x_{i,j,k} = \Delta x \times (i - 1)$ ,  $y_{i,j,k} = \Delta y \times (j - 1)$ ,  $z_{i,j,k} = \Delta z \times (k - 1)$  y  $(1 \leq i \leq nx, 1 \leq j \leq ny, 1 \leq k \leq nz)$ . Adicionalmente,  $m_{i,j,k}$  denota el valor de la HU en el pixel  $i^{th}$ ,  $j^{th}$  en el corte TAC  $k^{th}$  y  $\Delta x, \Delta y, \Delta z$  denotan las longitudes incrementales en las direcciones de las coordenadas  $x$ ,  $y$ , y  $z$ , respectivamente.

### 2.1.2. Segmentación de imágenes TAC

Para construir con éxito modelos 3D de estructuras heterogéneas a partir de imágenes TAC, es crucial identificar y segmentar las diferentes regiones que componen estas estructuras (Prince y Links, 2006). La segmentación es esencialmente un problema de clasificación donde se asigna a cada vóxel de la imagen médica una etiqueta que representa la región o material a la que este pertenece (Hu, 2009).

Dada una imagen  $I$ , se puede definir la segmentación como el proceso que divide a  $I$  en  $n$  regiones  $\Omega_1, \Omega_2, \dots, \Omega_n$ , que cumplen con las siguientes condiciones, de acuerdo con las propiedades de vecindad (subsección A.1.4) y conectividad (subsección A.1.5):

- Cada vóxel debe pertenecer a la imagen, es decir,  $\bigcup_{i=1}^n v_i = I$ .
- Los vóxeles, en una determinada región, deben de estar conectados en algún sentido predefinido.
- Las regiones deben de ser disjuntas  $\Omega_i \cap \Omega_j = \emptyset$  para todo  $i$  y  $j$  tal que  $i \neq j$ .
- La similitud de intensidad  $S$  de los puntos  $p_{x,y,z}$  de la región  $\Omega_i$  es verdadero  $S(\Omega_i) = \text{verdadero}$  para  $i = 1, 2, \dots, n$ .
- $S(\Omega_i \cup \Omega_j) = \text{falso}$  para cualquier región adyacente  $\Omega_i$  y  $\Omega_j$ .

Hoy en día existe una gran variedad de métodos de segmentación; por ello, la selección del método depende, entre otros factores, de la aplicación a desarrollar, del tipo de imagen y de sus características (Bankman, 2008; van Rikxoort, 2009). Los algoritmos de segmentación pueden clasificarse en dos grupos (González y Woods, 2002; Withey y Koles, 2007): algoritmos dependientes del dominio o de *reconocimiento*, que intentan reconocer regiones específicas en una escena, por ejemplo, núcleos anormales en citologías cervicales (Zhang *et al.*, 2017), y algoritmos independientes del dominio o de *delineación*, que producen una partición de la imagen original en un conjunto de regiones  $\Omega_i$  según un criterio de homogeneidad. En ese caso, el reconocimiento de las regiones se realiza en etapas posteriores.

A continuación se hace una descripción de dos métodos de *delineación* que gozan de popularidad en el campo de las imágenes médicas: umbralización y crecimiento de regiones. En estos métodos la salida del algoritmo representa la región ocupada por la región de interés (Lee, 1986; Ohlander *et al.*, 1978).

### 2.1.2.1. Umbralización

En este método se escoge un intervalo umbral  $[t_1, t_2]$  para la imagen médica de entrada, de modo que la región pueda ser especificada completamente por los vóxeles cuyas intensidades recaigan en este intervalo (figura 2.3). Para establecer este umbral, es necesario analizar la región de interés que se desea aislar, con el propósito de encontrar una HU o nivel de gris característico y exclusivo que la defina.

La figura 2.3a muestra una imagen de entrada en 2D, donde aparecen en color rojo los valores que están en el rango  $[t_1, t_2]$ . El resultado de este método es una imagen binaria  $I_0 = (I_0, g_0)$  en la que los vóxeles con intensidad 1 representan la región ocupada por la

región (figura 2.3b). La imagen binaria de salida se define de la siguiente forma, para cualquier  $v \in I_0$ :

$$g_0(v) = \begin{cases} 1 & \text{si } g_{ij}(v) \in [t_1, t_2] \quad \forall 1 \leq i, j \leq n \\ 0 & \text{en otro caso} \end{cases} \quad (2.6)$$

Donde  $g_{ij}(v)$  es el nivel de gris en el voxel  $v$ .

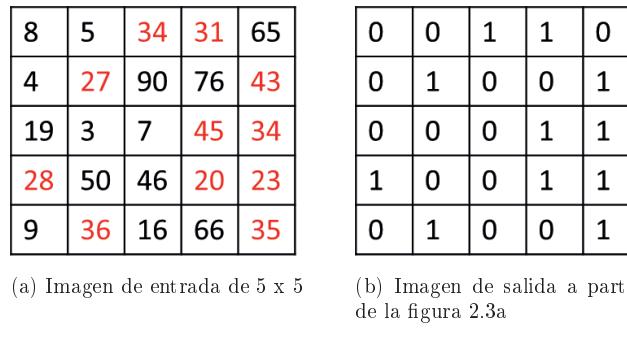


Figura 2.3: Aplicación del método de umbralización en 2D con  $t_1 = 20$  y  $t_2 = 45$

La umbralización es quizás el método de segmentación más utilizado por su simplicidad y eficiencia. No obstante, se aclara que los resultados de una umbralización rara vez son perfectos, aunque sí pueden resultar muy útiles como una primera aproximación.

### 2.1.2.2. Crecimiento de regiones

El crecimiento de regiones es un método utilizado en imágenes médicas bajo la hipótesis de que los píxeles pertenecientes a un mismo tejido u órgano presentan características similares con un significado semántico. Al igual que el de umbralización, este es un método basado en similitud, es decir, tiene como objetivo determinar los píxeles que cumplen con cierto criterio de semejanza para agruparlos en regiones (Dougherty, 2009).

El crecimiento de regiones es un proceso iterativo que inicia con puntos semilla (figura 2.4a), los cuales representan las características que definen a la región y que deben tener los píxeles candidatos para poder ser agregados allí. Este procedimiento se lleva a cabo mediante la comparación de los píxeles vecinos con los puntos semilla, o los puntos pertenecientes a la región. Por lo tanto, la selección de los puntos semilla es un paso

significativo en este método, ya que los resultados dependerán directamente del criterio de semejanza que se establezca.

Una vez seleccionados los puntos semilla, la región crecerá únicamente hacia los píxeles vecinos que cumplan con el criterio establecido con anterioridad, con lo cual se garantiza conectividad entre los píxeles que constituyen una región (figura 2.4b). En la imagen médica pueden existir puntos que cumplen con el criterio de similitud, pero si estos no son adyacentes a los puntos semilla o a algún píxel perteneciente a la región, no podrán adicionarse. Esto representa una protección contra el ruido, aunque puede producir regiones con espacios vacíos (huecos).

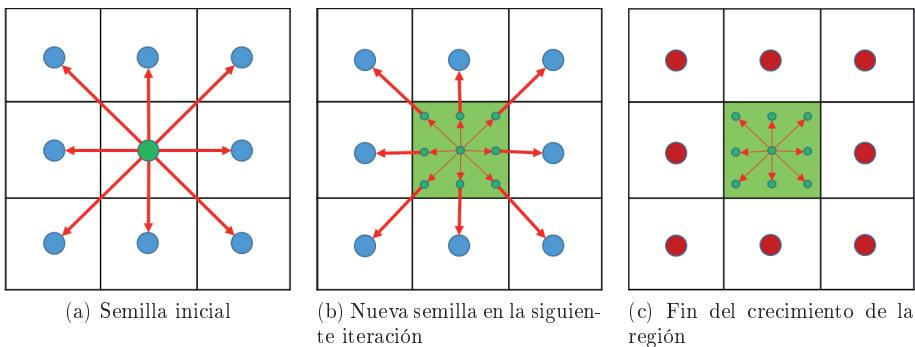


Figura 2.4: Aplicación del método de crecimiento de regiones

En general, este método puede resumirse en los siguientes pasos:

1. Se establece un conjunto de criterios para la inclusión de vóxeles en la estructura. Algunos ejemplos son:
  - a) La intensidad de un punto (vóxel) ha de estar en el intervalo  $[t_1, t_2]$ .
  - b) La intensidad media de los vóxeles incluidos en la región de crecimiento en cualquier momento durante el proceso debe estar en el intervalo  $[t_3, t_4]$ .
  - c) La varianza de la intensidad de los vóxeles incluidos en la región de crecimiento en cualquier instante durante el proceso debe estar en el intervalo  $[t_5, t_6]$ .
2. El usuario especifica una o más semillas en el interior de la región que se quiere segmentar, usando el puntero del ratón. Estas semillas se colocan en una cola y son especialmente señaladas.

3. Si la cola está vacía, se acaba el algoritmo. Si no, se borra el punto  $p$  de la cola, y se saca  $p$ .
4. Se examinan los vecinos, usualmente los 6, 14 o 26 más cercanos de  $p$  para la inclusión (figura 2.5). Aquellos que satisfagan el criterio y no hayan sido previamente marcados y encolados, son añadidos para volver al paso 3.

La figura 2.5 muestra el vecindario de un punto  $p$  marcado de color rojo. Los vecinos 6-conectados aparecen en color azul, a estos se agregan los puntos en color verde para obtener los vecinos 14-conectados y, finalmente, se agregan los puntos en color gris para obtener los vecinos 26-conectados.

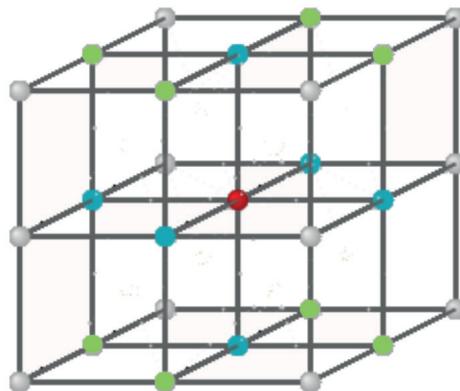


Figura 2.5: El vecindario en 3D de un punto  $p$  (rojo)

Si solo se usa el criterio a), y si  $t_1$  y  $t_2$  permanecen fijos durante el proceso de crecimiento, entonces el método anterior obtiene el mayor número de puntos (vóxeles) conectados a la semilla inicial, satisfaciendo el intervalo dado. Para cualquier combinación de criterios de a) hasta c), y si los valores de  $t_1, t_2, \dots, t_6$  no son fijos, no se puede garantizar que el conjunto de vóxeles  $O(v_1)$  obtenido con la semilla  $v_1$  sea el mismo que el conjunto  $O(v_2)$  resultante a partir de la semilla  $v_2 \neq v_1$ . Esta falta de robustez es un problema en la mayoría de algoritmos de crecimiento de regiones.

## 2.2. Reconstrucción geométrica de superficies

Cada región segmentada de la imagen TAC carece de una representación geométrica; por lo tanto, en esta sección se estudiarán los diferentes métodos y algoritmos para realizar

la reconstrucción geométrica de la superficie de las diferentes regiones que conforman la estructura heterogénea. En la subsección 2.2.1 se analizan los métodos paramétricos como las B-spline y las NURBS. Los métodos de superficies simpliciales, como el diagrama de Voronoi y la triangulación de Delaunay, se estudian en la subsección 2.2.2. En la subsección 2.2.3 se analizan algunos algoritmos para el modelado de superficies implícitas con base en funciones de base radial, partición de la unidad implícita y funciones indicatriz. Luego se hará una descripción matemática del ajuste de funciones implícitas por el método de mínimos cuadrados (subsección 2.2.4) y de la estimación de la normal mediante el análisis de componentes principales (subsección 2.2.5).

### 2.2.1. Métodos paramétricos

Los métodos paramétricos como B-spline y NURBS fueron las primeras técnicas utilizadas para resolver los problemas de reconstrucción de superficie mediante el ajuste con parches superficiales locales. En el contexto de las representaciones paramétricas, el problema de la reconstrucción de la superficie implica el cálculo de una superficie  $S$  definida por una función  $F(u, v)$ , que se aproxima, tanto como sea posible, a cada punto de la nube de puntos dada en  $\mathbb{R}^3$ , donde  $F$  pertenece a un espacio lineal específico de funciones. Ejemplos de tales superficies paramétricas son las superficies de Bézier, las B-spline y las NURBS (Farin y Farin, 2002). Debido a que las curvas de Bézier no se podían modificar localmente y el movimiento de los puntos de control afectaba la forma de la curva completa, se desarrolló el método de B-spline, que permite alcanzar una continuidad C2. Para la superficie B-spline en el parámetro  $(u, v)$  el punto de control se puede definir como:

$$P(u, v) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} N_{i,k}(u) M_{j,l}(v) \quad (2.7)$$

Donde  $B_{i,j}$  son los vértices de una malla de control poligonal, con  $i$  en el rango entre 0 y  $n$ , y  $j$  en el rango entre 0 y  $m$ .  $N_{i,k}(u)$  y  $M_{j,l}(v)$  son las funciones B-spline de base, dadas por las ecuaciones:

$$N_{i,k}(u) = \frac{u - u_i}{u_{i+k-1} - u_i} N_{i,k-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_{i+1}} N_{i+1,k-1}(u) \quad (2.8)$$

$$M_{j,l}(v) = \frac{v - v_j}{v_{j+l-1} - v_j} M_{j,l-1}(v) + \frac{v_{j+l} - v}{v_{j+l} - v_{j+1}} M_{j+1,l-1}(v) \quad (2.9)$$

Chen *et al.* (2010) usaron B-spline para construir varios modelos del interior de la pared del ventrículo izquierdo (LV). Los modelos se calcularon mediante B-spline integrales y las curvas se ajustaron en el ciclo cardíaco. Otro trabajo relacionado es el de Liu *et al.* (2005), cuyo algoritmo se aplica en el sentido de mínimos cuadrados (subsección 2.2.4), generando puntos de control adecuados a la curva de ajuste B-spline.

Las superficies NURBS (Non-Uniform Rational B-Spline) es una generalización de las superficies de Bézier y las B-splines que se desarrollaron porque los métodos Bézier y B-splines contienen algunas limitaciones al no poder representar con precisión las curvas cónicas y no modelar formas más complicadas que requieren curvas de Bézier de orden superior. Una superficie NURBS con parámetros  $(u, v)$  se define como:

$$P(u, v) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} S_{i,k}(u, v) \quad (2.10)$$

Donde  $B_{i,j}$  son los puntos 3D de la malla de control y  $S_{i,j}(u, v)$  son las funciones de superficie B-spline racionales, definidas como:

$$S_{i,j}(u, v) = \frac{h_{i,j} N_{i,k}(u) M_{j,l}(v)}{\sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N_{i,k}(u) M_{j,l}(v)} \quad (2.11)$$

Donde  $h_{i,j}$  es el peso,  $N_{i,k}(u)$  y  $M_{j,l}(v)$  son las funciones B-spline no-racionales de base, dadas por las ecuaciones anteriormente definidas.

Actualmente, los modelos NURBS son estándares industriales para la representación de superficies usados ampliamente en el campo de la ingeniería inversa (Tsai *et al.*, 2009; He y Qin, 2004). Además, las NURBS pueden usarse como un método para la aproximación o interpolación de datos dispersos, también incorporado a la mayoría de los sistemas de modelado geométrico actuales (Hoffmann, 2005; Rogers, 2000). Al usar NURBS, la superficie reconstruida es más suave y puede tratar con un conjunto de datos no uniforme (Zhao *et al.*, 2001).

### 2.2.2. Métodos de geometría computacional

Estos métodos, también conocidos como *métodos de superficies triangulares* o *métodos simpliciales*, dependen de algoritmos como la triangulación de Delaunay y los diagramas de Voronoi. Los métodos de geometría computacional interpolan los puntos dados mediante una nube de puntos, lo que los hace sensibles a la presencia de ruido. El

algoritmo Alpha Shapes (Edelsbrunner y Mücke, 1994) y el algoritmo Crust (Amenta *et al.*, 2000) son los dos ejemplos más exitosos de esta clasificación.

En el algoritmo Alpha Shapes la forma de la superficie se talla eliminando los simplices de la triangulación de Delaunay del conjunto de puntos. Un simplex se elimina si su esfera circunscrita es más grande que la esfera Alpha. El algoritmo Crust calcula el diagrama de Voronoi de los puntos en la nube de puntos. Luego, la triangulación de Delaunay se calcula utilizando los diagramas de Voronoi sobre la nube de puntos. La malla 3D se reconstruye obteniendo todos los triángulos que conectan tres puntos de la triangulación de Delaunay. Esta malla distingue triángulos que son parte de la superficie del objeto de aquellos que están en el interior, porque estos últimos tienen un vértice Voronoi. La complejidad de tiempo de este algoritmo es  $O(n \log n)$ .

Una mejora del algoritmo Crust es el algoritmo Power Crust (Amenta *et al.*, 2001), cuya idea principal es representar la superficie como un polígono usando el diagrama de Voronoi. Además, es robusto ya que se expresa en términos de celdas de la superficie de un objeto sólido. No se requiere ningún mecanismo de llenado de agujeros en este algoritmo de reconstrucción de superficie, pues captura la geometría real de esta. Sin embargo, los datos ruidosos provocan la dispersión de puntos lejos de la superficie.

El algoritmo Tight Cocone propuesto por Dey y Giesen (2003) es una mejora y extensión del diagrama de Voronoi. Aunque este es más rápido que los algoritmos Crust y Power Crust, al igual que los demás algoritmos de esta categoría, no es lo suficientemente robusto para manejar nubes de puntos ruidosos y de baja densidad.

### 2.2.3. Métodos implícitos

Una superficie implícita  $S$  en  $\mathbb{R}^3$  se puede definir como el conjunto de nivel cero de alguna función  $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ , tal que:

$$S = \{\mathbf{x} \in \mathbb{R}^3 : f(\mathbf{x}) = 0\} \quad (2.12)$$

Hay diferentes maneras de definir  $f(\mathbf{x})$ . Por un lado, puede tener una forma analítica cerrada; por ejemplo, el conjunto de niveles cero de la función  $f(x, y, z) = x^2 + y^2 + z^2 - 1^2 = 0$  define una esfera de unidad, clasificando el espacio en dos semiespacios:  $f(\mathbf{x}) > 0$  y  $f(\mathbf{x}) < 0$ . Por otro lado, una función implícita  $f(\mathbf{x})$  también se puede definir a través de métodos basados en datos que toman la forma de un campo de distancia con signo (Hoppe *et al.*, 1992), funciones de base radial (Carr *et al.*, 2001) o una función de indicatriz (Kazhdan *et al.*, 2006). Estos métodos se dividen en:

- *Métodos globales*: apuntan a construir una función única de manera que su conjunto cero interpola o aproxima la nube de puntos globalmente.
- *Métodos locales*: en este caso, la función global resulta de la combinación de funciones de forma locales, cada una de las cuales interpola o se aproxima a una subnube de puntos.

En las funciones de base radial el resultado es la rotación invariante alrededor de un cierto punto  $\mathbf{x}_i$ . Se pueden usar para interpolar una función con  $n$  puntos usando  $n$  funciones de base radial, centradas en estos puntos:

$$F(\mathbf{x}) = \sum_{i=1}^n w_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) \quad (2.13)$$

Donde el interpolador  $F(\mathbf{x})$  es el resultado de la suma de  $n$  funciones de base radial, cada una de las cuales está asociada con un centro distinto  $\mathbf{x}_i$ , y ponderado por un coeficiente adecuado (peso)  $w_i$ .

En esencia, hay dos pasos principales en un procedimiento para reconstruir una superficie muestrada dispersa con funciones de base radial (Carr *et al.*, 2001):

- Construcción de una función de distancia con signo.
- Ajuste de una función de base radial a la función de distancia resultante.

Los métodos implícitos sustentados en funciones de base radial (RBF) se utilizan ampliamente porque pueden reconstruir superficies de cualquier topología con casi cualquier distribución de puntos (Skala, 2013). Las RBF se conocen desde hace mucho tiempo como uno de los métodos más precisos y estables para resolver el problema de la interpolación de datos dispersos. Aunque este método proporciona la superficie de curvatura mínima que pasa a través de los puntos dados, es una técnica global y, en consecuencia, se perderán las características locales.

Ohtake *et al.* (2005) presentan un método llamado *partición de la unidad multinivel* (MPU) para la reconstrucción de superficies implícitas utilizando una técnica local. La idea básica de este método es dividir el dominio de datos de entrada  $\Omega \in \mathbb{R}^d$  en varios subdominios  $\Omega_i$  superpuestos que cubren  $\Omega$ , es decir,  $\Omega \subseteq \bigcup_{i=1}^n \Omega_i$ . Cada subdominio se approxima de manera independiente de los otros usando RBF de soporte compacto. Estos subdominios se grafican en 1D en la figura 2.6 con los colores verde, café, gris y azul, respectivamente.

La figura 2.6 muestra cuatro funciones locales  $f_i (i = 1, 2, 3, 4)$  combinadas con funciones de peso  $w_i$  centradas en los puntos  $\mathbf{x}_i$ . La función resultante  $F$  es la curva de contorno rojo.

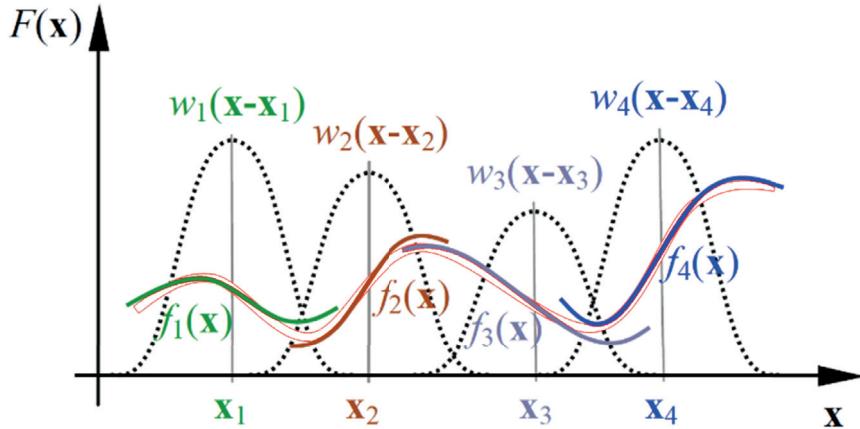


Figura 2.6: Partición de la unidad multinivel en 1D

La función de ajuste global  $F$  es entonces el resultado de una combinación de las funciones locales  $f_i$  ponderadas por las funciones de soporte compacto  $\phi_i$  de la siguiente manera:

$$F(\mathbf{x}) = \sum_{i=1}^n \phi_i(\mathbf{x}) f_i(\mathbf{x}) \quad (2.14)$$

Por ejemplo, en la figura 2.6, la función resultante  $F$  en contorno rojo se construye a partir de una combinación de cuatro funciones locales  $f_i$ , que están asociadas a cuatro funciones de ponderación  $\phi_i$ . La condición  $\sum_i \phi_i = 1$  se obtiene simplemente de las funciones de peso  $w_i$  usando la siguiente fórmula de normalización:

$$\phi_i(\mathbf{x}) = \frac{w_i(\mathbf{x})}{\sum_j w_j(\mathbf{x})}, \quad (j \neq i)$$

Esta fórmula toma el peso  $w_i$  de la actual función de soporte compacto  $\phi_i$  y lo divide por la sumatoria de los demás pesos  $w_j$ . La función de ajuste global  $F$  se puede reescribir entonces como:

$$F(\mathbf{x}) = \frac{\sum_i w_i(\mathbf{x}) f_i(\mathbf{x})}{\sum_j w_j(\mathbf{x})} \quad (2.15)$$

El último método que se analizará para la reconstrucción implícita de superficies forma una ecuación de Poisson para ajustar la superficie lo mejor posible a la nube de puntos dada  $P$ . Este enfoque, llamado *reconstrucción de superficies de Poisson* (Kazhdan *et al.*, 2006), requiere como entrada una nube densa de puntos  $P$  junto con las normales orientadas (figura 2.7).

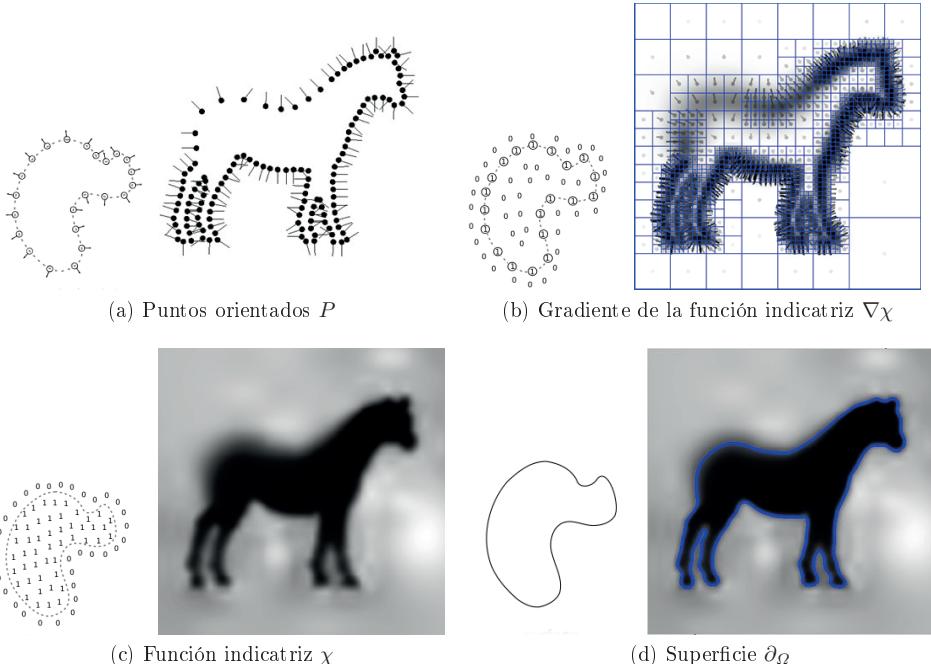


Figura 2.7: Método de reconstrucción de superficies de Poisson  
Adaptado de Kazhdan *et al.* (2006)

Con los datos de  $P$  se define una función indicatriz  $\chi$  cuyo valor es uno dentro de la región reconstruida  $\Omega$  y cero fuera de ella. El gradiente de la función indicatriz se equipara a un campo vectorial, construido a partir de los vectores normales de la nube de puntos. Luego se forma la ecuación de Poisson, la cual se resuelve para obtener la función indicatriz.

La idea clave de la reconstrucción de Poisson, ilustrada en dos dimensiones en la figura 2.7, es que existe una relación integral entre los puntos orientados  $P$  muestreados en la superficie de un objeto  $\Omega$  (figura 2.7a) y la función indicatriz  $\chi$  de dicho objeto.

En específico, el gradiente de la función indicatriz  $\nabla\chi$  (figura 2.7b) es un campo vectorial

$W$  que es cero casi en todas partes (ya que la función indicatriz es constante en casi todas partes), excepto en puntos cercanos a la superficie, donde apunta hacia adentro en la dirección de la normal a la superficie:

$$\nabla \chi = W \quad (2.16)$$

Por lo tanto, los puntos orientados se pueden ver como muestras del gradiente de la función indicatriz del modelo. Kazhdan *et al.* (2006) demostraron que la isosuperficie inducida por la función indicatriz (figura 2.7c) define la superficie reconstruida  $\partial_\Omega$  (figura 2.7d).

El cálculo explícito del campo gradiente  $\nabla \chi$ , en la ecuación 2.16, da como resultado un campo vectorial  $W$  con valores no acotados en el límite de la superficie debido a que la función indicatriz  $\chi$  es una función constante por partes. Para evitar esto, se hace una convolución entre la función indicatriz y un filtro de suavizado  $\psi$ , para así considerar el campo gradiente de la función suavizada:

$$\begin{aligned} \nabla(\chi * \psi) &= (W * \psi) \\ \nabla \tilde{\chi} &= V \end{aligned} \quad (2.17)$$

Una buena elección de filtro  $\psi$  es un filtro gaussiano cuya variación está en el orden de la resolución del muestreo.

Una vez formado el campo vectorial  $V$ , se quiere resolver la función  $\chi$  tal que  $\nabla \tilde{\chi} = V$ . Sin embargo,  $V$  generalmente no es integrable, por lo que casi nunca existe una solución exacta. Se decide entonces encontrar una solución en términos de mínimos cuadrados (subsección 2.2.4), es decir,  $\min_\chi \| \nabla \chi - \vec{V} \|$ , aplicando el operador de divergencia  $\nabla \cdot$  a ambos lados de la ecuación 2.17:

$$\nabla \cdot \nabla \tilde{\chi} = \nabla \cdot V \quad (2.18)$$

Como la divergencia del gradiente es el laplaciano, el problema se transforma en una ecuación de Poisson donde se quiere descubrir la función  $\chi$ , cuyo laplaciano es igual al gradiente del campo vectorial  $V$ :

$$\Delta \tilde{\chi} = \nabla \cdot V \quad (2.19)$$

Para resolver la ecuación de Poisson, la función indicatriz debe representarse en el espacio 3D. Primero, se debe elegir el espacio de funciones sobre el cual discretizar

el problema. El enfoque más directo es comenzar con una cuadrícula 3D regular. Sin embargo, una estructura tan uniforme resulta poco práctica para la reconstrucción detallada, ya que la resolución del espacio es cúbica, mientras que el número de triángulos de superficie crece de forma cuadrática. Afortunadamente, solo es necesaria una representación precisa de la función implícita cerca de la superficie reconstruida. Esto motiva el uso de un árbol octree adaptativo (subsección A.4.2) tanto para representar la función implícita como para resolver el sistema de Poisson (Kazhdan *et al.*, 2006), donde cada una de las hojas almacena los valores de  $\chi$  en diferentes puntos a través de la superficie reconstruida.

Finalmente, se extrae la superficie  $\partial\Omega$  de la función  $\chi$  utilizando el algoritmo de Marching Cubes (subsección 2.3.1). Para esto es necesario seleccionar un isovalor y luego extraer la isosuperficie correspondiente de la función indicatriz.

El algoritmo de Poisson fue ampliado en el trabajo de Kazhdan y Hoppe (2013) mediante la incorporación de pesos a los puntos asignados para la interpolación. La extensión se puede interpretar como una generalización del marco matemático subyacente a una ecuación de Poisson proyectada.

#### 2.2.4. Aproximación por mínimos cuadrados

El término *mínimos cuadrados* describe una técnica popular para resolver sistemas de ecuaciones específicos sobre determinados o inexactos. En lugar de resolver las ecuaciones exactamente, se calcula una minimalización de la suma de los cuadrados de los residuos. Esta subsección se enfocará en las aproximaciones de mínimos cuadrados para el ajuste de superficies en  $\mathbb{R}^3$ .

Dado un conjunto de puntos  $P$  de  $n$  puntos y un punto de referencia, se quiere obtener una función  $f : \mathbb{R}^3 \rightarrow \mathbb{R}$  que aproxime los puntos  $p_i$  de  $P$ , lo cual se puede realizar minimizando la suma de los cuadrados de los residuos. El error total  $E$  de todos los puntos de datos  $p_i$ , aproximado por una función  $f$ , viene dado por la siguiente ecuación:

$$E = \sum_{i=1}^n \| f(p_i) - f_i \|^2 \quad (2.20)$$

Esto lleva al siguiente problema de minimalización:

$$\min_{f \in \Pi_m^d} \sum_{i=1}^n \| f(p_i) - f_i \|^2 \quad (2.21)$$

Donde  $f$  se toma del espacio de polinomios  $\prod_m^d$  de grado total  $m$  en  $d$  dimensiones.

Por ejemplo, se pueden usar cuadráticos en tres dimensiones; por lo tanto,  $d = 3$  y  $m = 2$ , y entonces  $f(x)$  se puede escribir del modo:

$$f(x) = b(x)^T c \quad (2.22)$$

Donde  $b(x) = [b_1(x), b_2(x), \dots, b_k(x)]^T$  es el vector de base polinomial y  $c = [c_1, c_2, \dots, c_k]^T$  es el vector de los coeficientes desconocidos que se desea minimizar en la ecuación 2.21. En general, la cantidad de elementos en  $b(x)$  y  $c$  viene dada por la fórmula  $k = \frac{(d+m)!}{m!d!}$ .

La ecuación 2.21 se puede minimizar estableciendo todas las derivadas parciales en cero, es decir,  $\nabla E = 0$ , con  $\nabla = \left[ \frac{\partial}{\partial c_1}, \frac{\partial}{\partial c_2}, \dots, \frac{\partial}{\partial c_k} \right]$ . Tomando las derivadas parciales con respecto a los coeficientes desconocidos  $c_1, c_2, \dots, c_k$ , se obtiene un sistema lineal de ecuaciones a partir del cual se puede calcular  $c$ :

$$\begin{aligned} \frac{\partial E}{\partial c_1} &= 0 : \sum_{i=1}^n 2b_1(p_i) [b(p_i)^T c - f_i] = 0 \\ \frac{\partial E}{\partial c_2} &= 0 : \sum_{i=1}^n 2b_2(p_i) [b(p_i)^T c - f_i] = 0 \\ &\vdots \\ \frac{\partial E}{\partial c_k} &= 0 : \sum_{i=1}^n 2b_k(p_i) [b(p_i)^T c - f_i] = 0 \end{aligned}$$

Esto se puede escribir en forma de matriz de la siguiente manera:

$$2 \sum_{i=1}^n [b(p_i)b(p_i)^T c - b(p_i)f_i] = 0$$

Cuando se divide por la constante y la reorganización de los términos, esto da:

$$\sum_{i=1}^n [b(p_i)b(p_i)^T c] = \sum_{i=1}^n [b(p_i)f_i]$$

Que se puede resolver como:

$$c = \frac{\sum_{i=1}^n b(p_i)f_i}{\sum_{i=1}^n b(p_i)b(p_i)^T} \quad (2.23)$$

Si la matriz cuadrada  $\mathbf{A} = \sum_{i=1}^n b(p_i)b(p_i)^T$  es no singular, es decir,  $\det(\mathbf{A}) \neq 0$ , la sustitución de 2.23 en 2.22 proporciona la función de ajuste  $f$ .

Por ejemplo, si se quiere ajustar un polinomio cuadrático y bivariado en dos dimensiones, es decir,  $d = 2$  y  $m = 2$  y  $b(x) = [1, x, y, x^2, xy, y^2]$ , entonces el sistema de ecuaciones lineales resultante se ve de la siguiente manera:

$$\sum_{i=1}^n \begin{bmatrix} 1 & x_i & y_i & x_i^2 & x_i y_i & y_i^2 \\ x_i & x_i^2 & x_i y_i & x_i^3 & x_i^2 y_i & x_i y_i^2 \\ y_i & x_i y_i & y_i^2 & x_i^2 y_i & x_i y_i^2 & y_i^3 \\ x_i^2 & x_i^3 & x_i^2 y_i & x_i^4 & x_i^3 y_i & x_i^2 y_i^2 \\ x_i y_i & x_i^2 y_i & x_i y_i^2 & x_i^3 y_i & x_i^2 y_i^2 & x_i y_i^3 \\ y_i^2 & x_i y_i^2 & y_i^3 & x_i^2 y_i^2 & x_i y_i^3 & y_i^4 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} = \sum_{i=1}^n \begin{bmatrix} 1 \\ x_i \\ y_i \\ x_i^2 \\ x_i y_i \\ y_i^2 \end{bmatrix} f_i$$

Cuando se utiliza este enfoque, cada punto tiene un factor de peso igual y constante; por lo tanto, todos los puntos tienen la misma influencia en la función  $f$  resultante, incluso si están muy lejos del centro de aproximación. Para algunas aplicaciones se quiere que las funciones  $f$  den una aproximación local de los puntos  $p_i$ , lo que significa que los puntos cercanos al centro de aproximación deberían tener más influencia en la función  $f$  resultante que los puntos lejanos. Esta extensión se llama *mínimos cuadrados ponderados*, en cuya formulación el error funcional  $E$  ahora contiene un factor de ponderación. La minimalización tiene la siguiente forma:

$$\min_{f \in \prod_m^d} \sum_{i=1}^n \theta(\|a - p_i\|) \|f(p_i) - f_i\|^2 \quad (2.24)$$

Donde  $\theta : \mathbb{R}^3 \rightarrow \mathbb{R}$  asigna un peso a cada punto  $p_i$ , con base en su función de distancia euclíadiana desde el centro de aproximación  $a$ , que es generalmente uno de los puntos en  $P$ . Este problema de minimalización se puede resolver de forma análoga a los mínimos cuadrados, por lo que no se repetirá nuevamente.

La mayoría de las funciones de distancia con peso solo operan dentro de un cierto radio desde el centro de la aproximación  $a$ , ya que los puntos muy lejanos tendrán una influencia muy cercana a cero en la función  $f$  resultante. Al limitar la influencia de estas funciones, solo se debe evaluar una parte del conjunto de puntos, lo que da como resultado un cálculo totalmente local y, por lo tanto, más rápido.

### 2.2.5. Estimación de la normal mediante PCA

El análisis de componentes principales (PCA) implica un procedimiento matemático que transforma un número de variables (posiblemente) correlacionadas en un número

(más pequeño) de variables no correlacionadas llamadas *componentes principales*. El primer componente principal representa la mayor cantidad de variabilidad en los datos como sea posible, y cada componente sucesivo representa la mayor variabilidad restante (Smith, 2002).

En términos computacionales, los componentes principales se encuentran mediante el cálculo de los vectores propios y los valores propios de la matriz de covarianza de los datos. Este proceso es equivalente a encontrar el sistema de ejes en el que la matriz de covarianza es diagonal. El vector propio con el valor propio más grande es la dirección de mayor variación, seguido por la dirección (ortogonal) con la siguiente variación más alta, y así sucesivamente.

Sea  $A$  una matriz  $n \times n$ . Los valores propios de  $A$  se definen como las raíces de:

$$\text{determinante}(A - \lambda I) = |j(A - \lambda I)| = 0$$

Donde  $I$  es la matriz de identidad  $n \times n$ . Esta ecuación se llama *ecuación característica* (o *polinomio característico*) y tiene  $n$  raíces.

Sea  $\lambda$  un valor propio de  $A$ . Entonces existe un vector  $\mathbf{x}$  tal que  $A\mathbf{x} = \lambda\mathbf{x}$ . El vector  $\mathbf{x}$  se denomina *vector propio* de  $A$  asociado con el valor propio  $\lambda$ .

Para encontrar una solución numérica para  $\mathbf{x}$  se necesita establecer uno de sus elementos a un valor arbitrario (por ejemplo, 1), que da un conjunto de ecuaciones simultáneas para resolver los otros elementos. Si no hay solución, se repite el proceso con otro elemento. Por lo general, se normalizan los valores finales para que  $\mathbf{x}$  tenga una longitud, es decir:

$$\mathbf{x} \cdot \mathbf{x}^T = 1$$

En este caso, se está tratando con datos en 3D, lo que da una matriz de  $n \times 3$  de  $\mathbf{x}$ . Esto resulta en una matriz de covarianza de  $3 \times 3$ , como  $A_{n \times 3} \times A_{3 \times n}$  da una matriz  $A_{3 \times 3}$  de tamaño  $3 \times 3$ . Por lo tanto, dada una matriz  $A_{3 \times 3}$  con vectores propios  $x_1, x_2, x_3$  y valores propios  $\lambda_1, \lambda_2, \lambda_3$ , se tiene que:

$$Ax_1 = \lambda_1 x_1$$

$$Ax_2 = \lambda_2 x_2$$

$$Ax_3 = \lambda_3 x_3$$

Cada valor propio corresponde a un vector propio. Generalmente se utiliza el valor propio máximo para encontrar el vector propio correspondiente. Este es un vector de columna que almacena las coordenadas  $(x, y, z)$  de la posición de la normal.

## 2.3. Extracción de la geometría de las regiones

El proceso de modelado geométrico a partir de una región segmentada comienza con la generación de una malla de superficie usando el algoritmo Marching Cubes (MC) (Lorenzen y Cline, 1987) (subsección 2.3.1), el cual utiliza un isovalor para extraer y renderizar una isosuperficie desde los contornos segmentados. Esta malla inicial forma los datos de entrada de muchos métodos y algoritmos tanto paramétricos y de geometría computacional como de reconstrucción implícita (sección 2.2). En esta sección se estudiarán adicionalmente los métodos para extraer la representación geométrica a partir de las reconstrucciones paramétricas (subsección 2.3.2), las reconstrucciones con base en geometría computacional (subsección 2.3.3) y las reconstrucciones implícitas (subsección 2.3.4).

### 2.3.1. Marching Cubes

El principio detrás del algoritmo MC es subdividir el espacio en una serie de pequeños cubos conocidos como véxeles (figura 2.8). Cada punto de la malla es un vértice de la esquina de un cubo, y todos ellos están definidos por el mallado del volumen. Como cada uno de los ocho vértices de un cubo puede marcarse o no, existen  $2^8$  ( $256$ ) combinaciones posibles del estado de la esquina. Estas combinaciones se simplifican a 15 teniendo en cuenta las celdas que se duplican en las siguientes condiciones:

- Al rotar en cualquier grado sobre cualquiera de los tres ejes principales.
- Al reflejar la forma en cualquiera de los tres ejes principales.
- Al invertir el estado de todas las esquinas y voltear las normales de los polígonos relacionados.

El algoritmo avanza a través de cada uno de los cubos mientras prueba los puntos de esquina y reemplaza el cubo con un conjunto apropiado de polígonos. Cada vértice y cada arista del cubo (figura 2.8a) se indexan en tablas a las que accede el algoritmo MC para buscar y determinar qué aristas del véxel se cruzan con la isosuperficie, para así

poder crear parches triangulares que dividen el cubo en diferentes regiones que están dentro de la isosuperficie y por fuera de ella (figura 2.8b). Al conectar los parches de todos los véxeles en el límite de la isosuperficie, se obtiene una representación de superficie que se aproxima a la que describe el conjunto de datos volumétricos.

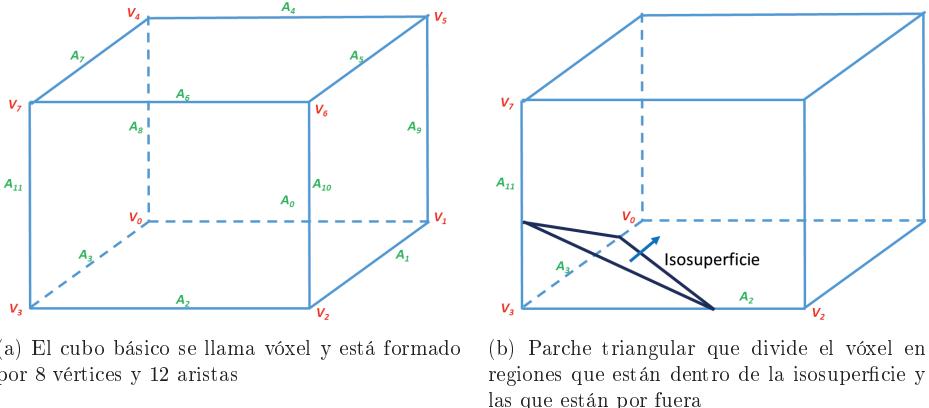


Figura 2.8: Algoritmo Marching Cubes

La figura 2.8a muestra el orden de indexación de las aristas  $A_i$  y los vértices  $V_i$ , mientras que la figura 2.8b corresponde a un caso en el que el vértice  $V_3$  está por debajo del valor de isosuperficie y esta última corta las aristas  $A_2$ ,  $A_3$  y  $A_{11}$ . La posición exacta de los vértices del parche triangular, en color azul oscuro, depende de la relación del valor de la isosuperficie y de los valores en los vértices 3-2, 3-0 y 3-7, respectivamente. La flecha verde indica el vector normal a la superficie del triángulo generado.

En caso de que un vértice esté por encima de la isosuperficie y un vértice adyacente esté por debajo, entonces la posición en la que la isosuperficie corta el borde es interpolada linealmente. La relación de la longitud entre los dos vértices será la misma que la relación del valor de la isosuperficie con los valores en los vértices de la celda de la grilla. Los puntos de intersección se pueden calcular por interpolación lineal. Si  $V_1$  y  $V_2$  son los vértices de un borde de corte y  $val_1$  y  $val_2$  son los valores escalares de cada vértice, el punto de intersección  $P$  viene dado por la siguiente ecuación:

$$P = V_1 + (isovalor - val_1)(V_2 - V_1)/(val_2 - val_1)$$

El paso final del algoritmo MC es calcular la normal de las caras de la triangulación para obtener una representación correcta de la superficie. Sin embargo, como los datos

segmentados de las imágenes médicas son en general grandes, irregulares e inexactos, la triangulación que produce MC refleja estos defectos (figura 3.5), lo cual impide usar este tipo de geometrías directamente en aplicaciones de ingeniería (Lin *et al.*, 2001).

### 2.3.2. Superficies paramétricas

Tradicionalmente, el algoritmo de reconstrucción de superficie paramétricas consta de cuatro pasos principales:

1. Generación de la malla a partir de la nube de puntos no organizada. Esto puede hacerse, por ejemplo, utilizando el algoritmo de Marching Cubes, triangulaciones de Delaunay o Alpha Shapes.
2. Partición de la malla en parches homeomórficos a discos. Estos parches también se conocen como mapas (*charts*). La partición de la malla de superficie se vuelve obligatoria cuando está cerrada o tiene cavidades ( $\text{genus} > 0$ ).
3. Para cada parche de malla se construye una parametrización local. Estas parametrizaciones locales se realizan juntas de forma continua, de modo que forman colectivamente una parametrización globalmente continua de la malla.
4. Terminado el paso de parametrización, el cual genera una colección de pares de parámetros  $(u_i, v_i)$  asociados a los puntos  $(x_i, y_i, z_i)$  de la nube, sigue el problema del ajuste de la superficie.

Este último problema consiste en minimizar la distancia entre cada punto  $(x_i, y_i, z_i)$  y su punto correspondiente de la superficie  $F(u_i, v_i)$ . El enfoque estándar de ajuste superficial se reduce al siguiente problema de minimización:

$$\min \sum_i \| \mathbf{x}_i - F(u_i, v_i) \|^2$$

Donde  $\mathbf{x}_i$  es el  $i$ -ésimo punto de la nube de entrada  $(x_i, y_i, z_i)$  y  $\| \cdot \|$  es la distancia euclíadiana entre  $\mathbf{x}_i$  y el punto correspondiente en la superficie  $F(u_i, v_i)$  en el espacio lineal de funciones mencionado. La función objetivo de este problema de minimización es la norma euclíadiana cuadrada. Su cálculo se puede hacer fácilmente por el método de mínimos cuadrados (subsección 2.2.4); de ahí el ajuste de estos últimos para la reconstrucción de superficies paramétricas (Cohen *et al.*, 2001), (Farin y Farin, 2002).

### 2.3.3. Superficies triangulares

La extracción de superficies con base en métodos de geometría computacional constan de dos pasos y termina obteniendo una partición espacial compuesta de tetraedros:

- Primero se construye la triangulación de Delaunay a partir de una nube de puntos, que consiste en una partición del casco convexo de estos en un conjunto finito de tetraedros.
- Terminada la triangulación de la nube de puntos, solo resta identificar y extraer los triángulos o simplices que pertenecen a la superficie.

Por consiguiente, la reconstrucción de las superficies triangulares consiste en encontrar el subgrafo de la triangulación de Delaunay del conjunto inicial de puntos (Edelsbrunner y Mücke, 1994). Sin embargo, la identificación de los triángulos de la superficie varía de un algoritmo a otro.

### 2.3.4. Superficies implícitas

Para superficies dadas como el conjunto de nivel cero  $d = 0$  de una función implícita  $f(x, y, z)$  se puede generar una malla de superficie aproximando la isosuperficie mediante el uso del algoritmo MC. Las funciones implícitas generalmente se muestran en una cuadrícula subyacente (Hoppe *et al.*, 1992; Carr *et al.*, 2001), donde la superficie reconstruida se encuentra a través de un isocontorno con un isovalor apropiado. Para adaptar la resolución de la cuadrícula a una densidad de muestreo puntual, otros utilizan cuadrículas como octrees (Kazhdan *et al.*, 2006; Kazhdan y Hoppe, 2013) y triangulaciones 3D adaptativas (Dey *et al.*, 2012). Sin embargo, extraer la superficie a partir de un octree presenta varias dificultades para garantizar que las mallas de superficie sean cerradas.

Extraer la superficie de una triangulación se puede realizar a través de Marching Tetrahedra, pero si se desea una malla con un número menor de triángulos y triángulos bien acondicionados, se pueden utilizar las técnicas de refinado de Delaunay (Dey *et al.*, 2012).

## 2.4. Extracción de la geometría de las estructuras heterogéneas

Para el modelado geométrico de estructuras naturales heterogéneas a partir de imágenes médicas segmentadas se cuenta con tres aproximaciones. En primer lugar, generar el modelo directamente a partir de las imágenes médicas segmentadas, utilizando algún método que extienda el algoritmo MC a múltiples regiones y refine la malla hasta obtener la calidad deseada (Wu y Sullivan Jr, 2003) (subsección 2.4.1). En segundo lugar, se pueden utilizar métodos mediante el refinamiento de Delaunay (Dey *et al.*, 2012) (subsección 2.4.2). Finalmente, modelar las partes que conforman la estructura heterogénea de manera independiente y después aplicar un método de ensamble (subsección 2.4.3). En este caso, las regiones que conforman la estructura heterogénea  $\Omega$  deben conformar una partición del espacio.

### 2.4.1. Extensión del algoritmo Marching Cubes

En la subsección 2.3.1 se observó que cuando se tiene un etiquetado binario, se puede utilizar el algoritmo MC. Cuando se tiene más de una etiqueta, se pueden utilizar extensiones de este algoritmo mediante reglas adicionales y una tabla de búsqueda extendida.

Sin embargo, estos métodos pueden sufrir algunos problemas, como ambigüedad en el etiquetado, defectos topológicos y un aumento algebraico en el número de casos con respecto a la cantidad de materiales. El resultado es que se pueden crear mallas con elementos que no coincidan de forma consistente. Se han propuesto varias soluciones a estos problemas, como la reparación de vértices ambiguos antes de extraer las mallas (Bloomenthal y Ferguson, 1995), la subdivisión de dominio (Reitinger *et al.*, 2005) y la asignación e interpolación de probabilidades en vértices para extraer elementos superficiales topológicamente correctos (Hege *et al.*, 1997).

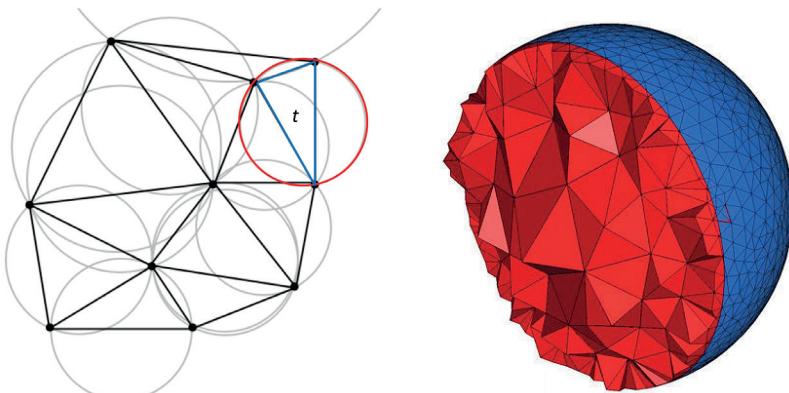
### 2.4.2. Refinamiento de Delaunay

Los métodos mediante el refinamiento de Delaunay son una extensión de los algoritmos típicos de refinamiento de triangulaciones que eliminan aquellos triángulos que poseen algún ángulo inferior a un umbral deseado, mediante la inserción de nuevos puntos en la triangulación original (Berg *et al.*, 2008; Dey *et al.*, 2012). En general, estos métodos

implican muchos pasos sutiles de preprocesamiento, incluida la necesidad de identificar y extraer las uniones triples como una red de curvas (Saye, 2015).

La triangulación de un conjunto  $P$  de puntos sobre el plano es de Delaunay si y solo si el circuncírculo<sup>1</sup> de cualquier triángulo  $t$  de la malla no contiene un punto de  $P$  en su interior. Esto se puede apreciar en la figura 2.9a, donde el circuncírculo en color rojo solo contiene aquellos puntos que forman el triángulo  $t$  en color azul.

Para obtener la triangulación de Delaunay, inicialmente se construye un gran triángulo que contiene todos los puntos del conjunto. Esto implica agregar tres puntos adicionales al conjunto de puntos  $P$ , que al final del proceso son eliminados. Este triángulo se considera como la triangulación inicial del conjunto  $P$  (You *et al.*, 2015). Posteriormente, se inserta cada punto a la vez hasta tener una triangulación que contenga todos los vértices del conjunto inicial  $P$ . La figura 2.9a muestra el resultado en 2D para un conjunto de 10 puntos, mientras que la figura 2.9b representa el proceso de refinamiento en 3D.



(a) Verificación de que el circuncírculo de  $t$  no contenga un vértice de otro triángulo en su interior  
 (b) Tetraedrallización de una esfera hecha en CGAL

Figura 2.9: Triangulación de Delaunay en 2D y 3D

Las mallas creadas son topológicamente consistentes, no tienen espacios vacíos ni superposiciones y, además, los elementos de malla que se encuentran en las uniones lo hacen compartiendo un borde común. Estos enfoques tienen la ventaja de permitir que los criterios de calidad se definan por superficie, haciendo que diferentes materiales

<sup>1</sup>Circuncírculo corresponde a la circunferencia que pasa por todos los vértices de un polígono regular. También se conoce como *circunferencia circunscrita*.

tengan diferentes resoluciones de malla; y las mallas de tetraedros de volumen, consistentes con la topología de la interfaz, a menudo se producen sin costo adicional.

### 2.4.3. Métodos de ensamble

Los modelos de ensamble son una forma natural e intuitiva de representar estructuras heterogéneas. Para representar estructuras heterogéneas usando un modelo de ensamblaje, se debe establecerse correctamente las particiones espaciales. Hay al menos dos métodos para lograr esto: el enfoque constructivista, que genera divisiones espaciales de los límites de las regiones o partes existentes, y la representación *rm-set*.

Kumar y Dutta (1997) introdujeron los conceptos de *rm-set* y *rm-object* para la representación de objetos sólidos heterogéneos. La representación *rm-set* es una extensión directa del modelo *r-set*, con distribuciones de materiales adicionales incluidas. Por lo tanto, los objetos heterogéneos están representados con múltiples *rm-set*. Cada *rm-set* se supone que es homogénea, pero diferentes *rm-set* pueden tener diferente composición de material.

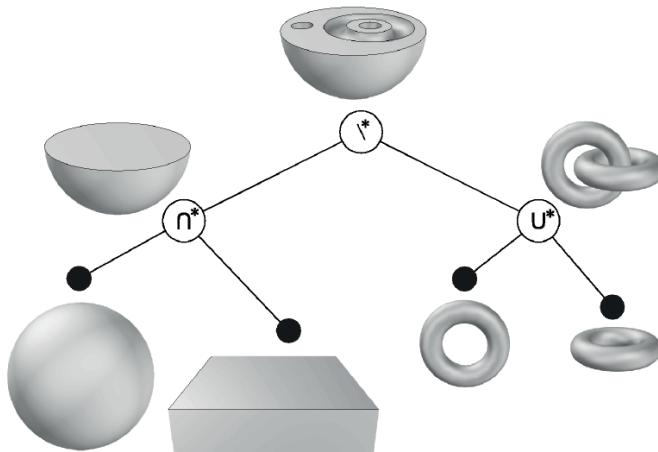


Figura 2.10: Modelado de un objeto sólido a partir de un árbol booleano  
Fuente Csikós (2006)

Otros investigadores prefieren el modelado constructivista, que es una manera elegante de construir modelos de objetos sólidos usando una estructura de datos en árbol con operaciones en los nodos y primitivas en las hojas (Wang *et al.*, 2011).

Las operaciones booleanas tradicionales se basan en la premisa de que los objetos son homogéneos en la composición material. La figura 2.10 muestra el modelado en 3D de un objeto sólido a partir de un árbol con cuatro nodos: una esfera, un rectángulo y dos toroides. Entre la esfera y el rectángulo se aplica una operación booleana de intersección, lo que da como resultado la parte inferior de la esfera. Entre los dos toroides se aplica una operación de unión. Finalmente, se calcula la diferencia entre la media esfera y la unión de los toroides para obtener el objeto deseado.

Se puede ver en la figura 2.10 que este método aplica solamente para el modelado de objetos sólidos compuestos de un solo material. Para modelar objetos heterogéneos es necesario ampliar las operaciones booleanas. Sun y Hu (2002) propusieron los operadores booleanos Ma dominante y Mb dominante, y aplicaron pesos de mezcla constantes W en cada subconjunto (región) (figura 2.11). Para el caso de la unión con Ma dominante se tiene que  $W_a = 1$  y  $W_b = 0$ , mientras que para la unión con Mb dominante se tiene que  $W_a = 0$  y  $W_b = 1$ .

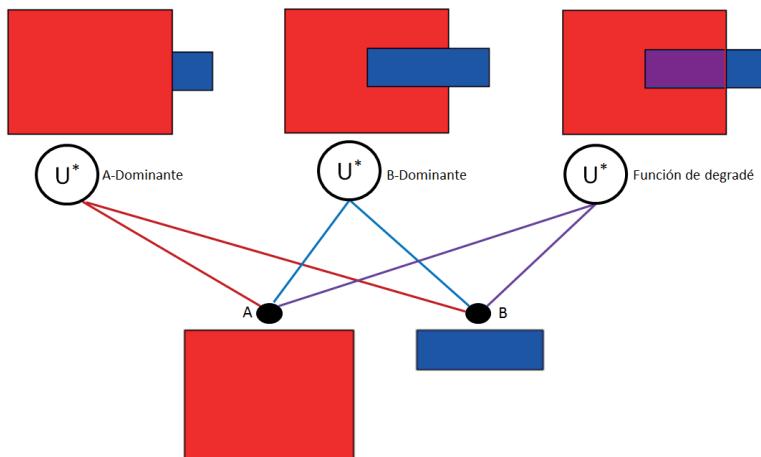


Figura 2.11: Tres diferentes resultados de aplicar la operación booleana de unión sobre dos primitivas de entrada A y B: unión dominante Ma, unión dominante Mb y unión con función de degradé (Sun y Hu, 2002)

La figura 2.11 muestra adicionalmente el caso de utilizar una función de degradé relacionada con el operador de unión, donde  $W_a = 0,5$  y  $W_b = 0,5$ . Adicionalmente, Siu y Tan (2002) propusieron operadores similares de inserción y de inmersión en su esquema basado en fuentes (*sources*).



# Método de modelado RAM

---

En este capítulo se presenta el método RAM (Region Aware Modeling Method) para el modelado 3D de estructuras anatómicas heterogéneas. Este método es el producto de varias investigaciones realizadas por el autor en el área de computación gráfica y modelado de objetos heterogéneos. El método RAM se basa en un enfoque de *dividir y conquistar* a través del cual un modelo matemático combina las características de un conjunto de superficies  $\Gamma_i$  y regiones  $\Omega_i$  mediante la relación  $\Gamma_i = \partial\Omega_i$  (sección 3.1). Esto permite que tanto la información de la geometría de las regiones como de toda la estructura anatómica se incorpore en el modelo matemático a través de un vector de funciones implícitas. El método propuesto consta de dos procesos principales: el de reconstrucción implícita (sección 3.2) y el de extracción de la geometría mediante mallas poligonales (sección 3.3).

## 3.1. Modelado matemático

Una estructura heterogénea es una colección o grupo de elementos, llamados *regiones*, con una interfaz que las separa, donde cada región puede tener asociado un tipo de material (Wang y Wang, 2005). Matemáticamente, las estructuras heterogéneas se pueden describir como:

$$\Omega = \bigcup_{i=1}^n \Omega_i \quad (3.1)$$

Donde  $\Omega$  es la estructura heterogénea modelada en  $\mathbb{R}^3$ ,  $\Omega_i$  es la i-ésima región que pertenece a  $\Omega$  y  $n$  es el número de regiones. La estructura  $\Omega$  puede estar definida por

una función  $F(\mathbf{p})$  donde el punto  $\mathbf{p} \in \mathbb{R}^3$ . El espacio euclíadiano  $\mathbb{R}^3$  puede extenderse a  $\mathbb{R}^4$  para representar la composición del material  $m$  de cualquier punto  $\mathbf{p}$  mediante la expresión:

$$\mathbf{p} = \{x, y, z, m\}^T \quad (3.2)$$

Dado que cada región  $\Omega_i$  tiene una superficie límitrofe cerrada  $\Gamma_i = \partial\Omega_i$ , la interfaz entre dos regiones  $i$  y  $j$  se puede denotar por:

$$\Omega_i \bigcap_{i \neq j} \Omega_j = \Gamma_i \bigcap_{i \neq j} \Gamma_j = \Gamma_{ij} = \Gamma_{ji} \quad (3.3)$$

Si dos regiones  $i$  y  $j$  no tienen frontera en común, entonces la interfaz estará vacía.

Ahora, sea  $\chi_{\Omega_i}$  la función indicatriz de la región  $\Omega_i$  tal que:

$$\chi_{\Omega_i}(\mathbf{p}) = \begin{cases} 1, & \mathbf{p} \in \Omega_i \\ 0, & \mathbf{p} \notin \Omega_i \end{cases} \quad (3.4)$$

Si  $f_i(\mathbf{p})$  es una función definida en  $\mathbb{R}^3$ , entonces la función  $\chi_{\Omega_i}(\mathbf{p})f_i(\mathbf{p})$  tiene la siguiente propiedad:

$$\chi_{\Omega_i}(\mathbf{p})f_i(\mathbf{p}) = \begin{cases} f_i(\mathbf{p}), & \mathbf{p} \in \Omega_i \\ 0, & \mathbf{p} \notin \Omega_i \end{cases} \quad (3.5)$$

Para la función  $f_i(\mathbf{p})$ , una vez esta se multiplica por la función  $\chi_{\Omega_i}(\mathbf{p})$ , su influencia fuera de la región  $\Omega_i$  disminuye, ya que el valor de la función resultante será cero para  $\mathbf{p} \notin \Omega_i$ , mientras que el valor de  $f_i(\mathbf{p})$  para  $\mathbf{p}$  dentro de la región  $\Omega_i$  no se cambiará.

Ahora, para una región  $\Omega_i$  representada implícitamente por la ecuación  $f_i(\mathbf{p}) = 0$ , considere la siguiente ecuación:

$$\chi_{\Omega_i}(\mathbf{p})f_i(\mathbf{p}) + (1 - \chi_{\Omega_i}(\mathbf{p}))\delta = 0, \quad \delta > 0 \quad (3.6)$$

Como se puede ver, esta tendrá exactamente las mismas soluciones que la ecuación 3.5 cuando se confine en la región  $\Omega_i$ . Sin embargo, la parte de la forma original definida por  $f_i(\mathbf{p}) = 0$  fuera de la región  $\Omega_i$  ha sido eliminada ya que cuando  $\mathbf{p} \notin \Omega_i$ , la ecuación 3.6 es diferente de cero.

Entonces, si se tiene en consideración que la estructura heterogénea  $\Omega$  es una partición<sup>1</sup>, esto es,  $\Omega = \bigcup_{i=1}^n \Omega_i$ , y que la forma geométrica de la región  $\Omega_i$  se representa implícitamente como  $f_i(\mathbf{p}) = 0$ , entonces el modelo geométrico general de la estructura heterogénea se puede expresar como:

$$F(\mathbf{p}) = \sum_{i=1}^n \chi_{\Omega_i}(\mathbf{p}) f_i(\mathbf{p}) \quad (3.7)$$

## 3.2. Proceso de reconstrucción implícita

El primer proceso en el modelado de las estructuras heterogéneas es la reconstrucción implícita de la superficie de las regiones que conforman una estructura (algoritmo 3.1).

---

**Algoritmo 3.1** Reconstrucción implícita de la geometría de las regiones que conforman la estructura heterogénea

Entrada: Imagen TAC segmentada con el conjunto de  $n$  regiones que componen la estructura heterogénea  $\Omega$ .

Salida: Conjunto de funciones implícitas  $f_i$ , organizadas en un VFI y un VFC.

Para todas las regiones  $\Omega_i \in \Omega$   $1 \leq i \leq n$

  Extraer región  $\Omega_i$ :  $Modelo \leftarrow Extraer(Region\Omega_i)$ .

  Generar nubes de puntos:  $P_i \leftarrow Transformar(Modelo)$ .

  Calcular función implícita:  $f_i \leftarrow Reconstruir(P_i)$ .

  Agregar la función  $f_i$  al VFI:  $VFI \leftarrow Agregar(f_i)$ .

  Actualizar VFC:  $VFC \leftarrow Actualizar()$ .

---

Fin Para

Este proceso se lleva a cabo, segmentando inicialmente cada región que compone la estructura (subsección 3.2.1), de donde se extrae una nube de puntos junto con el conjunto de normales a la superficie (subsección 3.2.2). A partir de esta nube de puntos se construye una representación geométrica compacta con base en funciones implícitas (subsección 3.2.3). Finalmente, se implementa un vector de funciones implícitas para representar y manipular el modelo computacional (subsección 3.2.4).

---

<sup>1</sup>Supóngase que  $\chi = \{\chi_{\Omega_i} : i = 1, 2, \dots, n\}$  es un conjunto de  $n$  funciones indicatrices en un dominio dado  $D \in \mathbb{R}^3$ . Se dice que  $\chi$  es una partición de una estructura heterogénea si para cada  $\mathbf{p} \in \Omega$ , entonces  $\sum_{i=1}^n \chi_{\Omega_i}(\mathbf{p}) = 1$ .

### 3.2.1. Extracción de regiones

El primer paso del algoritmo 3.1 es obtener la segmentación de cada una de las regiones que conforman la estructura heterogénea. El procedimiento se llama  $\text{Extraer}(\text{Region } \Omega_i)$ . Para esta tarea se utilizan imágenes TAC segmentadas que se pueden ver como una secuencia de datos dispuestos en una grilla tridimensional que almacena parámetros correspondientes a las diferentes regiones, por ejemplo, etiquetas de color (figura 3.1).

**Definición 1.** Región: en una imagen segmentada, es el conjunto máximo de véxeles que tienen el mismo valor o etiqueta.

En una imagen segmentada cada región es un conjunto de véxeles 6-conectados (figura 2.5). Dos véxeles con la misma etiqueta  $l$  pertenecen a la misma región con la etiqueta  $l$ , y dos regiones diferentes tienen dos valores diferentes de  $l$ . Considerar solo imágenes segmentadas es una optimización que permite recuperar inmediatamente, dada la etiqueta de un véxel, el componente conectado al que pertenece, llamado también *región*.

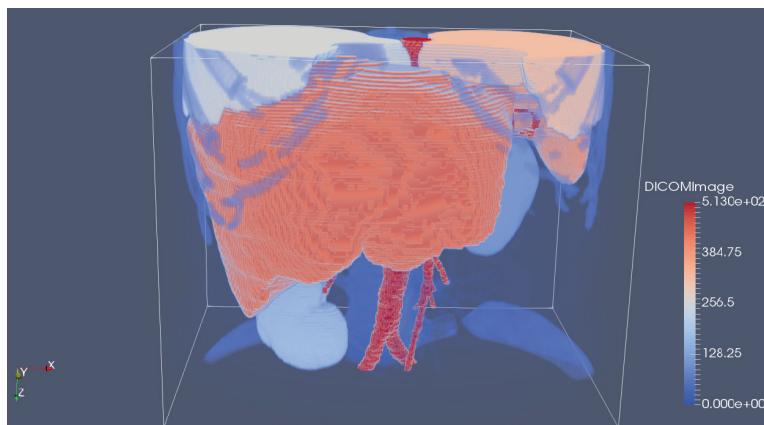


Figura 3.1: Regiones segmentadas a partir de una imagen TAC y etiquetadas por color

Adicionalmente, se debe tener en cuenta que se usará la etiqueta de las regiones como un identificador único en la construcción del *VFI* (subsección 3.2.4).

### 3.2.2. Generación de la nube de puntos

El método RAM propone reconstruir la geometría tanto de las regiones como de la estructura heterogénea con base en una reconstrucción de superficies implícita. El

requisito de entrada de todos los métodos de reconstrucción implícita es un conjunto de puntos tridimensionales  $P$  tomados de la superficie del objeto por modelar. El conjunto de puntos  $P$  suele obtenerse mediante escáneres láser, sondas táctiles y técnicas de visión por computador (Ohtake *et al.*, 2005). Sin embargo, para el método RAM se decidió extraer los puntos de las superficies junto con las normales a partir de cada región segmentada mediante el procedimiento *Extraer(Region $\Omega_i$ )* (figura 3.2).

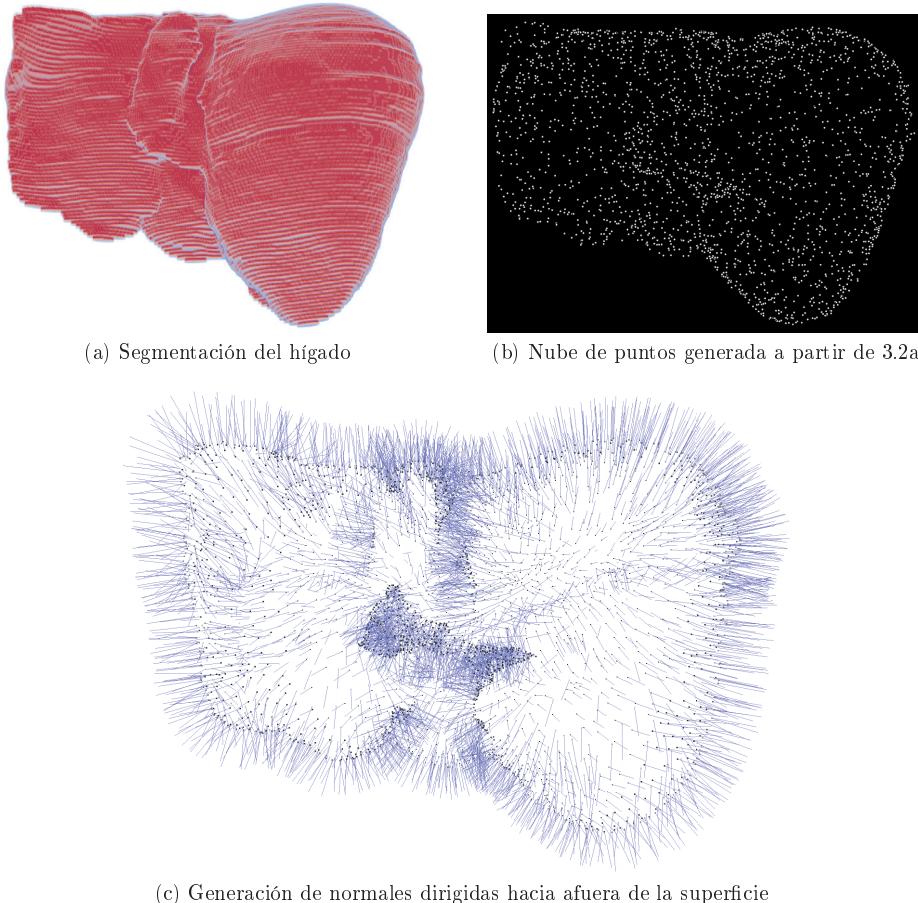


Figura 3.2: Generación de la nube de puntos a partir de la segmentación de un hígado (solo se muestra un 10%)

La figura 3.2a muestra un hígado segmentado, el cual es la entrada del procedimiento llamado *Transformar(Modelo)*, mediante el cual se realiza la triangulación de la superficie de la segmentación utilizando el algoritmo Marching Cubes (Lorensen y Cline,

1987) (sección 2.3.1). Luego se extrae la información relacionada con la posición de cada vértice de la malla de triángulos y se adiciona a la nube de puntos  $P_i$  (figura 3.2b).

En algunas ocasiones, saber hacia dónde se dirige una superficie es de vital importancia. De hecho, la mayoría de algoritmos de reconstrucción implícita esperan como entrada un conjunto normales  $N$  junto con la nube de puntos  $P$ , es decir, para cada punto  $\mathbf{p}_i \in P$  hay una normal  $\mathbf{n}_i \in N$  asociada (figura 3.2c).

Las normales de superficie se suelen calcular directamente desde la nube de puntos. Un método popular y simple para determinar la normal en un punto dado  $\mathbf{p}$  es mediante el análisis de componentes principales (PCA) en un vecindario local de  $\mathbf{p}$  (Hoppe *et al.*, 1992). Más específicamente, si un vecindario local de  $\mathbf{p}$  se denota por  $N_p$ , la forma más básica de aplicar PCA es mediante el análisis de los vectores propios  $C_p$  de la matriz de covarianza (subsección 2.2.5):

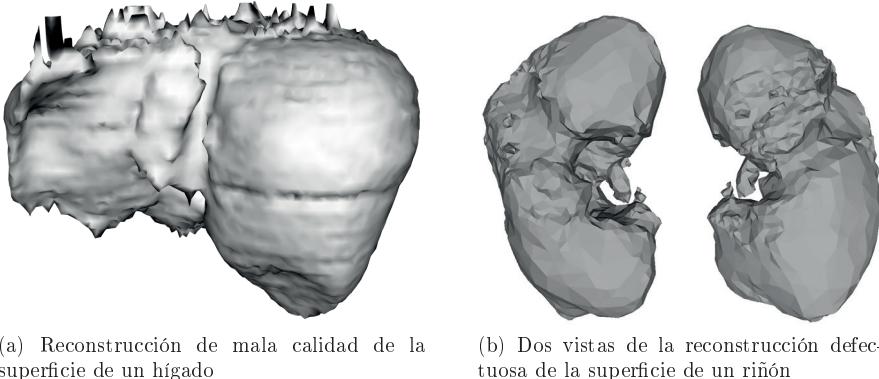
$$C_p = \sum_{\mathbf{q} \in N_p} (\mathbf{p} - \mathbf{q})(\mathbf{p} - \mathbf{q})^T \quad (3.8)$$

El vector propio de  $C_p$  asociado con el valor propio más pequeño define la normal no orientada. Se debe tener en cuenta que si el valor propio más pequeño es 0, entonces la región definida por  $\mathbf{p}$  y  $N_p$  es plana, ya que los vectores propios asociados con los segundos valores propios más grandes capturan toda la varianza en los datos. Sin embargo, el problema aquí es encontrar un vecindario local de puntos que sea lo suficientemente pequeño como para representar con precisión el espacio tangencial de un punto. Del mismo modo, la escala del vecindario debe ser proporcional a la densidad del muestreo en ese punto. Esto podría conducir a normales de superficie mal orientadas que impactan directamente en la calidad de la reconstrucción final (figura 3.3).

Como la estimación de la densidad de muestreo es en sí misma un tema desafiante, particularmente cuando se enfrenta a un muestreo no uniforme que se presenta en las imágenes TAC con un espaciado anisotrópico, la información relacionada con las normales se calcula directamente a partir de la propia segmentación. Sin embargo, hay algunos factores que hacen que la estimación de las normales de superficie sea especialmente difícil:

- Debido a la naturaleza binaria de las segmentaciones, siempre hay un cambio abrupto en los valores de los datos en la superficie del modelo, lo que deriva en defectos de reconstrucción debido al submuestreo (figura 3.3a).

- Las normales deben tener direcciones consistentes, ya sea apuntando hacia adentro o hacia afuera de la superficie de la región. No contar con esta condición conlleva reconstrucciones erróneas de la superficie (figura 3.3b).



(a) Reconstrucción de mala calidad de la superficie de un hígado

(b) Dos vistas de la reconstrucción defec-tuosa de la superficie de un riñón

Figura 3.3: Efectos negativos de normales mal orientadas sobre la reconstrucción geométrica de un objeto

Por lo tanto, cuando se desea la superficie tangencial de un voxel específico, es necesario echar un vistazo a su entorno. Para esto, se define un radio  $r$  de voxels, donde se especifican los voxels que se encuentran en la superficie. Dentro del radio definido, se derivarán todos los voxels de superficie de primer orden. Así, para un radio  $r = 1$ , la normal orientada  $\mathbf{n}$  a la superficie viene dada por (Hartmann, 1999):

$$\mathbf{n} = \frac{\nabla f}{\| \nabla f \|}, \quad \nabla f = \frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \quad (3.9)$$

Esta normal se calcula utilizando un operador de gradiente de diferencias centradas de seis puntos:

$$\mathbf{n} = (d(x+1, y, z) - d(x-1, y, z), d(x, y+1, z) - d(x, y-1, z), d(x, y, z+1) - d(x, y, z-1))$$

No siempre es claro el signo del vector normal a la superficie calculado de esta manera. No se sabe realmente si el vector normal de superficie apunta hacia adentro o hacia fuera. Sin embargo, como el resultado del proceso de segmentación es un modelo sólido, no solo un cascarón, se puede resolver fácilmente esta ambigüedad verificando los voxels a cada lado de la segmentación. Por un lado, debería haber voxels que forman parte de la región; por otro, debería haber voxels que forman parte del fondo y no pertenecen a la región. Esta suposición es válida ya que solo se consideran voxels de superficie. La

figura 3.2c representa un conjunto de normales a la superficie que apuntan hacia afuera de esta.

### 3.2.3. Cálculo de la función implícita

El resultado del procedimiento *Transformar(Modelo)* (subsección 3.2.2) es una nube de puntos orientados  $P_i = \{s_j = (p_1, n_1), \dots, s_m = (p_m, n_m)\}$  que consiste en muestras  $s_j$  con una posición  $s.p$  y una normal a la superficie  $s.n$  que mira hacia adentro. Cada muestra  $s_j$  se supone que se encuentra sobre o cerca de la superficie  $\partial\Omega_i$  de la segmentación de la región  $\Omega_i$ . Con estos datos se procede a calcular la función implícita  $f_i$  (algoritmo 3.1) a partir de la cual se pueda reconstruir la geometría de las diferentes regiones. El procedimiento para calcular la función implícita  $f_i$  se llama *Reconstruir( $P_i$ )*.

Con base en el análisis realizado en la sección 2.2 se optó por el algoritmo de reconstrucción de superficies de Poisson (Kazhdan *et al.*, 2006) (subsección 2.2.3). Formular la reconstrucción de una superficie implícita como un problema de Poisson ofrece una serie de ventajas. Muchos métodos de ajuste de superficie implícitas primero dividen los datos en regiones para un ajuste local y luego combinan estas aproximaciones locales utilizando funciones de fusión. Por el contrario, la reconstrucción de Poisson es una solución global que considera todos los datos a la vez, sin recurrir a particiones heurísticas y combinaciones.

Al igual que las funciones de base radial, la reconstrucción de Poisson crea superficies lisas que se adaptan de forma robusta a datos ruidosos. Pero, mientras que los RBF ideales son de soporte global y no decaen, la reconstrucción de Poisson admite una jerarquía de funciones de soporte local y, por lo tanto, su solución se reduce a un sistema lineal disperso bien condicionado.

La segmentación de una región  $\Omega_i$  tiene un valor para los puntos que están dentro de la región y otro para que están fuera de ella. Con base en esta observación, el campo normal que señala hacia adentro del límite de la segmentación puede interpretarse como el gradiente de la función indicatriz de la región  $\chi_{\Omega_i}$ .

El algoritmo de reconstrucción de Poisson toma como entrada un conjunto de puntos orientados  $P_i$ . A partir de estos se construye una triangulación 3D de Delaunay. La normal a la superficie de cada punto agregado durante el refinamiento de esta triangulación se establece en cero. Luego se resuelve en cada vértice de la triangulación de Delaunay la ecuación de Poisson para la función indicatriz  $\chi_{\Omega_i}$ , representada como una función lineal a trozos usando un solucionador lineal disperso.

La reconstrucción de la superficie de Poisson depende de una serie de parámetros. La calidad de la reconstrucción, el tiempo de cálculo, etc., se ven afectados por estos parámetros de control, por lo que es importante su correcta configuración.

- **Profundidad del octree.** Profundidad que se utiliza durante el proceso de reconstrucción. Un octree de profundidad  $D$  produce una malla tridimensional de resolución  $2D \times 2D \times 2D$ . A medida que se incrementa la profundidad del octree, aumenta la resolución de la malla. Entonces, el consumo de memoria también se incrementa drásticamente. El valor predeterminado de este parámetro es ocho.
- **Solver divide.** Especifica la profundidad hasta la cual se usa un solucionador de gradiente conjugado para resolver la ecuación de Poisson. Más allá de esta profundidad, se usa la relajación de Gauss-Seidel. Si se aumenta la división del solucionador, el tiempo de cálculo disminuye a mayor valor y se usa la relajación de Gauss-Seidel en lugar del solucionador de gradiente para resolver la ecuación.
- **Muestras por nodo.** Indican el número mínimo de puntos asignados en cada nodo hoja del octree. En caso de datos ruidosos, se asigna una mayor cantidad de puntos en un nodo, de modo que la superficie se interpola usándolos. Esto da como resultado la anulación del efecto del ruido en la superficie 3D generada. Sin embargo, cuando se dispone de datos de nubes de puntos precisos y libres de ruido, se puede asignar un valor de 1-5 a esta variable.
- **Compensación de superficie.** Indica un valor de corrección de umbral para la superficie reconstruida. El valor de 1 indica que no hay corrección, un valor menor que 1 se usa para la compensación interna y un valor mayor que 1 se usa para la compensación externa.

### 3.2.4. Creación de los vectores $VFI$ y $VFC$

El modelado de estructuras heterogéneas es esencialmente un modelado tanto geométrico como de material. En las subsecciones 3.2.1, 3.2.2 y 3.2.3 se ha mostrado cómo las diferentes regiones de la estructura heterogénea se pueden modelar de forma independiente utilizando funciones implícitas  $f_i$ . Pero estas funciones deben agruparse de alguna forma para representar y manipular el modelo final de la estructura heterogénea  $\Omega$ , ya que las regiones deben ser consideradas como un todo en lugar de partes aisladas.

El conjunto de funciones implícitas  $f_i$  ( $i = 1, \dots, n$ ) se podría organizar en un árbol de geometría sólida constructiva para generar la forma de la estructura heterogénea (Ricci, 1973; Pasko *et al.*, 1995; Wang *et al.*, 2011). Sin embargo, el modelado sólido implícito utiliza una función de distancia con signo (*FDS*) que define el sólido asignando a cada punto en el espacio la distancia más corta entre el punto actual y cualquier que pertenezca a la superficie (Ricci, 1973). El signo se usa para distinguir entre el interior y el exterior de una región y así poder recorrer el árbol de *FDS* (Wang *et al.*, 2011). La construcción del árbol de *FDS* se realiza mediante combinaciones de las *FDS*. Estas combinaciones crecen exponencialmente  $2^n$ , donde  $n$  es el número de *FDS*, lo cual se hace inmanejable de configurar.

Teniendo en cuenta el modelo matemático expuesto en la sección 3.1 y siguiendo la idea de Vese y Chan (2002), el concepto de partición de una estructura heterogénea se puede describir mejor con un vector de funciones implícitas (*VFI*)  $F$  y un vector de funciones indicatrices (*VFC*)  $\chi(F)$  correspondiente a  $F$ . El conjunto de  $n$  funciones implícitas  $f_i$  está organizado en un *VFI* de modo que:

$$F(\mathbf{p}) = [f_1(\mathbf{p}), f_2(\mathbf{p}), \dots, f_n(\mathbf{p})], \quad \mathbf{p} \in \mathbb{R}^3 \quad (3.10)$$

$F$  debe ser mínimo, es decir, cada dimensión en  $F$  representa una región única rellena del material correspondiente a dicha dimensión. La etiqueta de material  $i$  se asigna a un punto  $\mathbf{p} \in \mathbb{R}^3$  si (y solo si)  $f_i(\mathbf{p}) > f_j(\mathbf{p}), \forall i \neq j$ . Además, si  $\mathbf{p}$  pertenece al interior de  $\Omega_i$ , no existe  $j (\neq i)$  tal que  $f_i(\mathbf{p}) = f_j(\mathbf{p})$ , lo que enfatiza que el componente escalar debe tener un máximo único en el interior de un régión.

Con base en el *VFI* se organiza el *VFC*  $\chi(F)$ , cuyos elementos contienen valores vectoriales:

$$\chi(F) = [\chi_1(F), \chi_2(F), \dots, \chi_n(F)] \quad (3.11)$$

Cada función indicatriz dentro de  $\chi_i(F)$  tiene un valor de 1 o 0. Por lo tanto, las regiones  $\Omega_i$  ( $i = 1, \dots, n$ ) se pueden definir mediante el *VFC*  $\chi(F)$  tal que:

$$\Omega_i = \{\mathbf{p} : \chi(F(\mathbf{p})) = \chi_i(f_i(\mathbf{p})), \mathbf{p} \in \mathbb{R}^3\} \quad (3.12)$$

Para fines de ilustración, considérese un caso de cuatro regiones ( $n = 4$ ). Se necesitan cuatro funciones implícitas, como se ilustra en la figura 3.4. En este caso, el vector de funciones indicatrices  $\chi(F)$  tiene cuatro valores vectoriales posibles  $\chi(F) =$

$\{[1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], [0, 0, 0, 1]\}$ , correspondientes a cada una de las regiones. Adicionalmente, la región  $\Omega_0 \notin \Omega$  que no pertenece a la estructura heterogénea se representa mediante  $[0, 0, 0, 0]$ .

La información de material se representa explícitamente, dentro del *VFI*, con una etiqueta  $l_i : 1 \leq l_i \leq m$ , donde  $m$  es el número total de materiales que se encuentran dentro de la estructura heterogénea; es decir, para cada región  $\Omega_i$  representada implícitamente mediante  $f_i$ , hay un material  $l_i$  del cual está compuesta.

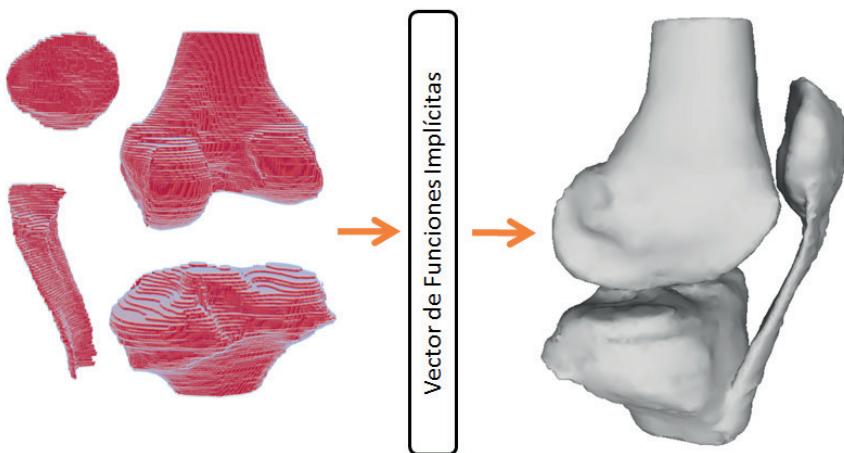


Figura 3.4: *VFI* en el modelado de una rodilla con cuatro regiones: el fémur, la tibia, la rótula y el ligamento rotular

Se han creado dos procedimientos para manipular el *VFI* y así obtener una representación implícita de la estructura heterogénea: *Agregar*( $f_i$ ) y *Quitar*( $f_i$ ). El orden en que se ejecutan estas operaciones determina la configuración final del modelo resultante; por ejemplo, la región que se procesa primero probablemente perderá la mayor parte de su geometría y no tendrá influencia en las otras regiones de la estructura heterogénea. Asimismo, la región que se procese de últimas no se verá afectada por otras.

Por último, se debe actualizar el vector de funciones indicatrices *VFC* para poder generar correctamente los modelos geométricos tanto de superficie como de volumen que serán utilizados finalmente en las diferentes aplicaciones (sección 3.3). Esto se logra mediante el procedimiento *Actualizar()*, que es invocado automáticamente cuando se utilice los procedimientos *Agregar*( $f_i$ ) y *Quitar*( $f_i$ ) del vector de funciones implícitas *VFI*.

### 3.3. Proceso de extracción de la geométrica

El modelo implícito resultante del proceso de reconstrucción implícita (subsección 2.2.3) puede ser visualizado directamente utilizando un motor de trazado de rayos como POV-Ray<sup>2</sup> o X-Seed (Barbour, 2001). Sin embargo, para obtener una representación fotorrealista de la estructura heterogénea o un modelo válido para aplicaciones en ingeniería es necesario exportar estos modelos como una malla tridimensional.

Una vez que las superficies implícitas  $f_i$  están ajustadas al conjunto de puntos discretos  $P_i$  y el vector de funciones implícitas  $VFI$  está disponible, se puede generar una malla de superficie  $M_S$  (subsección 3.3.1), o una malla de volumen  $M_V$  (subsección 3.3.2) tanto de cada una de las regiones que componen la estructura heterogénea como de la estructura heterogénea en general (algoritmo 3.2).

---

**Algoritmo 3.2** Mallado de la superficie y del volumen de la estructura heterogénea y sus regiones incidentes

---

Entrada: Vector de funciones implícitas  $VFI$ .

Salida: Modelo geométrico: malla de superficie  $M_S$  y malla de volumen  $M_V$ .

Para todas las  $f_i \in VFI$   $1 \leq i \leq n$

    Extraer malla de superficie:  $M_{S_i} \leftarrow \text{Superficie } (f_i)$ .

    Extraer malla de volumen:  $M_{V_i} \leftarrow \text{Volumen } (f_i)$ .

Fin Para

    Extraer malla de superficie de la estructura:  $M_S \leftarrow \text{Superficie } (VFI)$ .

    Extraer malla de volumen de la estructura:  $M_V \leftarrow \text{Volumen } (VFI)$ .

---

Para analizar las estructuras anatómicas heterogéneas se requiere que las mallas representen cada estructura con la calidad apropiada y ajustadas a sus límites. Para esto, es necesario ajustar el tamaño y el ángulo mínimo de los elementos de la malla, a fin de evitar tener triángulos y tetraedros de baja calidad tanto en la superficie como en el interior del modelo geométrico.

Hay dos tipos de triángulos de baja calidad: agudos y planos. El primero se refiere a un triángulo que tiene un ángulo cercano a cero y dos cerca a 90 grados. El segundo tiene dos ángulos cercanos a cero y uno cercano a 180 grados. Aunque los primeros pueden mejorar utilizando técnicas de colapso de bordes, que unirán los dos nodos opuestos al ángulo cercano a cero, no hay una solución adecuada para el segundo problema.

---

<sup>2</sup><http://www.povray.org/>

### 3.3.1. Malla de superficie

Una malla triangular es un tipo de malla poligonal que comprende un conjunto de triángulos que se conectan por sus lados o vértices comunes y aproximan la superficie de un objeto.

**Definición 2.** Una malla de superficie  $M_S = (V, T)$  es una malla triangular cerrada que rodea el interior de un objeto de interés  $\Omega$ , donde  $V = \{\mathbf{v}_i : 1 \leq i \leq n_V\}$  denota el conjunto de vértices, y  $T = \{t_k : 1 \leq k \leq n_T\}$  representa el conjunto de triángulos orientados (figuras 3.5 y 3.6).

Los triángulos  $t_k$  de la malla  $M_S$  cumplen:

- $\Omega = \bigcup_{k=1}^{n_t} t_k$ .
- Cada triángulo es cerrado y con interior no vacío.
- Para diferentes triángulos  $t_i$  y  $t_j$  la intersección de sus interiores es vacía.
- Para diferentes triángulos  $t_i$  y  $t_j$  su intersección es un lado común, un vértice común o es vacía.

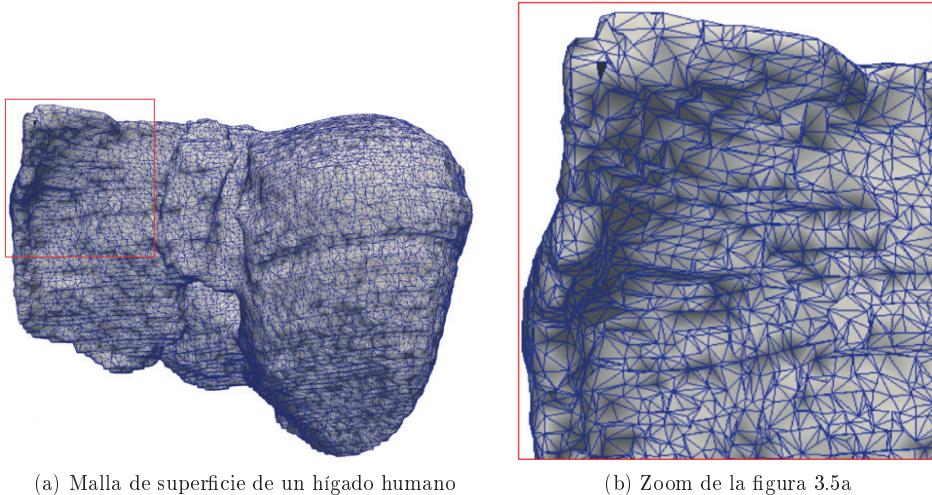
La mayor ventaja de las superficies implícitas es que la clasificación de un punto relativo a la superficie es trivial. Solo se tiene que mirar el signo de la función en el punto dado teniendo en cuenta las siguientes propiedades:

- $d < 0$  para puntos dentro de la superficie.
- $d = 0$  para puntos en la superficie.
- $d > 0$  para puntos por fuera de la superficie.

Para superficies dadas como el conjunto de nivel cero  $d = 0$  de una función implícita  $f(x, y, z)$ , se puede generar una malla  $M_S$  aproximada a la isosuperficie usando el algoritmo Marching Cubes (subsección 2.3.1). Para el caso de funciones indicatrices, el isovalor  $d$  puede establecerse como el promedio de las muestras de la función indicatriz reconstruida  $\chi$ , ponderado por el área de las muestras (Kazhdan *et al.*, 2006). Este método es realmente bueno para producir mallas triangulares de aspecto agradable que se pueden usar en diversas aplicaciones, como juegos, museos virtuales y otros. Sin embargo, las mallas producidas por el algoritmo Marching Cubes (Lorensen y Cline,

1987) pueden contener huecos y una gran cantidad de triángulos, muchos de ellos de baja calidad y con direcciones erróneas (figura 3.5).

La figura 3.5b muestra un rectángulo de color rojo que hace un acercamiento de una parte de la malla generada por el algoritmo Marching Cubes en la figura 3.5a.



(a) Malla de superficie de un hígado humano

(b) Zoom de la figura 3.5a

Figura 3.5: Malla de superficie generada por el algoritmo Marching Cubes, donde se evidencian huecos y triángulos agudos

Debido a lo anterior, en lugar de utilizar el algoritmo de Marching Cubes, se emplea el *framework* propuesto por Jamin *et al.* (2015), que calcula un conjunto de puntos de muestra en la superficie implícita y extrae una malla de superficie que interpola a partir de la triangulación de los puntos de muestra. La malla resultante se adapta a las características geométricas de las aplicaciones de ingeniería. La malla superficial solo contiene triángulos de calidad, de un tamaño uniforme o con un tamaño adaptado a la curvatura, es decir, pocos triángulos donde la superficie es plana y muchos triángulos donde la superficie tiene altas curvaturas (figura 3.6).

En muchas aplicaciones, poder saber el área de una región determinada o de la estructura heterogénea es una operación clave. El procedimiento para hallar el área de una malla de triángulos  $M_S$  se llama  $\text{Area}(\Omega_i)$ , la cual es calculada como:

$$\begin{aligned}
 Area(M_S) &= 1/2 \int_{\partial S} \mathbf{r} \times d\mathbf{l} \\
 &= 1/2 \sum_{i=1}^{n-2} (v_1, v_{i+1}) \times (v_1, v_{i+2})
 \end{aligned} \tag{3.13}$$

Donde  $\partial S$  es el límite de la región  $\Omega$  y  $d\mathbf{l}$  es el vector tangente diferencial del límite (Zhang y Chen, 2001). Por lo tanto, el área de superficie  $M_S$  es igual a la suma de las áreas de todos los triángulos que la conforman:

$$Area(\Omega_i) = \sum_{k=1}^{k=m} Area(triangulo_{i,k}) \tag{3.14}$$

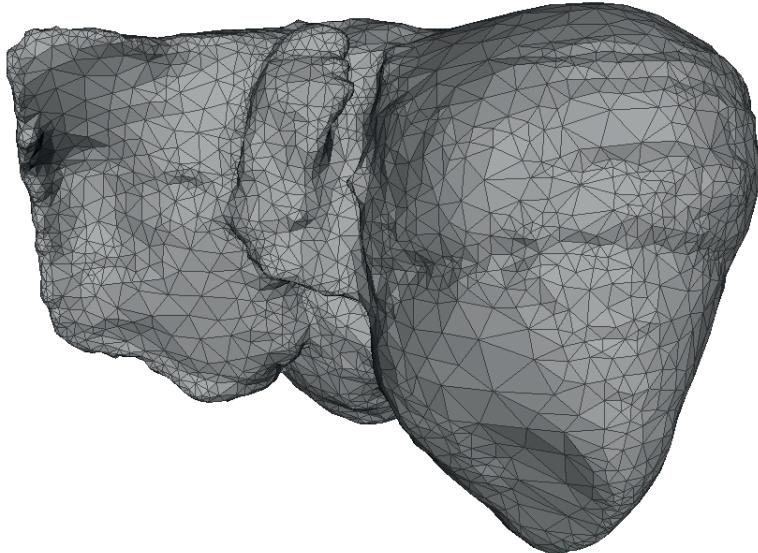


Figura 3.6: Malla de superficie generada por el método RAM

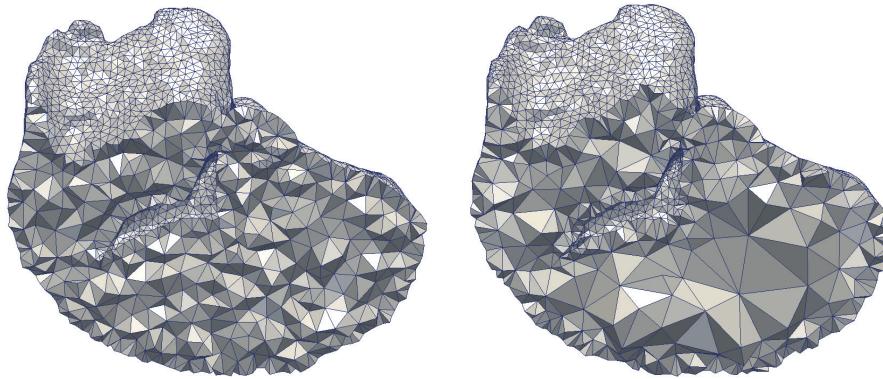
### 3.3.2. Malla volumétrica

Aunque en muchos campos de ingeniería las mallas de superficie  $M_S$  son una poderosa herramienta para estudiar las propiedades de los objetos, en otros más, como la fabricación de partes, la medicina, la realidad virtual y la ingeniería biomédica, estas no son suficientes, especialmente para aquellas aplicaciones en las que se necesita

información sobre el interior de los objetos y se requiere contar con mallas volumétricas  $M_V$ .

Las mallas volumétricas son una representación poligonal del volumen interior de un objeto. A diferencia de las mallas de superficie, estas discretizan la estructura interior del objeto mediante una colección de tetraedros.

**Definición 3.** Sea  $M_V = \{V, C\}$  una malla volumétrica que discretiza el interior de una región de interés  $\Omega$ , donde  $V = \{v_i : i = 1, 2, \dots, n\}$  denota el conjunto de vértices, y  $C = \{c_k : k = 1, 2, \dots, m\}$  representa el conjunto de tetraedros (figura 3.7).



(a) Corte de una malla volumétrica con tamaño de celda homogénea      (b) Corte de una malla volumétrica con tamaño de celda variable

Figura 3.7: Generación de mallas volumétricas

Existen diferentes algoritmos basados en grillas (Ju *et al.*, 2002; Anderson *et al.*, 2010) y en triangulaciones de Delaunay (Boltcheva *et al.*, 2009; Dey *et al.*, 2012; Jamin *et al.*, 2015) para generar mallas volumétricas. El primer paso para ello es introducir la descripción de la geometría. La longitud de las aristas para cada punto en el espacio debe especificarse como una función  $h(x, y, z)$ . Así, la longitud de la arista estándar se calcula de forma normalizada como en Jamin *et al.* (2015) y se define un rango de aceptación a partir del cual los triángulos fuera del rango se refinrarán o engrosarán.

En la figura 3.7a se muestra una malla volumétrica con un tamaño de celda homogénea, mientras que en la figura 3.7b se usa un tamaño de celda variable. Por lo general, se desea un tamaño pequeño cerca de la superficie de la región y un tamaño grande lejos de esta. Como se puede ver, se eliminan muchos elementos cuando se utiliza un método de mallado adaptativo de celda.

Para generar mallas volumétricas de estructuras heterogéneas  $\Omega$  se cuenta con una función  $A(\Omega)$  definida en todas partes en el dominio  $D$  y un valor arbitrario  $A_i = A(\Omega_i)$ , el cual se interpreta como el tamaño deseado de los elementos de la malla de volumen  $M_V$  en un punto  $\mathbf{p}_k$  de la región  $\Omega_i$ .

Dado que la geometría de la región se especifica mediante la función implícita que se encuentra en el *VFI*, entonces un valor de la función  $A_i = A(\Omega_i)$ , que describe la región  $\Omega_i$  en un punto  $\mathbf{p}_k$ , se puede usar como una medida de la distancia más cercana desde  $\mathbf{p}_k$  a la región  $\Omega_i$ :

$$d(\mathbf{p}_k) = d(F(\mathbf{p}_k)) \quad (3.15)$$

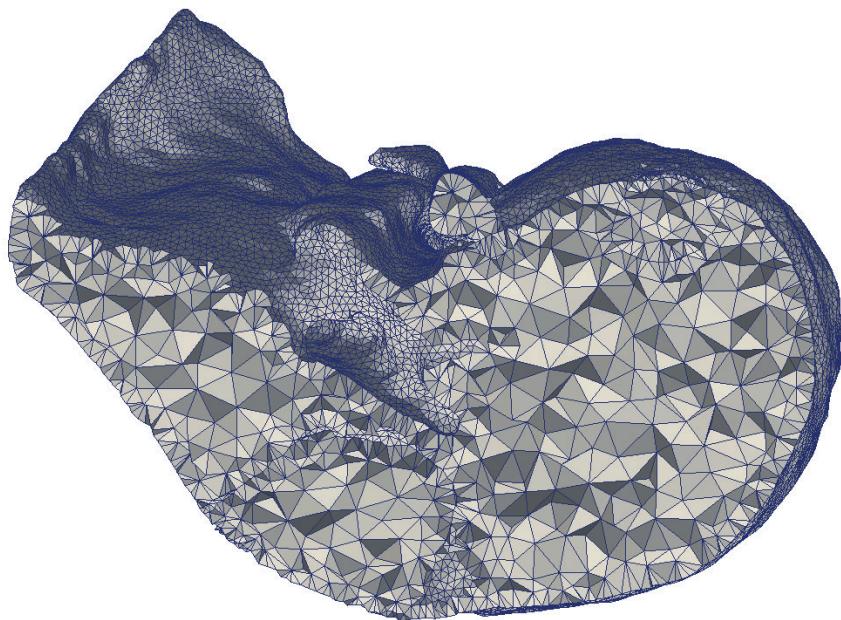
El atributo de tamaño de elemento  $A_i$  generado por la  $i^{th}$  región se define de manera que:

$$A_i(\mathbf{p}_k) = \begin{cases} h_{\min} & if F_i(\mathbf{p}_k) \leq 0 \\ \min(h_{\max}, (F_i(\mathbf{p}_k) * (c_i - 1) + h_{\min})/c_i) & if F_i(\mathbf{p}_k) > 0 \end{cases} \quad (3.16)$$

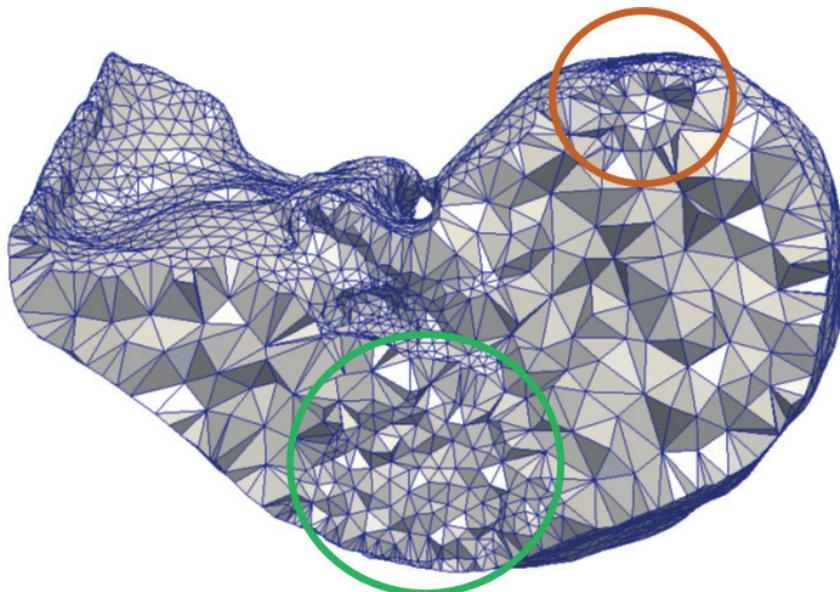
Donde  $h_{\min}$  y  $h_{\max}$  son los tamaños mínimos y máximos admisibles de los elementos,  $c_i$  es el coeficiente de la progresión  $c_i \geq l$  y  $F_i(\mathbf{p}_k)$  es la descripción funcional de la región  $\Omega_i$ . Esta fórmula proporciona la ley de progresión geométrica para el aumento del tamaño de los elementos de la malla.

En la figura 3.8a se muestra la malla volumétrica, con un tamaño de celda homogénea, de una estructura heterogénea compuesta de tres regiones que representan un hígado con dos tumores adentro, marcados con dos círculos de color verde y amarillo. En la figura 3.8b se observa la malla de la misma estructura heterogénea pero con un tamaño de celda variable. Se puede apreciar que cuando el tamaño de las celdas es variable, se perciben más fácilmente las diferentes regiones que conforman la estructura heterogénea, al igual que las interfaces que los separa.

En algunas ocasiones es importante calcular el volumen de una región particular o de la estructura heterogénea en general. Este procedimiento se llama *Volumen*( $\Omega_i$ ).



(a) Corte de una malla volumétrica con el mismo tamaño de celda para todas las regiones



(b) Corte de una malla volumétrica con un tamaño de celda diferente para cada región

Figura 3.8: Malla volumétrica de una estructura anatómica heterogénea compuesta por tres regiones

Sea  $C_k$  un tetraedro con vértices  $\{\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\} \in V$ , donde el vértice  $\mathbf{v}_0$  está ubicado en el origen. Las coordenadas de los vértices son  $\mathbf{v}_0 = (0, 0, 0)$ ,  $\mathbf{v}_1 = (x_1, y_1, z_1)$ ,  $\mathbf{v}_2 = (x_2, y_2, z_2)$  y  $\mathbf{v}_3 = (x_3, y_3, z_3)$ . Se define una transformación lineal  $T$  como:

$$T = \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{pmatrix} \quad (3.17)$$

Aplicando esta transformación al tetraedro  $c_k = \{\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ , se obtiene una unidad ortogonal del tetraedro  $W_k = \{\mathbf{v}'_0, \mathbf{v}'_1, \mathbf{v}'_2, \mathbf{v}'_3\}$ . Entonces, el volumen de  $C_k$  es:

$$Vol(C_k) = \int \int \int_{C_k} dv = \| T \| \int \int \int_{W_k} dv = \frac{\| T \|}{6} \quad (3.18)$$

Donde  $\| T \|$  es el jacobiano, que es igual al valor absoluto del determinante de la matriz  $T$ . Es decir:

$$Vol(C_k) = \frac{1}{6}(-x_{k3}y_{k2}z_{k1} + x_{k2}y_{k3}z_{k1} + x_{k3}y_{k1}z_{k3} - x_{k1}y_{k3}z_{k2} - x_{k2}y_{k2}z_{k3} + x_{k1}y_{k2}z_{k3}) \quad (3.19)$$

Entonces el volumen de una malla volumétrica  $M_V$  es la suma de todos los tetraedros que la conforman (Zhang y Chen, 2001):

$$Volumen(\Omega_i) = \sum_{k=1}^m C_{i,k} \quad (3.20)$$



# Experimentos

---

En este capítulo se lleva a cabo una serie de experimentos para probar la versatilidad del método RAM propuesto en el capítulo 3. En la sección 4.1 se detalla el entorno experimental creado para este fin. La sección 4.2 se dedica al modelado de las regiones que forman parte de las estructuras anatómicas, mientras que la sección 4.3 aborda el modelado de estructuras anatómicas heterogéneas con diferentes configuraciones.

## 4.1. Entorno experimental

Todos los experimentos presentados en este documento se realizaron en una única aplicación multiplataforma escrita en el lenguaje de programación C++ e implementada en un computador con un microprocesador Pentium-i7 de 3.40 GHz y 8 gigas de memoria RAM. Las herramientas y bibliotecas que proporcionan una funcionalidad de soporte pertenecen al grupo de tecnologías *software libre*, por lo que su aplicación está exenta de taras económicas y legales (Pavarino *et al.*, 2013). Las siguientes son las más importantes: G++, ITK (<http://www.itk.org/>) (Johnson *et al.*, 2015), VTK (<http://www.vtk.org/>) (Avila *et al.*, 2010), CGAL (<http://www.cgal.org/>) (Fabri y Pion, 2009), Open GL y Eigen. Finalmente, para la creación de la aplicación y su enlace con las diferentes herramientas y bibliotecas se utilizó CMake (<http://www.cmake.org/>) (Martin y Hoffman, 2010).

Para probar el método RAM se utilizaron cuatro conjuntos de imágenes médicas tomadas de rodilla, abdomen, columna y cerebro, respectivamente. En la tabla 4.1 se pueden visualizar las vistas axial, sagital y coronal de cada una de las imágenes TAC. El procedimiento *Extraer(RegionΩ)*, definido en la subsección 3.2.1, se aplicó a

las diferentes imágenes para obtener las regiones de interés que serán utilizadas en los experimentos descritos en las secciones 4.2 y 4.3.

Tabla 4.1: Vista axial, sagital y coronal de las imágenes TAC

Nombre	Vista axial	Vista sagital	Vista coronal
Rodilla			
Abdomen			
Columna			
Cerebro			

## 4.2. Modelado de regiones anatómicas

En esta sección se presenta el modelado de las regiones que luego formarán parte de diferentes las estructuras heterogéneas. En la subsección 4.2.1 se presenta el modelado de regiones que poseen una forma redondeada, como es el caso de los órganos y algunos

huesos humanos. Luego, en la subsección 4.2.2 se muestra el uso del método RAM para modelar regiones con ramificaciones y huecos, que son geometrías más complejas. Para llevar a cabo los experimentos de esta sección se segmentaron de cada imagen TAC dos regiones de interés, como se puede apreciar en las figuras 4.1 y 4.6.

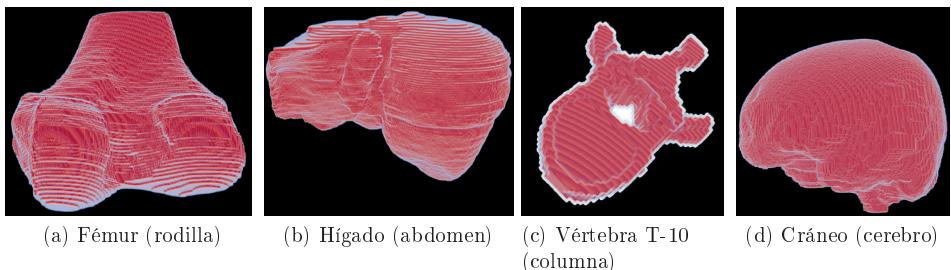


Figura 4.1: Segmentación de regiones simples

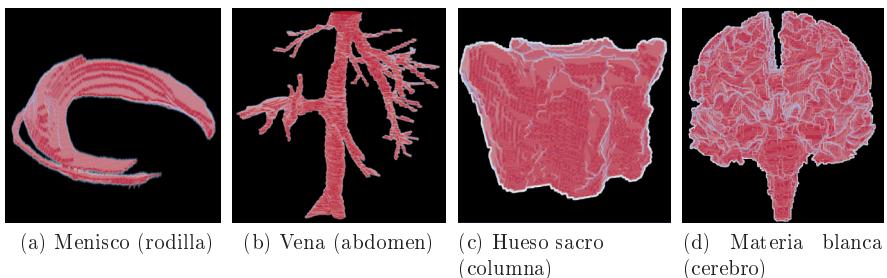


Figura 4.2: Segmentación de regiones complejas

#### 4.2.1. Regiones anatómicas simples

En esta subsección se presenta el modelado de cuatro estructuras anatómicas que se caracterizan por tener una forma redondeada. En la figura 4.1 se visualiza la segmentación de un fémur, un hígado, una vértebra y un cráneo. A partir de cada una de estas segmentaciones se procedió a generar una nube de puntos y a reconstruir implícitamente la superficie de la región en cuestión.

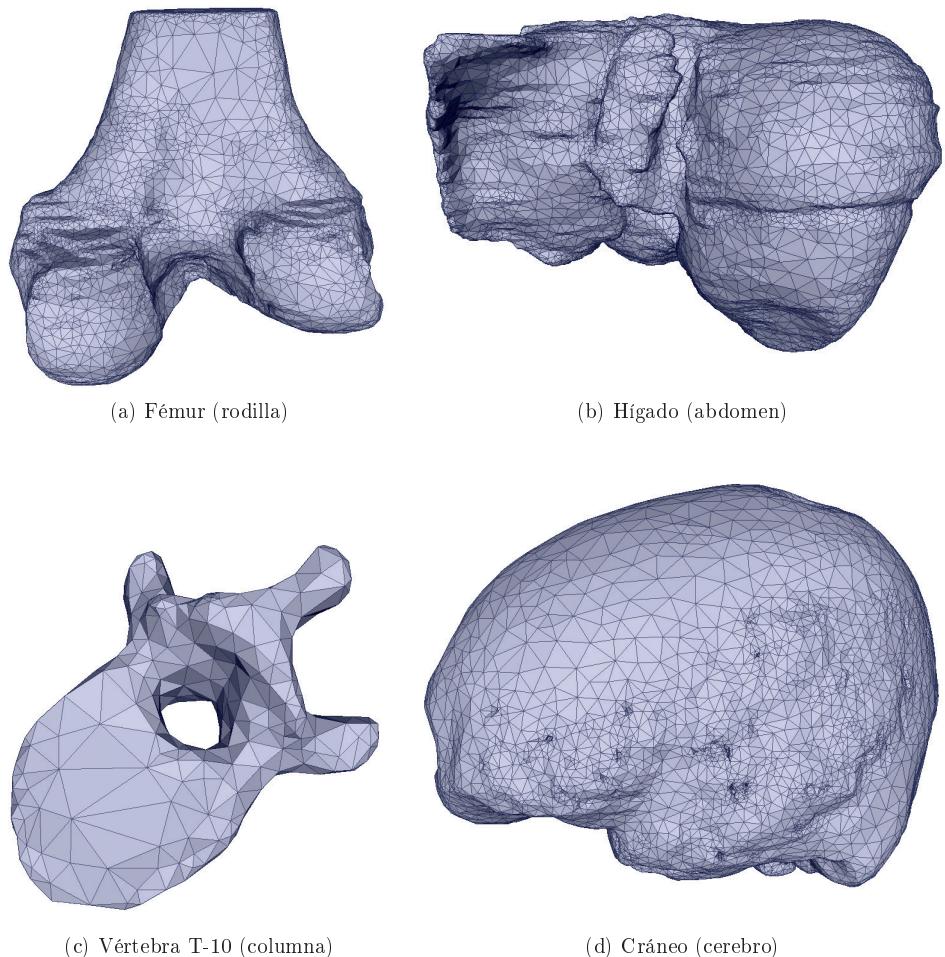


Figura 4.3: Mallas de superficie ajustadas a las funciones implícitas

La figura 4.3 muestra las mallas de la superficie del fémur, el hígado, la vértebra torácica 10 y el cráneo, obtenidas a partir de sus respectivas funciones implícitas. Se puede ver que el efecto de escalón, que aparece en la figura 4.1, se ha eliminado y en su lugar todas las regiones se presentan con superficies suaves y redondeadas.

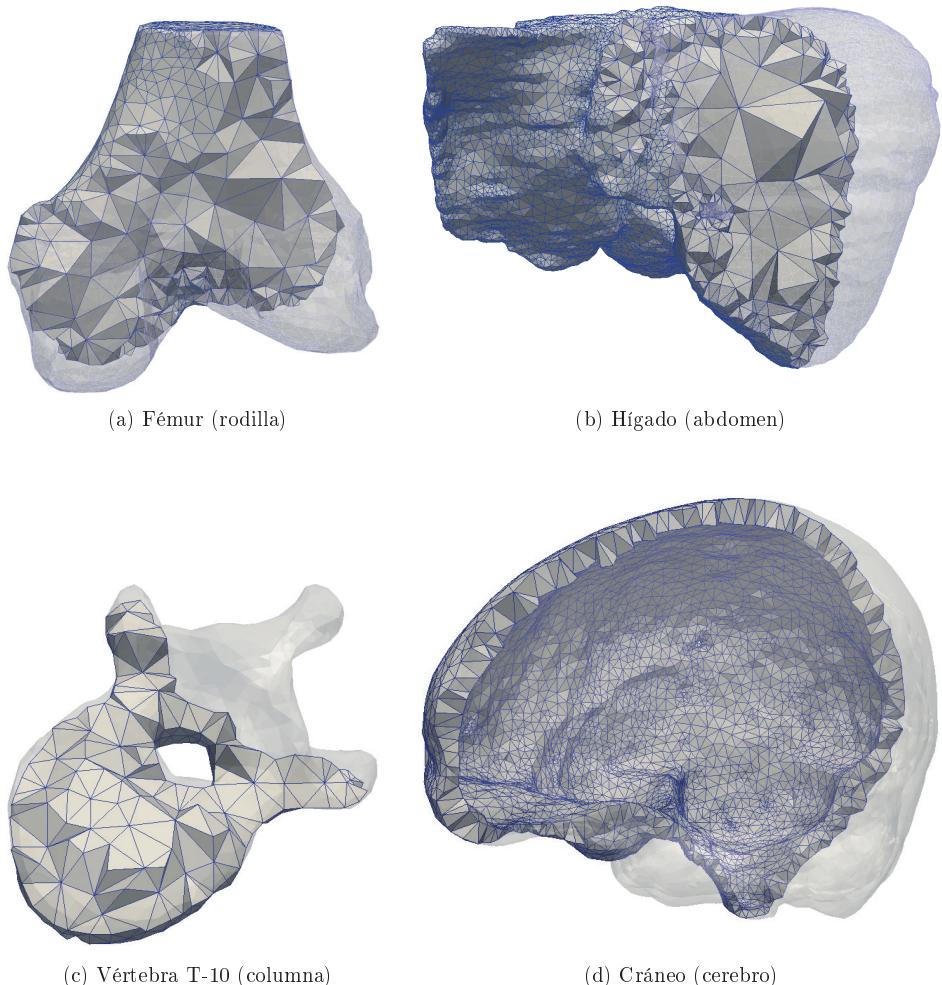
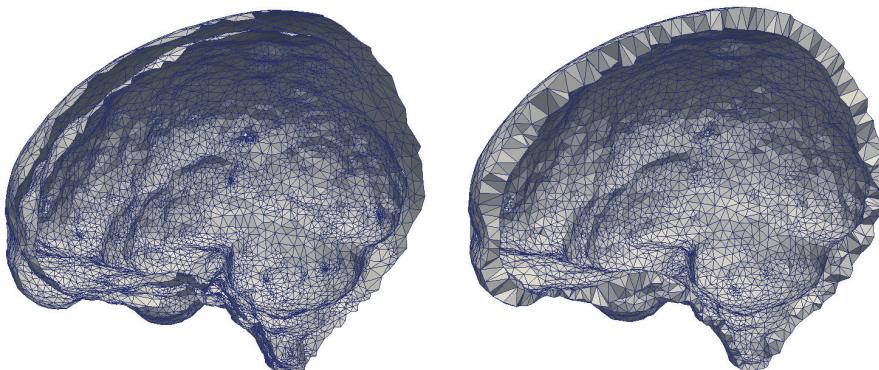


Figura 4.4: Corte vertical translúcido de las mallas volumétricas

Una vez se obtienen las mallas de superficie, el método RAM también puede generar mallas de volumen. En la figura 4.4 se muestra un corte translúcido que permite apreciar la estructura interna de las diferentes mallas de volumen obtenidas a partir de las regiones de interés de la figura 4.1. Cada malla volumétrica se genera una vez se obtiene la malla de superficie de la región de interés. Esta estrategia permite refinar los tetraedros dentro del volumen siempre que el circuncentro no provoque la eliminación de uno de los triángulos en la superficie.

La figura 4.5 permite ver más detalladamente la diferencia entre una malla de superficie y una malla volumétrica. La figura 4.5a muestra un corte de la malla de superficie del cráneo. Se ve cómo en este caso se generaron dos mallas: una externa y una interna, debido a que el cráneo es hueco en su interior y el método RAM puede modelar esta condición. La figura 4.5b muestra un corte de la malla de volumen, donde se aprecia claramente que el volumen del cráneo se encuentra entre las dos superficies que lo limitan y que impiden que se inserten tetraedros en el espacio interior (hueco) de este modelo.



(a) Corte de la malla de superficie que deja en evidencia la existencia de una malla interna      (b) Corte de la malla volumétrica que muestra la característica hueca del cráneo

Figura 4.5: Diferencia entre un modelo de superficie y un modelo de volumen

#### 4.2.2. Regiones anatómicas complejas

En esta subsección se demuestra la capacidad del método RAM para generar modelos tridimensionales a partir de regiones de interés más complejas que aquellas determinadas en la subsección 4.2.1. En la figura 4.2 se pueden apreciar las segmentaciones del menisco lateral que da soporte a la rodilla, del sistema venoso que irriga al hígado y riñones, del hueso sacro que conecta el coxis con la vértebra L5 y la pelvis, y de la materia blanca localizada en el cerebro.

En la figura 4.6 se pueden ver las mallas de superficie generadas a partir de las funciones implícitas que representan las regiones de interés de la figura 4.2. La complejidad en el caso de regiones anatómicas como el menisco lateral radica en el modelado de los diferentes segmentos de la región, que se unen en ciertas partes y dejan separaciones

en otras. El método RAM modela todos los componentes dejando las correspondientes separaciones, lo que permite calcular adecuadamente tanto el área como el volumen de este tipo de estructuras.

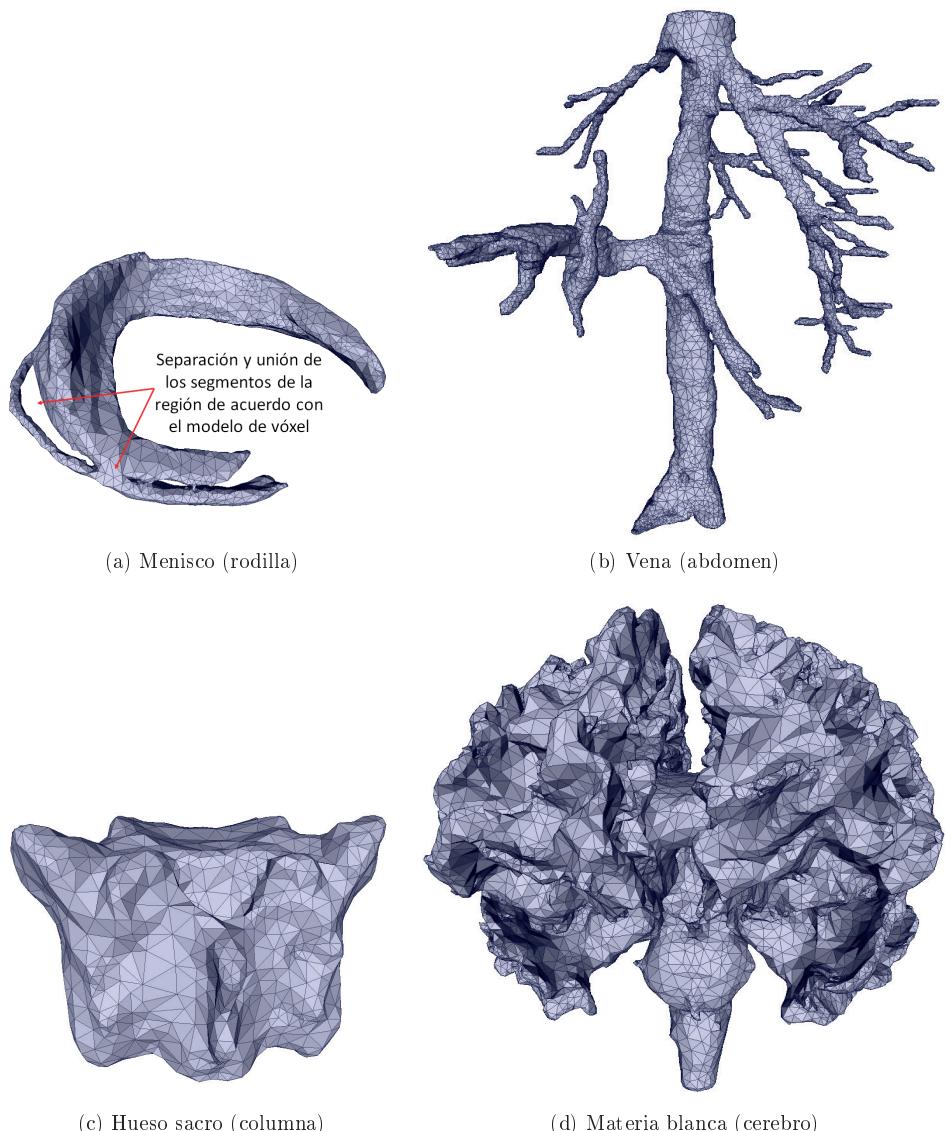


Figura 4.6: Mallas de superficie de regiones complejas

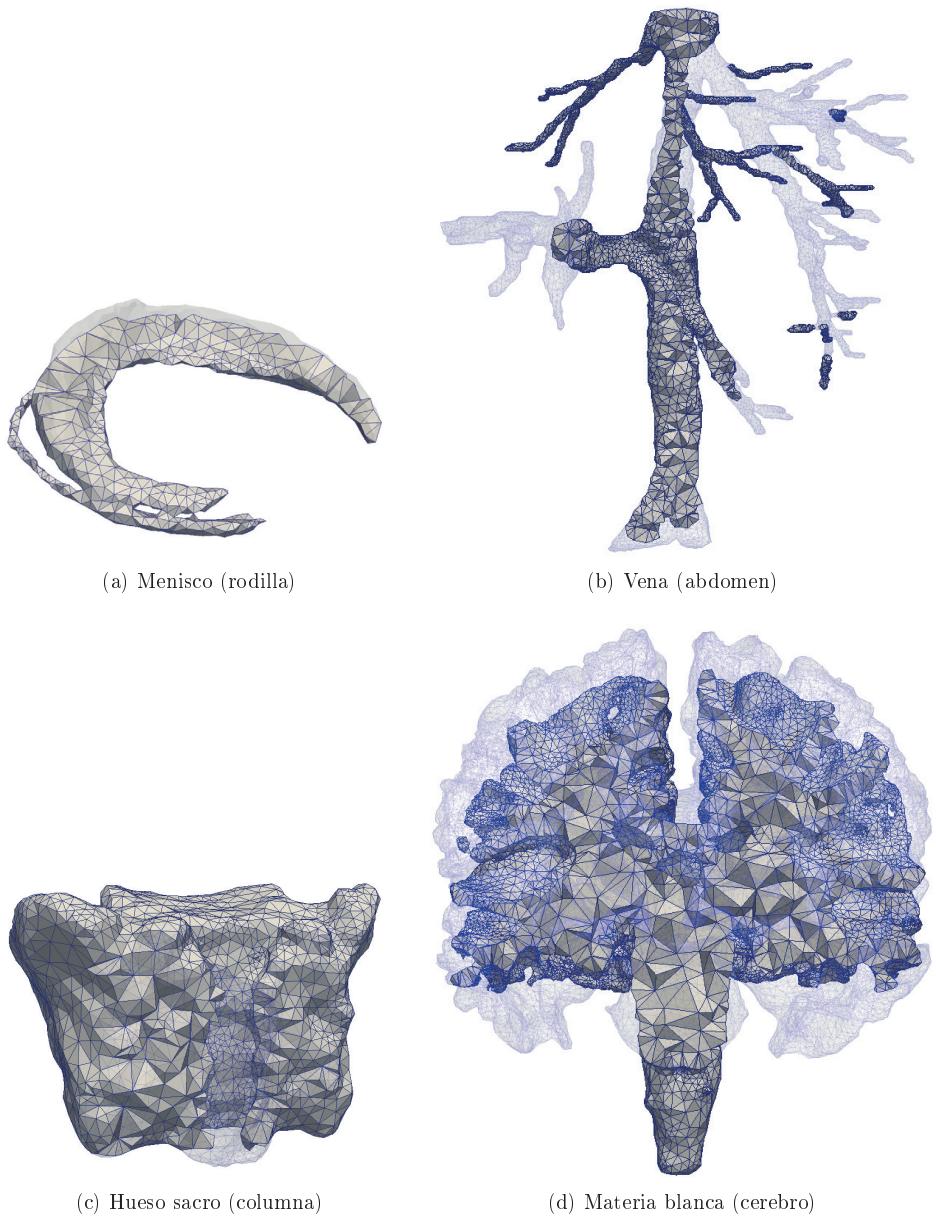


Figura 4.7: Mallas volumétricas de regiones complejas

Las venas, las arterias y, en general, las estructuras tubulares son otro ejemplo de estructuras anatómicas complejas. En este caso, las cavidades tubulares de las regiones

de interés en las segmentaciones son representadas por muy pocos vértices, lo que podría ocasionar que algunas ramas sean omitidas al momento de hacer la reconstrucción de la superficie. El método RAM puede lidiar con este tipo de estructuras haciendo un correcto modelado de todas las ramificaciones. Esto permite hacer cálculos más exactos del área y el volumen de la estructura bajo estudio.

Otras estructuras presentan de forma natural, o por algún tipo de lesión, cavidades o huecos que atraviesan completamente la región anatómica, como es el caso de las vértebras, el hueso sacro y la pelvis. Se busca que al modelar estas estructuras no sean omitidas estas características. El método RAM, al igual que en el modelado del menisco lateral, puede modelar tales peculiaridades.

Finalmente, existen estructuras anatómicas con superficies muy irregulares, como es el caso de la materia blanca y la materia gris localizadas en el cerebro. Estas estructuras están conformadas por picos pronunciados y valles profundos. El método RAM detecta estas características y genera modelos tridimensionales que representan tales estructuras con bastante realismo.

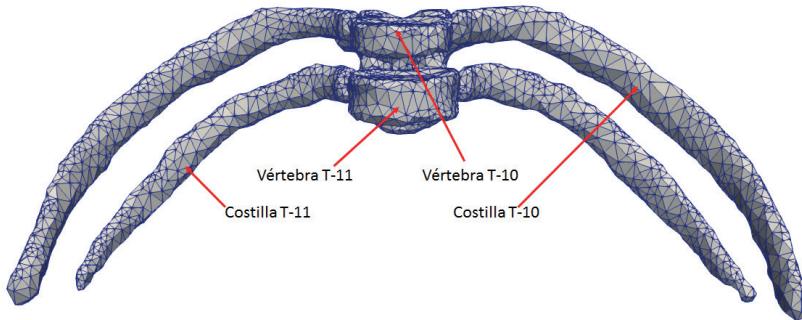
En la figura 4.7 se puede apreciar la conformación interna de las mallas de volumen de las diferentes regiones complejas que el método RAM es capaz de modelar. Aunque los tetraedros al interior pueden tener diferentes tamaños, esto no afecta la malla en la superficie ya que esta se adapta correctamente a la superficie reconstruida por la función implícita.

### 4.3. Modelado de estructuras anatómicas heterogéneas

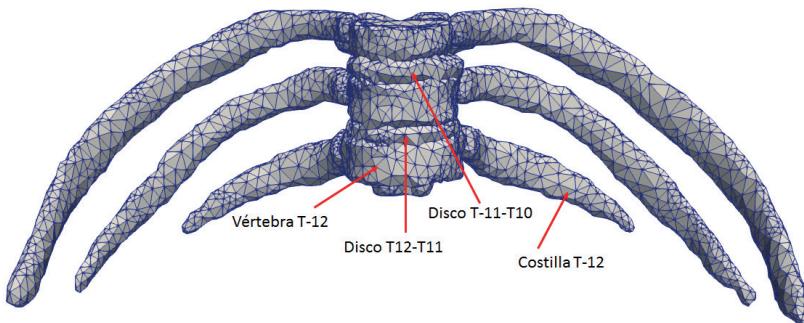
Los experimentos de la sección 4.2 han demostrado la capacidad del método RAM para generar modelos geométricos 3D de diferentes regiones anatómicas simples o complejas. En esta sección se presenta el modelado de estructuras anatómicas heterogéneas con diferentes configuraciones. Primero, se considera el caso de modelado de tres estructuras anatómicas heterogéneas compuestas de regiones colindantes (subsección 4.3.1). Luego se muestra el modelado de dos estructuras anatómicas heterogéneas compuestas con regiones anidadas (subsección 4.3.2). Finalmente, se presenta un caso donde se modela una estructura anatómica heterogénea con una configuración más compleja combinando los dos escenarios anteriores (subsección 4.3.3).

#### 4.3.1. Estructuras heterogéneas con regiones colindantes

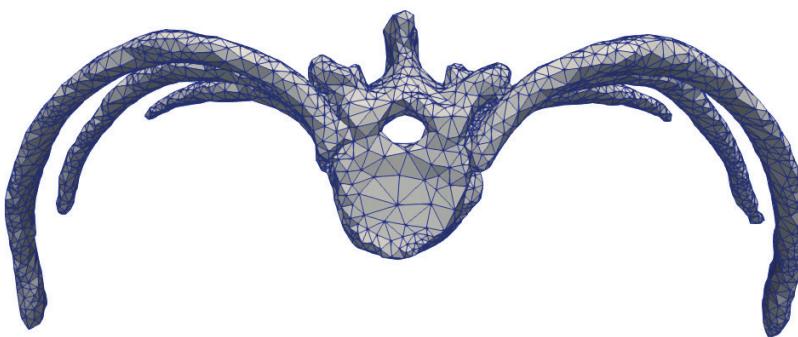
En esta subsección se modelan tres estructuras heterogéneas: la zona torácica de una columna vertebral (figura 4.8), la zona lumbar de la misma columna vertebral (figura 4.9) y una rodilla (figura 4.10).



(a) Modelo *Columna-a* formado por cuatro regiones



(b) Modelo *Columna-b* compuesto por ocho regiones

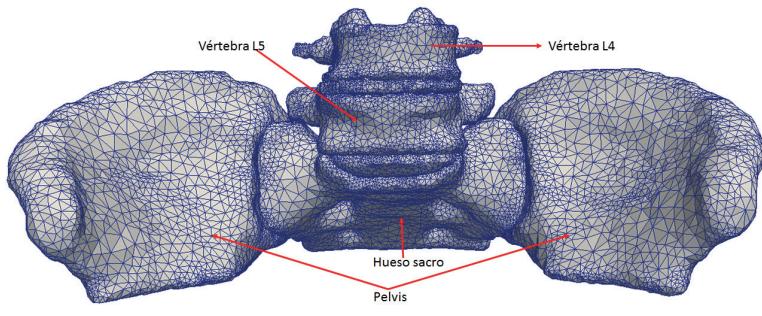


(c) Vista superior del modelo de la figura 4.8b

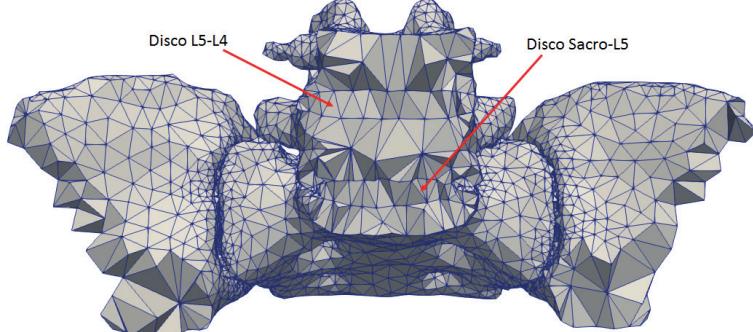
Figura 4.8: Modelado de dos estructuras anatómicas heterogéneas de la zona torácica

Para el modelado de la zona torácica de la columna vertebral se segmentaron ocho regiones de interés: las costillas T10, T11 y T12, las vértebras T10, T11 y T12 y los discos intervertebrales T11-T10 y T12-T11. A partir de cada región de interés se ha reconstruido una función implícita que ha sido agregada a un vector de funciones implícitas *VIF* (subsección 3.2.4).

El vector *VFI* está compuesto por  $F(\mathbf{p}) = [f_1(\mathbf{p}), \dots, f_8(\mathbf{p})]$ , donde  $f_1(\mathbf{p})$ ,  $f_2(\mathbf{p})$  y  $f_3(\mathbf{p})$  almacenan las funciones implícitas de las costillas,  $f_4(\mathbf{p})$ ,  $f_5(\mathbf{p})$  y  $f_6(\mathbf{p})$  las funciones implícitas de las vértebras y  $f_7(\mathbf{p})$  y  $f_8(\mathbf{p})$  las funciones implícitas de los discos intervertebrales. La figura 4.8a muestra el modelo *Columna-a* formado por cuatro regiones: las vértebras T10 y T11 y las costillas T10 y T11. En la figura 4.8b se puede apreciar el modelo *Columna-b* con ocho regiones: las vértebras T10, T11 y T12, las costillas T10, T11 y T12, y los discos intervertebrales T12-T11 y T11-T10. La figura 4.8b muestra una vista superior de la malla volumétrica del modelo *Columna-b*.

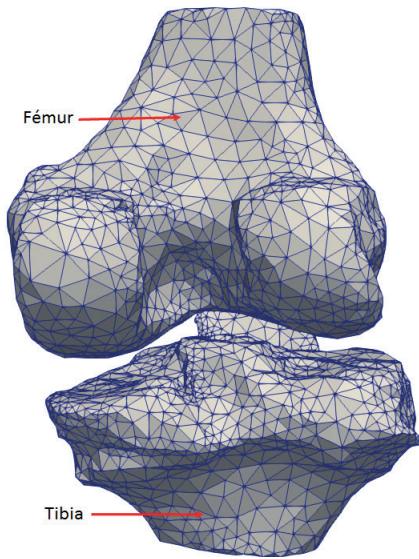


(a) Vista anterior del modelo *Pelvis* formado por seis regiones

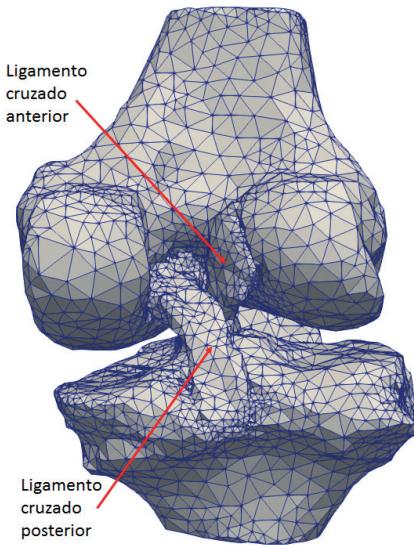


(b) Corte coronal del modelo *Pelvis* que muestra la ubicación de los discos intervertebrales

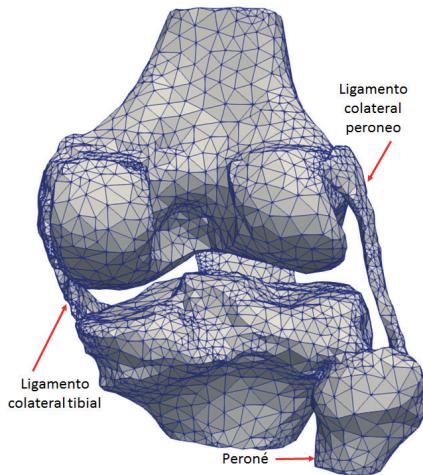
Figura 4.9: Modelado de la estructura heterogénea *Pelvis*



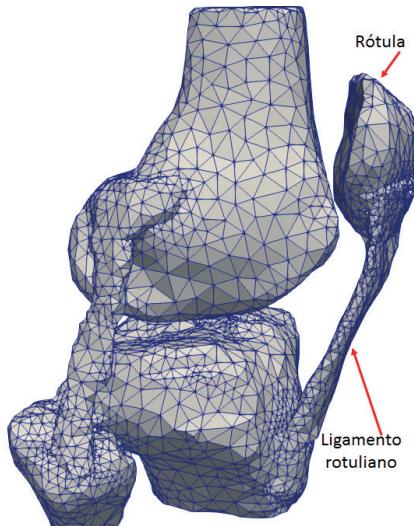
(a) Modelo *Rodilla-a* compuesto por cuatro regiones



(b) Estructura heterogénea *Rodilla-b* modelada a partir de la figura 4.10a, más los ligamentos cruzados anterior y posterior



(c) Modelo *Rodilla-c* formado a partir de la figura 4.10a, más la tibia y los ligamentos colaterales peroneo y tibial



(d) Vista lateral del modelo *Rodilla-c*

Figura 4.10: Estructuras anatómicas heterogéneas modeladas a partir de regiones de la rodilla

Para modelar la zona lumbar de la columna vertebral se segmentaron seis regiones de interés: la pelvis, el hueso sacro, las vértebras L5 y L4, y los discos intervertebrales sacro-L5 y L5-L4. Las funciones implícitas que representan las diferentes regiones se ingresan en el *VFI*, a partir del cual se obtuvo el modelo presentado en la figura 4.9.

La figura 4.9a muestra la vista anterior, en tanto la figura 4.9b presenta la vista posterior del modelo de la región lumbar de la columna vertebral, donde se puede observar cómo los diferentes elementos de la malla de volumen se adaptan a cada región para generar una sola estructura.

Para modelar la rodilla se segmentaron nueve regiones de interés: el fémur, la tibia, la rótula, el ligamento patelar, el peroné, el ligamento cruzado anterior, el ligamento cruzado posterior, el ligamento colateral fibular y el ligamento colateral tibial. La figura 4.10 muestra las mallas de volumen que corresponden a diferentes configuraciones de la rodilla hechas a partir del vector *F*.

La figura 4.10a muestra una vista frontal de una configuración inicial compuesta por el fémur, la tibia, la rótula y el ligamento rotular. En la figura 4.10b se puede apreciar una configuración donde se han agregado al modelo inicial los dos ligamentos cruzados: el anterior y el posterior, mientras que en la figura 4.10c se ve la configuración inicial con el peroné y dos ligamentos colaterales: el femoral y el peroneo. La figura 4.10d ofrece una vista lateral de esta última configuración.

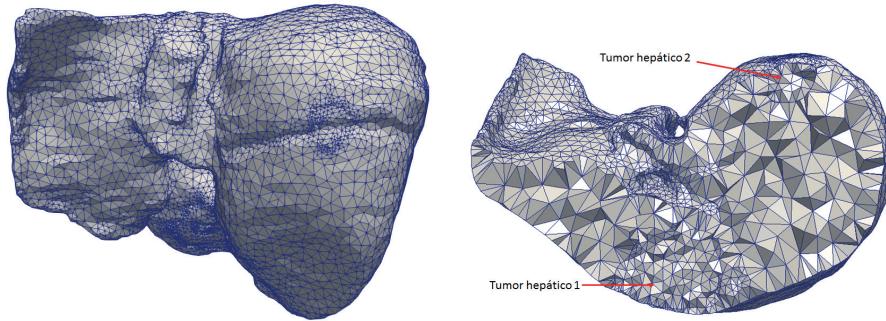
### 4.3.2. Estructuras heterogéneas con regiones anidadas

En esta subsección se modelan dos estructuras heterogéneas cuyas regiones se encuentran anidadas: un hígado con dos tumores dentro (figura 4.11) y una estructura cerebral con cuatro regiones (figura 4.12).

Para modelar la primera estructura se segmentaron del conjunto de datos del abdomen tres regiones de interés: el hígado y dos tumores hepáticos. El vector *VFI* está compuesto por  $F(\mathbf{p}) = [f_1(\mathbf{p}), f_2(\mathbf{p}), f_3(\mathbf{p})]$ , donde  $f_1(\mathbf{p})$  almacena la función implícita del hígado y  $f_2(\mathbf{p})$  y  $f_3(\mathbf{p})$  las funciones implícitas de los tumores hepáticos. A partir de este vector se obtiene la estructura de la figura 4.11a.

Para apreciar la ubicación de los tumores dentro del hígado se utilizó el método RAM, con un tamaño de celda diferente para cada región. La figura 4.11a muestra la vista anterior de la malla de volumen, donde se puede apreciar la discontinuidad en la malla de superficie debido a que los tumores hepáticos están ubicados muy cerca de la superficie del hígado.

Cuando se genera la malla de volumen, el método RAM calcula la intersección de las diferentes regiones para poder asignar a cada una el tamaño de celda correspondiente (subsección 3.3.2). Esto hace que no solo los elementos del volumen sean de tamaño diferente, sino que los elementos de la superficie de cada región también lo sean (figura 4.11b).



(a) Vista anterior de la malla del modelo *Hígado-tumores* (b) Corte transversal de la malla volumétrica de la figura 4.11a

Figura 4.11: Modelado de una estructura anatómica heterogénea compuesta por tres regiones: un hígado y dos tumores hepáticos

La segunda estructura anatómica por modelar es una estructura cerebral. Para esto se segmentaron del conjunto de datos *Cerebro* el cráneo, la materia blanca, la materia gris y el cerebelo. La figura 4.12 muestra la evolución en el modelado de esta estructura.

En la figura 4.12a se puede apreciar la estructura con dos regiones de interés: la materia blanca y el cerebelo. En la figura 4.12b se observa la misma estructura, incluyendo la materia blanca. Finalmente, en la figura 4.12c se puede apreciar la estructura a la que se le adicionó el cráneo. En la figura 4.12d se puede observar el interior de la estructura cerebral. En color rosado aparece la región de la materia gris que separa la materia blanca y el cerebro. En este caso se utilizó el método RAM, con tamaño de celda igual para todas las regiones que componen la estructura anatómica.

Los dos experimentos anteriores permiten concluir que para el caso de estructuras heterogéneas con regiones anidadas la geometría de la superficie de la estructura guarda correspondencia con la geometría de la región que anida a las otras.

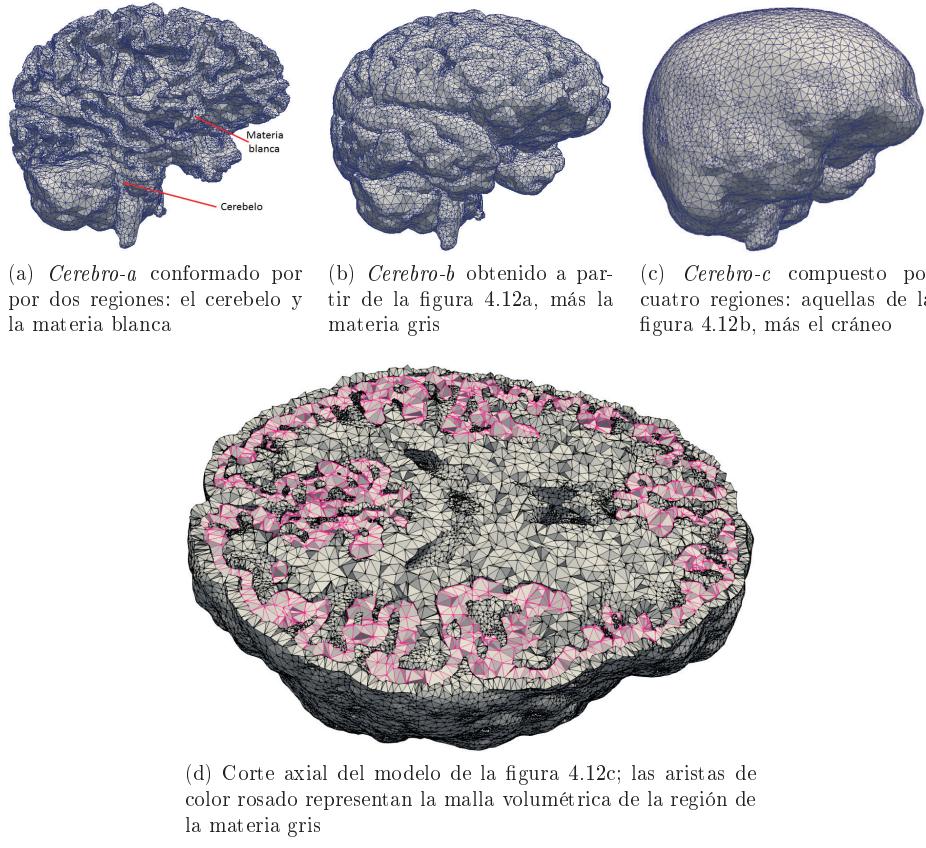


Figura 4.12: Diferentes estructuras anatómicas heterogéneas modeladas a partir de regiones del cerebro

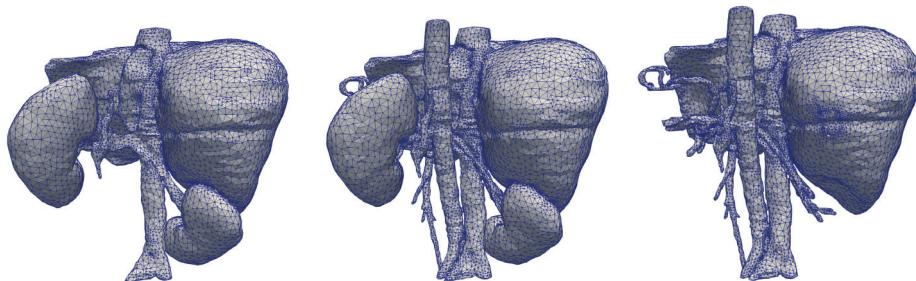
### 4.3.3. Estructuras heterogéneas combinadas

En este experimento se obtiene el modelo de una estructura heterogénea más compleja que combina una estructura heterogénea con regiones colindantes y una estructura heterogénea con regiones anidadas. Para ello se segmentaron siete regiones de interés a partir de la fuente de datos del abdomen: el hígado, dos tumores hepáticos, el riñón izquierdo, el riñón derecho, la vena y la arteria que interconectan el hígado y los riñones.

Con cada región de interés se reconstruyó una función implícita que se agregó a un vector  $VFI$   $F(\mathbf{p}) = [f_1(\mathbf{p}), f_2(\mathbf{p}), f_3(\mathbf{p}), f_4(\mathbf{p}), f_5(\mathbf{p}), f_6(\mathbf{p}), f_7(\mathbf{p})]$ , donde  $f_1(\mathbf{p})$  almacena la función implícita de la vena,  $f_2(\mathbf{p})$  y  $f_3(\mathbf{p})$  las funciones implícitas de los dos tumores,

$f_4(\mathbf{p})$  y  $f_5(\mathbf{p})$  las funciones implícitas de los dos riñones,  $f_6(\mathbf{p})$  la del hígado y  $f_7(\mathbf{p})$  almacena la función implícita de la arteria.

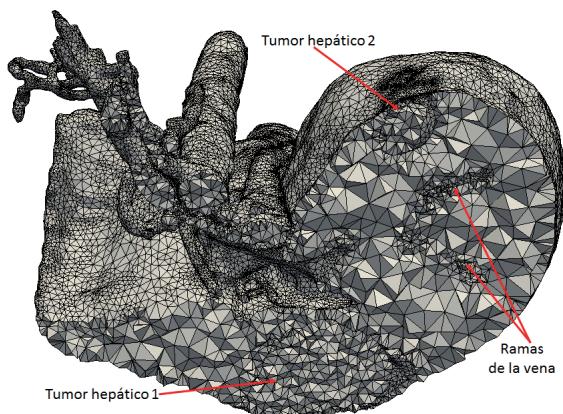
La figura 4.13 evidencia el orden en que se modeló la estructura anatómica. Primero, en la figura 4.13a se crea un modelo de una estructura heterogénea con regiones colindantes compuesta por la vena, el hígado y los dos riñones. Luego, en la figura 4.13b se agrega la arteria. En la figura 4.13c se tiene una configuración donde se eliminan los dos riñones y se adicionan los dos tumores hepáticos. En la figura 4.13d se puede ver el interior de la estructura heterogénea de la figura 4.13c. Como el tamaño de los tumores hepáticos, al igual que el de la vena y arteria, es más pequeño que el tamaño del hígado, la geometría de la estructura heterogénea refleja este aspecto.



(a) *Abdomen-a* formado por cuatro regiones colindantes

(b) *Abdomen-b* compuesto por cinco regiones colindantes

(c) *Abdomen-c* con cinco regiones: tres colindantes y dos anidadas



(d) Corte axial de la figura 4.13c; se muestra con las aristas en color negro para mejor visualización

Figura 4.13: Estructuras anatómicas heterogéneas modeladas a partir de regiones del abdomen

# Análisis de resultados

---

En este capítulo se analizan los datos obtenidos del proceso de modelado geométrico de las diferentes regiones y estructuras presentada en el capítulo 4. Tres criterios se han tomado para realizar este análisis: tiempo computacional relacionado con la ejecución de los diferentes procesos (sección 5.1), precisión en cuanto a la conservación del área y del volumen de las regiones y estructuras anatómicas (sección 5.2) y calidad geométrica de los modelos (sección 5.3).

## 5.1. Tiempo computacional

Una medida importante de eficiencia de los métodos y algoritmos para el modelado 3D de estructuras anatómicas heterogéneas es el tiempo computacional transcurrido desde el comienzo del proceso de modelado hasta que se obtiene el modelo geométrico final. La unidad de tiempo utilizada en esta sección es *segundos*.

Tabla 5.1: Tiempo para el modelado de las regiones segmentadas del cerebro

Regiones del cerebro	Puntos de entrada	Reconstrucción implícita	Mallado superficie	Mallado volumen	Total tiempo
Cerebelo	47.542	2,43	2,85	2,29	7,57
Cráneo	244.560	17,95	8,62	11,18	37,75
Materia blanca	287.520	16,93	9,28	35,91	62,12
Materia gris	540.360	31,19	46,95	45,95	124,09

En la tabla 5.1 se listan las regiones segmentadas de la fuentes de datos del cerebro, los puntos de entrada de cada región, los tiempos para la reconstrucción implícita de la

superficie y los tiempos para la extracción de la geométrica mediante el mallado de esta y del volumen. En la última columna se muestra el tiempo total, en segundos, requerido para el modelado de cada región de interés. Las regiones en cada tabla están ordenadas de menor a mayor de acuerdo con la cantidad de puntos de entrada.

Algunas regiones del cerebro, como la materia blanca y la materia gris, aunque no tienen un gran volumen en comparación con un órgano como el hígado (sección 5.2), sí tienen formas muy complejas con pliegues y ondulaciones que obligan a extraer un gran número de puntos para poder representarlas. La tabla 5.1 muestra también que los puntos de entrada para estas regiones son 287.520 y 540.360, respectivamente. Esto hace que la extracción de la geometría tanto de la superficie como del volumen a partir de la función implícita sea un proceso que consuma la mayoría del tiempo del modelado (figura 5.1). Este tiempo tiende a disminuir cuanto menos compleja es la forma de la región por modelar. Para el caso del cráneo, el mallado de la superficie tarda 8,62 segundos, el mallado del volumen 11,18 segundos, mientras que la reconstrucción implícita dura 17,95 segundos.

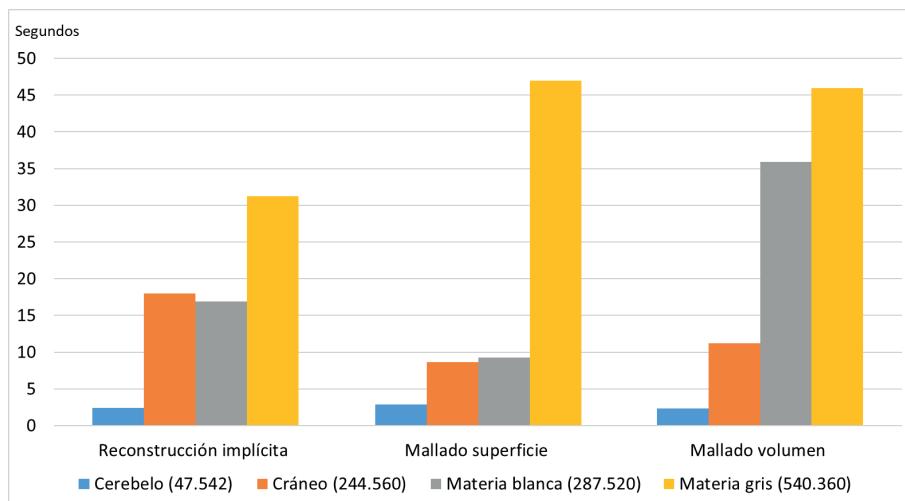


Figura 5.1: Tiempo para el modelado de las regiones del cerebro

La figura 5.1 muestra el tiempo para el modelado de las regiones del cerebro ordenadas por el número de puntos de entrada (tabla 5.1), que aparece entre paréntesis. Este mismo esquema se aplica en todas las figuras de esta sección. En la tabla 5.2 se puede observar que el tiempo asociado a la reconstrucción implícita de las regiones está relacionado directamente con el tamaño de la nube de puntos de entrada. Es así

como la reconstrucción implícita del Tumor2 (tabla 5.2), representado por 5.228 puntos, es menor que la del Tumor1, con un número de puntos de 29.510, y la de este sea menor que la del riñón derecho, que tiene 38.582 puntos, y así sucesivamente.

Tabla 5.2: Tiempo para el modelado de las regiones segmentadas del abdomen

Abdomen	Puntos de entrada	Reconstrucción implícita	Mallado superficie	Mallado volumen	Total tiempo
Tumor2	5.228	0,35	0,13	0,19	0,67
Tumor1	29.510	1,84	0,72	1,59	4,15
Riñón derecho	38.582	2,47	0,78	3,92	7,17
Riñón izquierdo	46.822	3,11	2,57	4,69	10,37
Arteria	57.946	4,27	2,79	6,64	13,7
Vena	93.714	7,07	6,31	13,87	27,25
Hígado	221.402	19,32	4,45	4,73	24,5

Se puede observar en la figura 5.2 que el proceso de extracción de la geometría es bastante rápido para regiones que tienen una forma suave y cóncava, como la de los riñones e hígado.

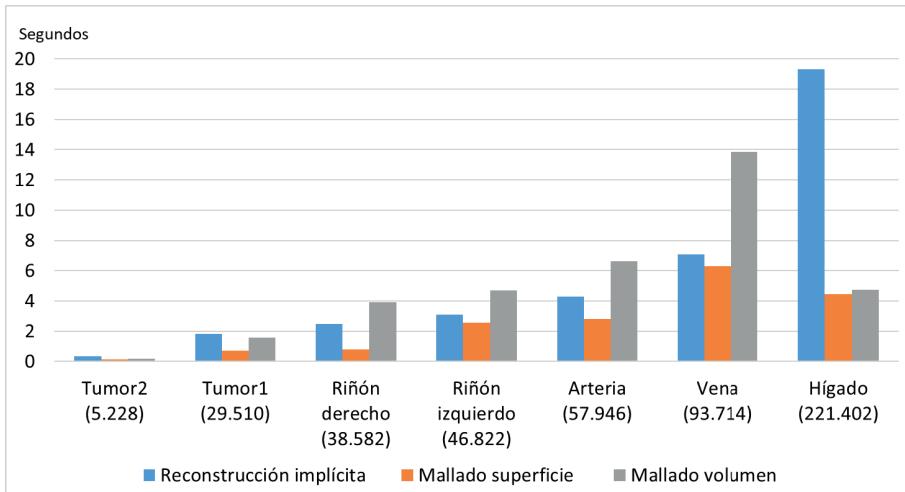


Figura 5.2: Tiempo para el modelado de las regiones del abdomen

Sin embargo, cuando la forma se torna compleja, como en el caso de la vena y la arteria (figura 5.2), este tiempo aumenta y generalmente es mayor que el necesario para realizar la reconstrucción implícita.

Por otro lado, se puede observar en la tabla 5.3 que el tiempo para extraer la malla de superficie no depende del tamaño de la nube de puntos de entrada, sino de la forma que tiene la función implícita reconstruida.

Tabla 5.3: Tiempo para el modelado de las regiones segmentadas de la rodilla

Rodilla	Puntos de entrada	Reconstrucción implícita	Mallado superficie	Mallado volumen	Total tiempo
Ligamento anterior	10.952	0,72	10,22	0,66	11,6
Ligamento posterior	16.096	0,98	12,28	0,82	14,08
Ligamento fibular	16.286	1,12	44,23	0,85	6,2
Ligamento tibial	22.478	1,52	57,31	1,07	7,9
Fíbula	25.762	1,63	2,41	0,77	4,81
Patela	33.244	2,18	2,03	0,94	5,15
Ligamento patelar	43.462	2,91	4,76	2,4	10,07
Tibia	133.576	9,72	4,19	4,24	18,15
Fémur	164.480	12,25	4,14	4,33	20,72

En las tablas 5.1 y 5.2 se puede observar que el mallado volumétrico de la función implícita toma más tiempo que el mallado de la superficie. Esto se debe al tamaño y la forma compleja que tienen algunas regiones como la vena y la arteria. Esta misma tendencia aparece cuando la estructura es delgada y compleja con valles y crestas, como es el caso de la materia gris y la materia blanca del conjunto de datos del cerebro. Esto se puede contrastar con el cerebelo, que aunque tiene una topología parecida a la materia gris y la materia blanca, es mucho más pequeño en tamaño y no tiene un grosor tan delgado. Sin embargo, en las tablas 5.3, 5.4 y 5.5 se puede apreciar que el tiempo para extraer la malla volumétrica es menor al tiempo necesario para extraer la malla de superficie.

La figura 5.3 muestra que el tiempo para la reconstrucción implícita de una región depende tanto del número de puntos de entrada como de la forma de esta. Se puede observar que las regiones con mayor puntos de entrada, la tibia y el fémur, tardan más tiempo en el proceso de la reconstrucción implícita que aquellas con un número menor de puntos, como los ligamentos; sin embargo, estos últimos necesitan más tiempo para el proceso de mallado de la superficie. Esto se debe a que cuando las regiones tienen formas alargadas y delgadas (como los ligamentos, la materia gris y la materia blanca), el algoritmo de mallado debe hacer cálculos adicionales que prevengan la aparición de huecos o triángulos invertidos debido a la proximidad de las superficies que definen la región.

Adicionalmente, se puede observar que el tiempo para el proceso de mallado del volumen se acerca al tiempo para el mallado de la superficie en aquellas regiones con una gran cantidad de puntos, como la tibia y el fémur, y al tiempo para la reconstrucción implícita si el número de puntos no es tan grande como en el caso de los ligamentos.

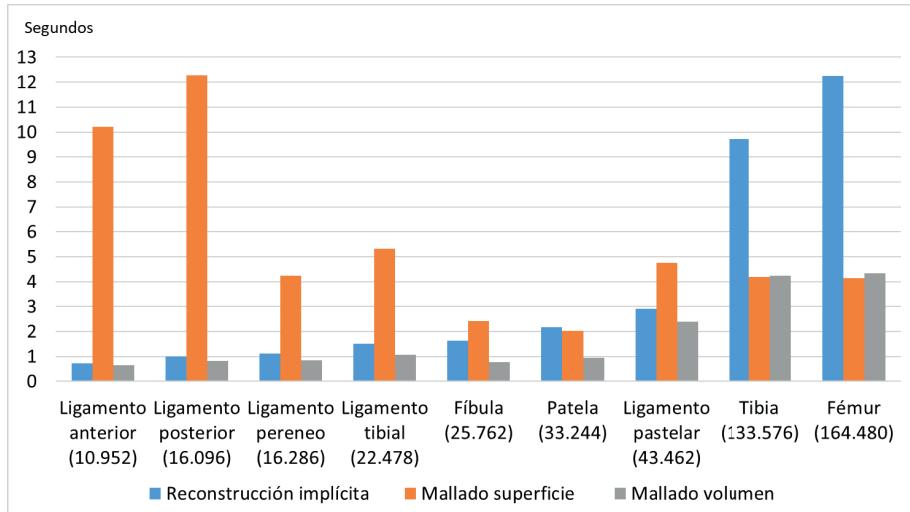


Figura 5.3: Tiempo para el modelado de las regiones de la rodilla

En la tabla 5.4 se disponen los datos del modelado de las regiones de la zona torácica. Estos datos se grafican en la figura 5.4, donde se puede observar que la extracción de la malla de superficie toma la mayor parte del tiempo del proceso de modelado.

Tabla 5.4: Tiempo para el modelado de las regiones de la zona torácica

Columna (zona torácica)	Puntos de entrada	Reconstrucción implícita	Mallado superficie	Mallado volumen	Total tiempo
Disco T12-T11	1.892	0,14	8,39	0,25	8,78
Disco T11-T10	1.942	0,13	17,06	0,28	17,47
Costilla T12	3.716	0,25	4,36	0,52	5,13
Vértebra T10	4.718	0,28	5,65	0,96	6,89
Vértebra T11	5.230	0,30	4,85	1,03	6,18
Vértebra T12	6.146	0,35	18,21	1,13	19,69
Costilla T11	6.494	0,39	6,25	0,83	7,47
Costilla T10	9.676	0,57	10,5	1,44	12,51

Caso especial es el disco intervertebral T11-T10, cuyo tiempo total de reconstrucción es de 17,47 segundos, de los cuales 17,06 segundos son dedicados al mallado de la superficie.

Algo similar sucede con el mallado de la superficie de la vértebra T12. Este se explica en la forma delgada que tienen los discos intervertebrales, lo que hace que las superficies externas estén muy cerca entre sí. El caso de las vértebras se debe al hueco que tienen en su interior, y que el algoritmo de mallado debe mantener.

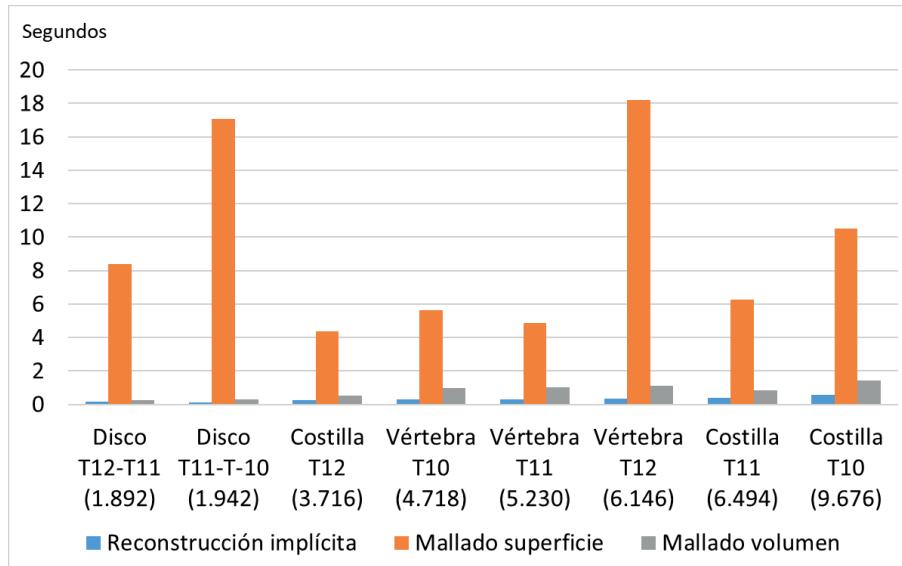


Figura 5.4: Tiempo para el modelado de las regiones de la zona torácica

La tabla 5.5 muestra los tiempos de reconstrucción de las regiones de la zona lumbar. Estos datos se grafican en la figura 5.5, que presenta el tiempo total para el modelado de las regiones de dicha zona. Como se ha mencionado, el tiempo de reconstrucción de una región depende tanto del número de puntos de entrada como de la forma de esta. Así, el tiempo total de los discos sacro-L5 y L5-L4 es mayor que el tiempo de las vértebras L4 y L5, aunque estas últimas tienen casi cuatro veces más puntos de entrada.

Tabla 5.5: Tiempo para el modelado de las regiones de la zona lumbar

Columna (zona lumbar)	Puntos de entrada	Reconstrucción implícita	Mallado superficie	Mallado volumen	Total tiempo
Disco sacro-L5	2.104	0,15	5,15	0,38	5,68
Disco L5-L4	2.674	0,17	3,28	0,51	3,96
Vértebra L4	8.158	0,46	1,30	1,79	3,55
Vértebra L5	8.830	0,48	1,63	1,74	3,85
Hueso sacro	16.398	0,87	0,76	2,79	4,42
Pelvis	34.510	2,15	0,73	2,01	4,89

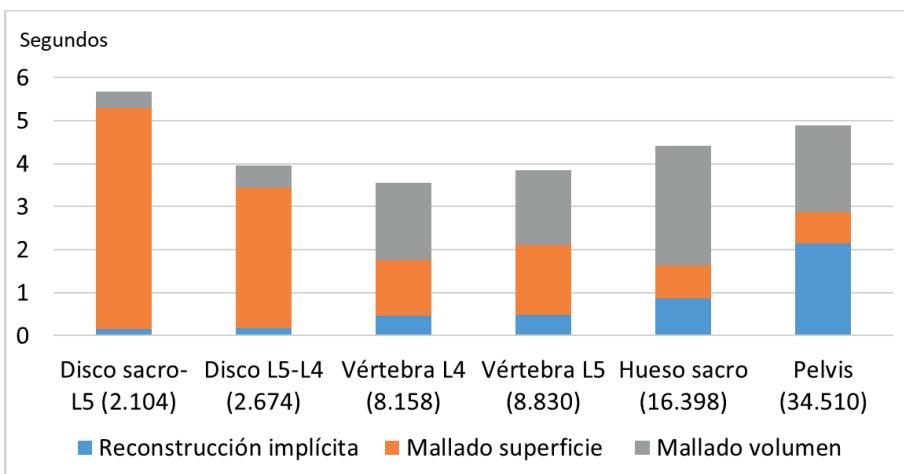


Figura 5.5: Tiempo para el modelado de las regiones de la zona lumbar

Finalmente, en la tabla 5.6 se listan los tiempos relacionados con los procesos de reconstrucción implícita, mallado de la superficie y mallado del volumen de las diferentes estructuras anatómicas heterogéneas modeladas en la sección 4.3.

Tabla 5.6: Tiempo para el modelado de las estructuras anatómicas heterogéneas

Nombre del modelo	Número de regiones	Reconstrucción implícita	Mallado superficie	Mallado volumen	Total tiempo
Columna-a	4	1,40	0,77	3,17	5,34
Columna-b	8	2,1	1,11	8,18	11,39
Pelvis	6	4,11	1,38	10,1	15,59
Rodilla-a	4	36,4	5,1	5,25	46,75
Rodilla-b	6	31,86	6,43	8,1	46,39
Rodilla-c	7	39,57	6,75	9,38	55,7
Abdomen-a	4	36,4	8,17	21,71	66,28
Abdomen-b	5	36,71	10,39	33,54	80,64
Abdomen-c	5	37,79	9,85	31,17	78,81
Hígado-tumores	4	11,85	3,84	9,1	24,79
Cerebro-a	2	19,9	12,86	30,37	63,13
Cerebro-b	3	39,93	17,29	109,8	167,02
Cerebro-c	4	58,05	19,15	150,24	227,44

Se puede observar que los modelos que toman menor tiempo son aquellos asociados al conjunto de datos de la columna, que solo necesitan algunos segundos para ser generados. En segundo lugar se tienen los modelos de la rodilla, que toman casi un

minuto para ser obtenidos. Luego se tienen los modelos asociados al conjunto de datos del abdomen, que tienen estructuras de buen tamaño y algunas con forma tubular. El tiempo requerido es de casi dos minutos para el modelo *Abdomen-c*, compuesto de cinco regiones. Finalmente, los modelos que toman mayor tiempo son aquellos asociados al conjunto de datos del cerebro, debido principalmente al tamaño y complejidad de la forma que presentan sus diferentes regiones. Sin embargo el tiempo total no superó los cuatro minutos para el caso del modelo *Cerebro-c*, que está compuesto de cuatro regiones (figura 4.12 ).

## 5.2. Precisión de los modelos

En esta sección se compara la malla generada por el algoritmo Marching Cubes (Lorensen y Cline, 1987) y la malla resultante del método RAM. Se escogió este algoritmo debido a que produce una malla que sigue muy de cerca la región segmentada de la imagen TAC, proporcionando una malla de referencia (subsección 2.3.1).

Tabla 5.7: Datos de los modelos de las regiones del cerebro

Región del cerebro	Modelo de entrada				Modelo obtenido			
	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )
Cerebelo	47.542	95.200	35.971	120.152	7.498	15.132	26.849	120.446
Cráneo	244.560	489.116	183.299	455.790	18.333	36.726	164.620	453.333
Materia blanca	287.520	287.520	209.232	533.385	59.411	119.982	159.246	524.197
Materia gris	540.360	1.082.116	388.734	553.408	132.756	268.928	269.101	562.816

En la tabla 5.7 se listan los datos relacionados con el número de puntos, el número de triángulos, el área y el volumen de las regiones segmentadas del cerebro, tanto del modelo de entrada como del modelo de salida generado por el método RAM.

Una característica importante del método RAM es la reducción de puntos para representar las diferentes regiones y estructuras después del proceso de extracción geométrica. Este hecho conlleva igualmente una reducción del espacio en disco necesario para almacenar la geometría de los modelos computacionales.

La figura 5.6 muestra el porcentaje de reducción de puntos y de triángulos en el proceso de modelado de las regiones del cerebro, así como el porcentaje de conservación del área y del volumen de estos modelos. El porcentaje de reducción de puntos está por encima

del 75 % para las regiones del cerebro (figura 5.6), del 88 % para las regiones del abdomen (figura 5.7) y del 89 % para las regiones de la rodilla (figura 5.8). Para las regiones de la zona torácica de la columna, este porcentaje supera el 80 % (figura 5.9) y el 89 % para la zona lumbar (figura 5.10).

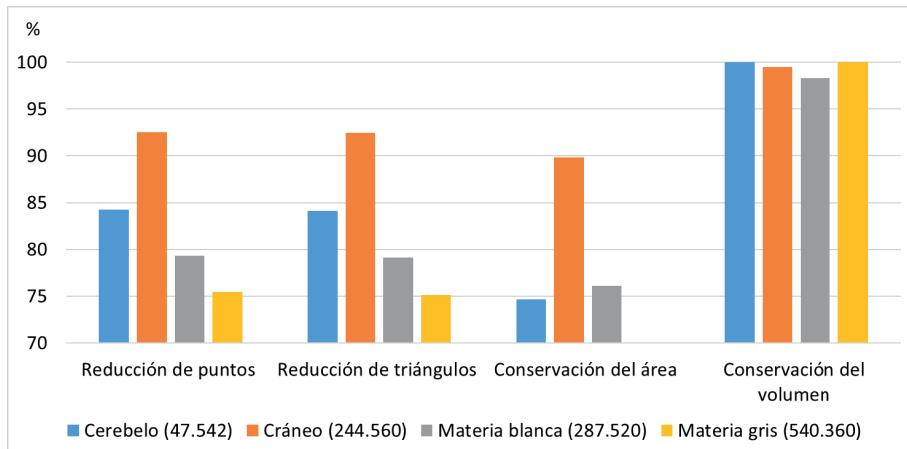


Figura 5.6: Estadísticas de la reconstrucción de las regiones del cerebro

Otra característica del método RAM está asociada a la reducción de triángulos de las mallas que representan la geometría de la superficie de las diferentes regiones. La tabla 5.8 y la figura 5.7 muestran que la reducción en el número de triángulos para representar la vena es del 88 %, llegando a una reducción del 97 % para el caso del hígado. Se puede observar que la reducción de triángulos está directamente relacionada con el porcentaje de reducción de puntos.

Tabla 5.8: Datos de los modelos de las regiones del abdomen

Región del abdomen	Modelo de entrada				Modelo obtenido			
	Puntos	Triángulos	Área (mm <sup>2</sup> )	Volumen (mm <sup>3</sup> )	Puntos	Triángulos	Área (mm <sup>2</sup> )	Volumen (mm <sup>3</sup> )
Tumor-2	5.228	10.452	2.325	7.500	330	656	1.999	7.374
Tumor-1	29.510	59.016	12.919	84.337	1.512	3.020	11.043	84.116
Riñón derecho	38.582	77.160	17.120	111.919	1.340	2.676	14.610	111.066
Riñón izquierdo	46.822	93.640	21.492	145.727	1.556	3.108	18.701	144.632
Arteria	57.946	115.626	27.141	59.052	5.958	11.940	23.253	57.979
Vena	93.714	186.992	41.720	73.068	10.666	21.336	35.373	72.055
Hígado	221.402	442.788	99.265	1.489.295	7.448	14.892	85.690	1.487.591

Una característica que se busca en los modelos 3D es que estos conserven el mayor área de superficie posible pero con el menor número de puntos. Por supuesto, entre mayor sea la reducción de puntos y triángulos para representar las regiones, menor será el porcentaje de conservación del área asociada a cada región. Esto se puede apreciar en la figura 5.6, donde la materia gris es el caso más sobresaliente, con una conservación del área de 70 %.

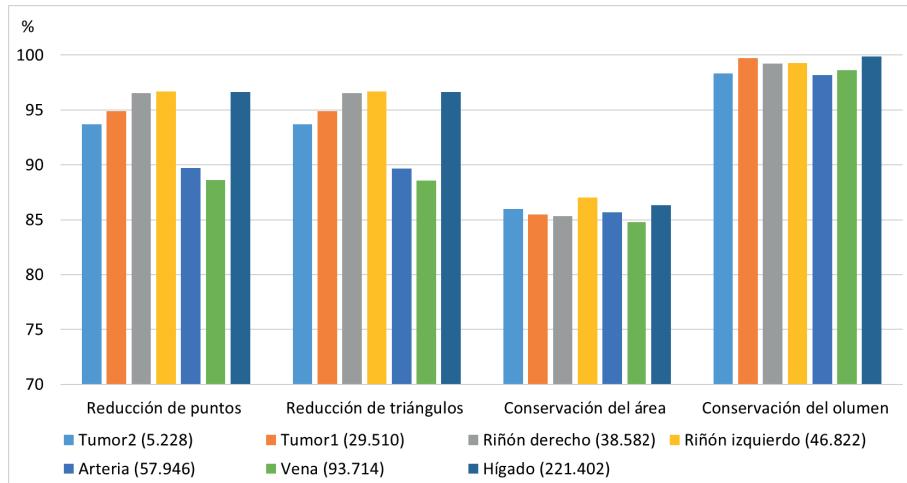


Figura 5.7: Estadísticas de la reconstrucción de las regiones del abdomen

La reducción en el área de superficie se debe principalmente a la suavidad que el proceso de reconstrucción implícita del método RAM aplica a los modelos (sección 3.2). Emplear un procedimiento simple, como el suavizado laplaciano, reduce de manera significativa el área en la región que exhibe una gran variación (Taubin, 1995).

La tabla 5.9 contiene los datos del modelado de las regiones de la rodilla. La conservación del área va desde el 82 % para el caso del ligamento posterior, hasta el 87 % para el caso del fémur (figura 5.8)

Tabla 5.9: Datos de los modelos de las regiones de la rodilla

Región de la rodilla	Modelo de entrada				Modelo obtenido			
	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )
Ligamento anterior	10.952	21.900	1.379	1.769	804	1.604	1.185	1.732
Ligamento posterior	16.096	32.188	1.961	2.452	1.190	2.376	1.590	2.417
Ligamento fibular	16.286	32.564	2.139	2.200	1.132	2.260	1.780	2.152
Ligamento tibial	22.478	44.948	2.970	3.367	1.242	2.480	2.495	3.321
Fíbula	25.762	51.520	3.081	10.608	1.071	2.138	2.628	10.565
Patela	33.244	66.484	4.366	15.789	1.221	2.438	3.781	15.725
Ligamento patelar	43.462	86.948	5.757	6.122	2.476	4.952	4.873	6.080
Tibia	133.576	267.060	14.474	91.582	5.899	11.794	12.245	91.547
Fémur	164.480	328.956	19.986	141.400	5.618	11.232	17.331	141.231

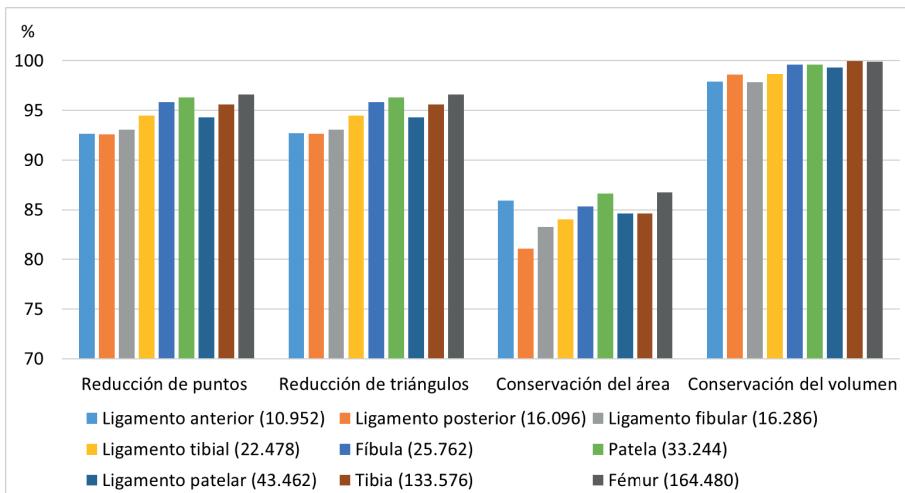


Figura 5.8: Estadísticas de la reconstrucción de las regiones de la rodilla

Otra característica que buscan los modelos 3D es la conservación del volumen. En todos los experimentos realizados este fue superior al 97 %. Sin embargo, se puede observar en las tablas 5.10 y 5.11 que para algunas regiones, como las costillas T-11 y T-10 y la pelvis, el volumen de los modelos aumentó con respecto a los datos de entrada. Este

hecho se puede apreciar claramente en las figuras 5.9 y 5.10.

Tabla 5.10: Datos de los modelos de las regiones de la zona torácica

Región de la zona torácica	Modelo de entrada				Modelo obtenido			
	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )
Disco T12-T11	1.892	3.780	1.624	3.092	162	320	1.357	2.813
Disco T11-T10	1.942	3.880	1.673	3.524	148	292	1.396	3.262
Costilla T12	3.716	7.424	3.398	4.678	746	1.484	3.107	5.610
Vértebra T10	4.718	9.436	4.212	7.839	513	1.026	3.468	7.553
Vértebra T11	5.230	10.460	4.662	9.755	511	1.022	3.773	9.413
Vértebra T12	6.146	12.292	5.460	11.678	640	1.280	4.364	11.258
Costilla T11	6.494	12.980	5.884	8.120	766	1.524	5.836	11.100
Costilla T10	9.676	19.344	8.856	13.587	976	1.944	8.663	17.938

Tabla 5.11: Datos de los modelos de las regiones de la zona lumbar

Región de la zona lumbar	Modelo de entrada				Modelo obtenido			
	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )	Puntos	Triángulos	Área ( $mm^2$ )	Volumen ( $mm^3$ )
Disco sacro-L5	2.104	4.204	1.769	3.057	199	394	1.411	2.851
Disco L5-L4	2.674	5.344	2.291	4.267	248	492	1.866	3.896
Vértebra L4	8.158	16.316	7.295	16.913	862	1.724	5.921	16.473
Vértebra L5	8.830	17.660	7.780	19.951	860	1.728	6.333	19.442
Hueso sacro	16.398	32.796	14.503	56.947	1.308	2.616	12.099	56.591
Pelvis	34.510	69.012	29.760	114.367	2.880	5.752	25.328	118.729

Para explicar el aumento en el volumen de algunas regiones, se debe recordar que el proceso de mallado del método RAM permite el control sobre la resolución de las mallas; es decir, el proceso de extracción de la geometría crea pocos triángulos donde la superficie es plana y más triángulos donde la superficie tiene curvaturas irregulares. Lo anterior puede causar que triángulos grandes en las zonas planas de la región no sigan fielmente los contornos del modelo de voxel producido por el proceso de segmentación original de la imagen TAC, lo que puede afectar el cálculo del área y del volumen de los modelos obtenidos.

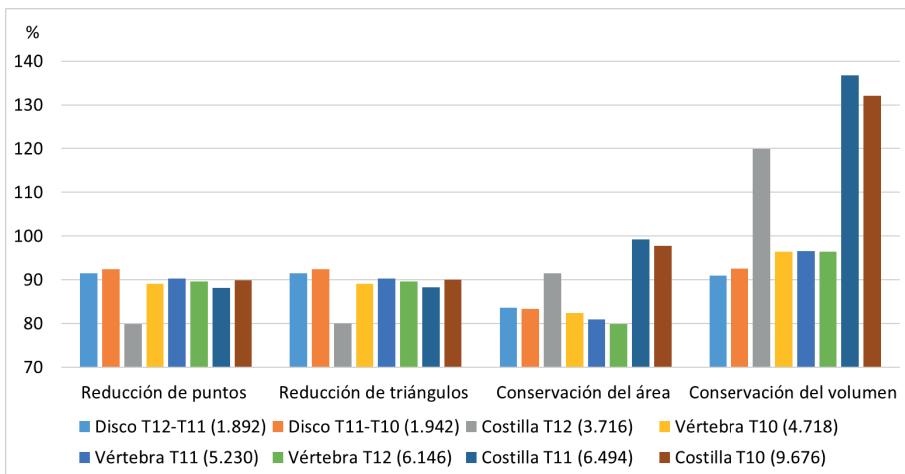


Figura 5.9: Estadísticas de la reconstrucción de las regiones de la zona torácica

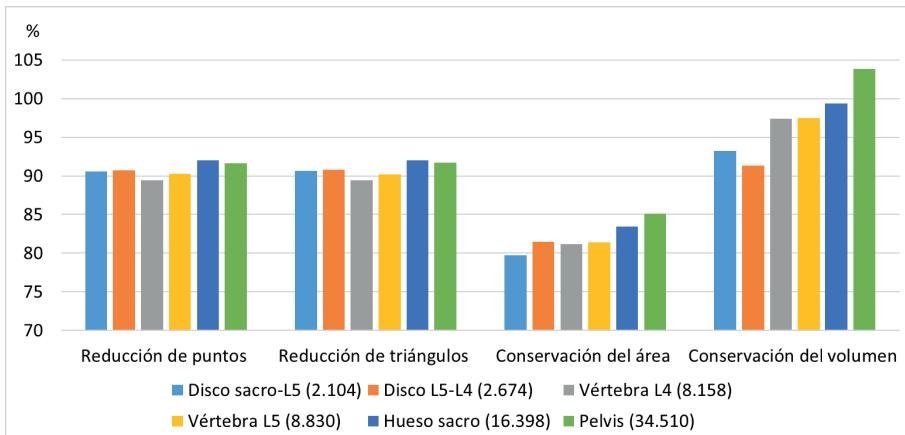


Figura 5.10: Estadísticas de la reconstrucción de las regiones de la zona lumbar

La figura 5.10 muestra la conservación del volumen de las diferentes regiones de la zona lumbar. Los valores van desde el 92 % para el disco L5-L4, hasta el 104 % para la región de la pelvis. Finalmente, en la tabla 5.12 se presentan los datos relacionados con el número de regiones, el número de triángulos, el área, el número de tetraedros y el volumen de las diferentes estructuras heterogéneas modeladas en la sección 4.3.

Tabla 5.12: Datos de los modelos de las estructuras anatómicas heterogéneas

Nombre del modelo	Número de regiones	Triángulos	Área ( $mm^2$ )	Tetraedros	Volumen ( $mm^3$ )
Columna-a	4	3.618	18.780	9.313	37.436
Columna-b	8	4.642	23.932	15.838	59.335
Pelvis	6	6.250	44.018	30.109	212.919
Rodilla-a	4	17.150	36.230	14.614	253.135
Rodilla-b	6	22.656	37.445	19.484	257.436
Rodilla-c	7	23.508	40.776	21.099	269.352
Cerebro-a	2	60.042	178.218	129.915	644.796
Cerebro-b	3	76.376	220.294	469.626	1.219.210
Cerebro-c	4	82.366	310.569	50.8011	1.673.130
Abdomen-a	4	34.916	170.536	105.689	1.798.700
Abdomen-b	5	43.758	191.599	122.213	1.856.540
Abdomen-c	5	44.072	176.113	122.681	1.600.610
Hígado-tumores	4	14.876	98.869	41.930	1.486.890

No se presenta una comparación con los datos de entrada ya que estos los proporcionan las regiones que componen estas estructuras y que están plasmados en las tablas 5.7, 5.8, 5.9, 5.10 y 5.11. Se debe resaltar el hecho de que las estructuras heterogéneas se obtuvieron a partir de vectores de funciones implícitas y no de nubes de puntos que representaran las estructuras como tal. El método RAM permite este tipo de situaciones, las cuales serían consideradas como un modelado de regiones (sección 4.2) y no un modelado de estructuras heterogéneas.

## 5.3. Calidad geométrica

La calidad geométrica se refiere a cuán cerca de la estructura anatómica real se encuentra el modelo geométrico obtenido. El coeficiente de Dice (subsección 5.3.1) y la distancia de Hausdorff (subsección 5.3.2) son las medidas escogidas para analizar la calidad geométrica de los modelos geométricos obtenidos mediante el método RAM.

### 5.3.1. Coeficiente de Dice

El coeficiente de Dice  $C_{Dice}$  se usa para medir la relación de superposición de dos modelos como un porcentaje del volumen total ocupado por ambos (Dice, 1945). Se

calcula como el doble de la relación del número de véxoles contenidos en la intersección de dos modelos con el número total de véxoles contenidos en cada uno:

$$C_{Dice} = \frac{2 | A \cap B |}{| A | + | B |} \quad (5.1)$$

La tabla 5.13 dispone los resultados del cálculo del coeficiente de Dice, utilizando el *software* MeshValmet (Xu, 2011), para todas las regiones del abdomen, la rodilla, la columna y el cerebro, modeladas en la sección 4.2.

Tabla 5.13: Coeficiente de Dice de los modelos de las regiones

Regiones del abdomen	Dice	Regiones de la columna (zona torácica)	Dice
Vena	0,9435	<i>Costilla T11</i>	<i>0,7762</i>
Arteria	0,9602	<i>Costilla T10</i>	<i>0,7807</i>
Tumor-2	0,9764	<i>Costilla T12</i>	<i>0,8356</i>
Tumor-1	0,987	Vértebra T10	0,924
Riñón derecho	0,9873	Disco T12-T11	0,9264
Riñón izquierdo	0,9885	Vértebra T11	0,9291
Hígado	0,9946	Vértebra T12	0,9318
		Disco T11-T10	0,9345
Regiones de la rodilla	Dice	Regiones de la columna (zona lumbar)	Dice
Ligamento fibular	0,9596	Disco sacro-L5	0,9116
Ligamento patelar	0,9629	Disco L5-L4	0,9234
Ligamento tibial	0,9643	Vértebra L4	0,9353
Ligamento posterior	0,9651	Pelvis	0,9397
Ligamento anterior	0,968	Vértebra L5	0,9426
Fíbula	0,9879	Hueso sacro	0,966
Patela	0,9885		
Tibia	0,9932		
Fémur	0,9942		
Regiones del cerebro	Dice		
Materia gris	0,888		
Materia blanca	0,9406		
Cráneo	0,9575		
Cerebelo	0,959		

Al no contar con regiones reales para hacer las comparaciones del caso, se toman como base las regiones segmentadas (modelos de véxel) a partir de las cuales se crea una malla de prueba por medio del algoritmo Marching Cubes. Para los cálculos se toma el modelo obtenido con el método RAM ( $A$ ) y el modelo generado directamente a partir de

las segmentaciones por el algoritmo de Marching Cubes ( $B$ ). Los valores del coeficiente de Dice van desde 0 (sin superposición) hasta 1 (congruencia total).

Se puede observar que los valores del coeficiente de Dice están por encima de 0,94 para las estructuras generadas del conjunto de datos del abdomen, de 0,96 para las estructuras de la rodilla, de 0,88 para las estructuras del cerebro, y de 0,78 y 0,91 para las regiones de la zona torácica y zona lumbar, respectivamente.

El coeficiente de Dice presenta una correlación directa con el porcentaje de conservación de volumen. Así, las estructuras anatómicas con un coeficiente de Dice cerca a 1 son aquellas que han conservado su volumen en un porcentaje superior al 99 %.

La figura 5.11 muestra esta relación para el caso de las estructuras anatómicas del abdomen, misma tendencia que se presenta en los conjuntos de datos de la rodilla, la columna y el cerebro. Para hacer la comparación se normalizó el coeficiente de Dice multiplicándolo por 100. Los valores más bajos del coeficiente de Dice corresponden a las costillas T10, T11 y T12, resaltados en color rojo en la tabla 5.13. Estas regiones son las mismas que presentan un aumento del volumen con respecto a los datos de entrada (tabla 5.10).

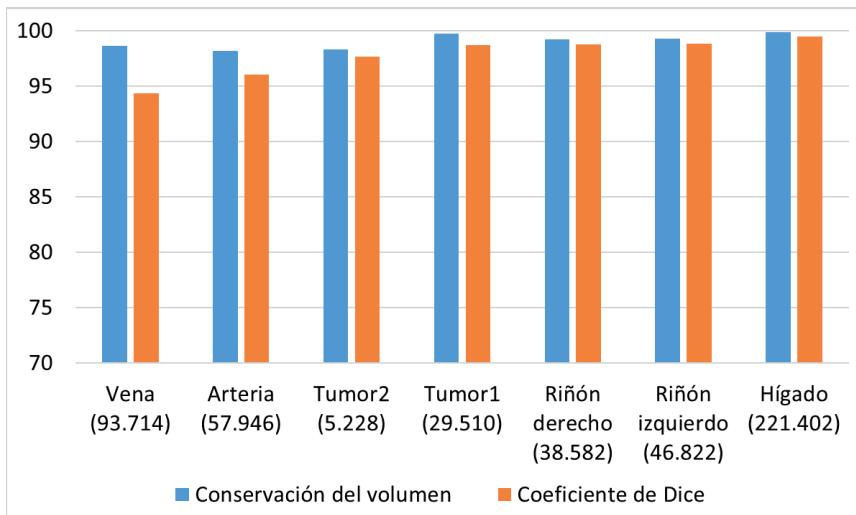


Figura 5.11: Coeficiente de Dice vs. conservación del volumen en regiones del abdomen

En la figura 5.12 se presenta la relación entre el coeficiente de Dice y la conservación del volumen de las regiones de la zona torácica. Se puede observar que cuando la conservación del volumen sobrepasa el 100 %, el coeficiente de Dice comienza a

disminuir. Esto se debe a que es una medida de superposición entre dos modelos, y entre más se desvíe esta superposición, más disminuirá el valor del coeficiente, como queda evidenciado en la figura 5.11.

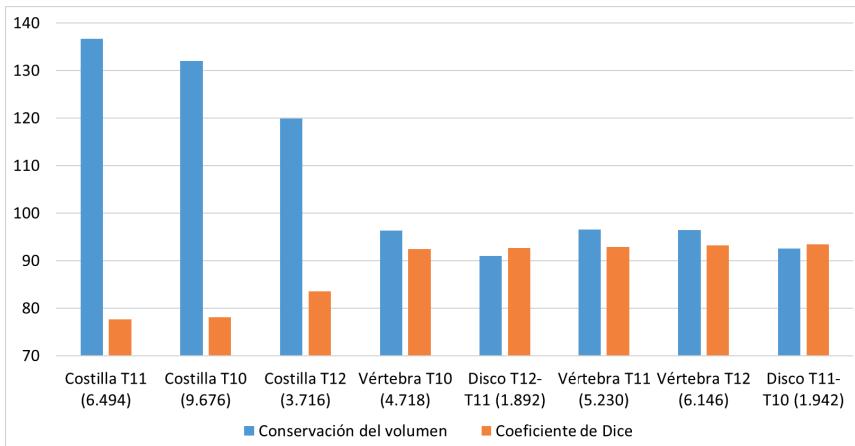


Figura 5.12: Coeficiente de Dice de las regiones de la zona torácica

En la tabla 5.14 se presentan los resultados del cálculo del coeficiente de Dice en las diferentes estructuras anatómicas heterogéneas modeladas en la sección 4.3 con el método RAM. Para hacer los cálculos se generó una malla inicial (B) para cada estructura heterogénea, a partir de la combinación de los diferentes modelos de voxel que conforman dicha estructura utilizando el *software* Paraview (Ayachit, 2015).

Tabla 5.14: Coeficiente de Dice de las estructuras heterogéneas modeladas

Nombre del modelo	Coeficiente de Dice
Columna-a	0,9119
Columna-b	0,9056
Rodilla-a	0,9879
Pelvis	0,9497
Rodilla-b	0,9844
Rodilla-c	0,9756
Cerebro-a	0,9374
Cerebro-b	0,9016
Cerebro-c	0,9415
Abdomen-a	0,9065
Abdomen-b	0,9071
<i>Abdomen-c</i>	<i>0,8448</i>
Hígado-tumores	0,9651

Se puede observar, que el coeficiente de Dice es superior al 0,90 para todos los modelos, a excepción del modelo *abdomen-c*, que es de 0,84. Esto se debe principalmente a que el modelo posee dos estructuras tubulares: la vena y la arteria, que interconectan los riñones y el hígado.

### 5.3.2. Distancia de Hausdorff

Dado que los modelos obtenidos con el método RAM pueden tener alguna anomalía geométrica, y todas las medidas de superposición como coeficiente de Dice pueden dar una muy buena puntuación, se decidió utilizar la distancia de Hausdorff para detectar cualquier desviación fuerte con respecto a la malla base.

La distancia de Hausdorff  $D_{Hausdorff}$  mide la diferencia entre dos modelos geométricos en términos de las distancias entre muestras de puntos en las superficies de ambos modelos (Hausdorff, 2008). Dados dos conjuntos de puntos finitos  $A = \{a_1, \dots, a_n\}$  y  $B = \{b_1, \dots, b_m\}$ , para cada punto en la superficie  $A$  la distancia a la superficie  $B$  se define como la distancia euclíadiana al punto más cercano en la superficie  $B$ :

$$D_{Hausdorff}(A, B) = \max(d(A, B), d(B, A)) \quad (5.2)$$

Donde  $d(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|$  y  $\|\cdot\|$  es la distancia entre los puntos  $a$  y  $b$  medida por alguna norma ( $L_1, L_2, L_\infty$ ).

La función  $d(A, B)$  se conoce como la distancia directa de Hausdorff desde  $A$  a  $B$ , y ordena cada punto de  $A$  con base en su distancia al punto más cercano de  $B$ . Así, los valores de la distancia de Hausdorff van desde 0, que significa que hay una total coincidencia entre los dos modelos, hasta cualquier valor que indica la distancia de un punto de  $A$  al punto más cercano de  $B$ . Es importante observar que esta distancia es en general no simétrica, es decir,  $d(A, B) \neq d(B, A)$ .

La tabla 5.15 muestra los resultados de calcular la distancia de Hausdorff, utilizando el software MeshValmet (Xu, 2011), entre el modelo obtenido por el método RAM( $A$ ) y el modelo generado por el algoritmo Marching Cubes( $B$ ). Esta distancia es inferior a 0,88 para las estructuras del abdomen, e inferior a 0,46 para las estructuras de la rodilla. Sin embargo, la distancia de Hausdorff aumenta hasta 1,39 para las estructuras del cerebro, y hasta 1,83 y 2,69 para las estructuras de la zona lumbar y zona torácica, respectivamente.

Tabla 5.15: Distancia de Hausdorff de los modelos de las regiones segmentadas del abdomen, la rodilla, la columna y el cerebro

Regiones del abdomen	$d(A, B)$	Regiones de la columna (zona torácica)	$d(A, B)$
Tumor-2	0,7315	Vértebra T10	1,0872
Vena	0,7729	Vértebra T11	1,134
Arteria	0,7768	Disco T11-T10	1,1424
Riñón izquierdo	0,7961	Vértebra T12	1,2421
Riñón derecho	0,799	Disco T12-T11	1,4857
Tumor-1	0,8492	<i>Costilla T12</i>	<i>1,7375</i>
Hígado	0,8778	<i>Costilla T11</i>	<i>2,4622</i>
		<i>Costilla T10</i>	<i>2,6883</i>
Regiones de la rodilla	$d(A, B)$	Regiones de la columna (zona lumbar)	$d(A, B)$
Ligamento fibular	0,3768	Vértebra L4	1,217
Ligamento tibial	0,3869	Disco sacro-L5	1,2762
Ligamento posterior	0,4012	Vértebra L5	1,306
Patela	0,4104	Hueso sacro	1,3125
Ligamento anterior	0,4287	Disco L5-L4	1,4904
Fíbula	0,4431	<i>Pelvis</i>	<i>1,8277</i>
Ligamento patelar	0,4501		
Fémur	0,459		
Tibia	0,4603		
Regiones del cerebro	$d(A, B)$		
Cerebelo	1,106		
Materia blanca	1,217		
Materia gris	1,3774		
Cráneo	1,3817		

La figura 5.13 compara la conservación del área, la conservación del volumen y la distancia de Hausdorff de las regiones de la zona torácica. Para hacer esta comparación se normalizó dividiendo por 100, tanto la conservación del área como la conservación del volumen.

Se puede ver que no hay una relación directa entre la conservación del área y la distancia de Hausdorff. Sin embargo, esta relación sí existe entre la distancia de Hausdorff y la conservación del volumen. Pero, a diferencia del coeficiente de Dice, cuando la conservación del volumen sobrepasa el 100%, la distancia de Hausdorff comienza a aumentar, como se aprecia en las regiones resaltadas en rojo en la tabla 5.15.

Asimismo, se puede observar en la tabla 5.15 que la distancia de Hausdorff de la región de la pelvis es igualmente alto. Esto evidencia que puede haber un aumento en el volumen del modelo resultante, como lo demuestra la figura 5.10. Sin embargo, este

hecho no fue detectado por el coeficiente de Dice.

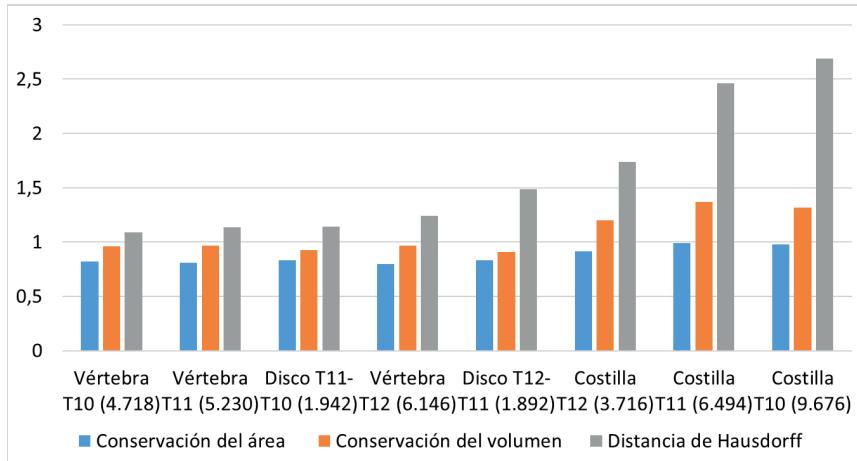


Figura 5.13: Distancia de Hausdorff vs. conservación del área y del volumen de las regiones de la zona torácica

En la tabla 5.16 se presenta la distancia de Hausdorff de las estructuras heterogéneas con diferentes configuraciones modeladas en la sección 4.3. Se puede observar que para el caso de las estructuras heterogéneas con regiones anidadas la distancia de Hausdorff coincide con la distancia de Hausdorff de la región que anida las otras. Así, la distancia de Hausdorff del modelo hígado-tumores coincide con la distancia de Hausdorff del hígado.

Tabla 5.16: Distancia de Hausdorff de las estructuras heterogéneas modeladas

Nombre del modelo	Distancia $d(A, B)$
Columna-a	1,3472
Columna-b	1,468
Pelvis	1,3543
Rodilla-a	0,7665
Rodilla-b	0,5356
Rodilla-c	0,5634
Cerebro-a	1,324
Cerebro-b	1,6123
Cerebro-c	1,608
Abdomen-a	1,5568
Abdomen-b	1,5299
Abdomen-c	2,0997
Hígado-tumores	0,8867

# Conclusiones

---

En este libro se abordó el problema del modelado 3D de estructuras anatómicas heterogéneas a partir de imágenes médicas. Para contribuir con la solución a este problema se ha propuesto el método RAM (Region Aware Modeling), que tiene en cuenta todas las regiones que conforman las estructuras anatómicas heterogéneas representadas en las imágenes médicas TAC.

El método RAM aplica un enfoque sistemático de modelado ascendente (*bottom-up*) con base en ingeniería inversa. En este enfoque, las estructuras anatómicas heterogéneas se descomponen en regiones a partir de las cuales se extrae una nube de puntos, a través de la cual se realiza la reconstrucción implícita de la superficie de cada región. La geometría total de la estructura anatómica es modelada por medio de un vector de funciones implícitas (*VFI*). Como resultado, los modelos 3D son independientes de la resolución, precisos y matemáticamente rigurosos.

El método RAM consta de dos procesos principales: el proceso de reconstrucción implícita y generación del vector de funciones implícitas, y el proceso de reconstrucción geométrica mediante el mallado de superficie y de volumen. Tanto las mallas de superficie como de volumen se generan automáticamente con la calidad que requieren las aplicaciones de ingeniería y con límites acordes con las diferentes regiones que conforman la estructura anatómica (Wesseling *et al.*, 2016; Wu *et al.*, 2016).

Aunque el enfoque general del método RAM hace énfasis en la fidelidad de los modelos 3D a las estructuras reales representadas en las diferentes segmentaciones de las imágenes médicas, los siguientes factores afectan la precisión del método:

- **El algoritmo de reconstrucción de superficies implícitas.** En la subsección 2.2.3 se revisaron tres algoritmos: funciones de base radial (Carr *et al.*, 2001), partición de la unidad multivariante (Ohtake *et al.*, 2005) y reconstrucción de superficies de Poisson (Kazhdan *et al.*, 2006). La elección del algoritmo para generar la superficie implícita desde la nube de puntos de entrada hace la

diferencia. Aunque todos los algoritmos deberían producir una superficie implícita que divida el espacio en regiones internas y externas, la forma en que los puntos se interpolarán o aproximarán variará de un algoritmo al siguiente.

- **Parámetros del algoritmo de mallado.** Al igual que en el ítem anterior, la elección del algoritmo para generar mallas de superficie y mallas de volumen a partir de la representación implícita incide en la calidad de los modelos. Incluso si se utiliza el mismo algoritmo de mallado, tales algoritmos a menudo tienen parámetros de entrada. Por ejemplo, en el proceso de reconstrucción geométrica del método RAM tres parámetros de entrada controlan el procedimiento de mallado de superficie, y cinco parámetros el procedimiento de mallado de volumen (sección 3.3). Modificar los valores de los parámetros puede alterar los resultados.
- **El tipo de imagen médica.** Con cualquier tipo de imagen médica, el espacio donde se encuentran las estructuras anatómicas se muestrea y reconstruye matemáticamente; como resultado, la calidad del modelo 3D está limitada por la precisión del algoritmo de reconstrucción y la resolución de la imagen (subsección 2.1.1). Al utilizar imágenes de mejor calidad, es decir, de mayor resolución, se pueden obtener modelos 3D más precisos; sin embargo, se afectaría la eficiencia computacional al necesitar mayores recursos para procesar una mayor cantidad de información.

Además, los siguientes factores afectan la eficiencia computacional del método RAM:

- **Número de regiones.** Hay una sobrecarga asociada con cada región adicional que deba agregarse al modelado de la estructura heterogénea. El proceso de reconstrucción geométrica del método RAM debe moverse a través del conjunto de regiones implícitas y luego determinar la ubicación del límite entre las regiones que forman parte de las estructuras heterogéneas.
- **Tamaño de la nube de puntos.** Como el tamaño de las estructuras segmentadas puede variar mucho, esto se traduce en una gran diferencia en el número de puntos necesarios para representar la estructura anatómica. Las nubes de puntos más grandes requieren más tiempo para obtener la reconstrucción de la superficie implícita.
- **Complejidad de forma.** Las estructuras con formas más complejas requieren más iteraciones del algoritmo de reconstrucción implícita para capturar los detalles del límite. Igualmente, el algoritmo de mallado tardará más en determinar si el

modelo generado coincide con los límites de la función implícita. Intuitivamente, las formas más complejas serán más difíciles de reconstruir y mallar que las más simples.

- **Coincidencias de límites.** Los límites compartidos por más de dos regiones tomarán más tiempo para determinar dónde deben ajustarse las interfaces entre las regiones.

Sin embargo, el enfoque de modelado presentado en este libro tiene características importantes que se pueden resumir como:

- **Preservación de la forma.** El método RAM minimiza los errores de aproximación geométrica, de modo que los detalles resultantes parecen similares a sus contrapartes originales. El método RAM también genera una transición suave y continua entre las regiones interiores y la región exterior.
- **Robustez.** El presente enfoque permite al usuario modelar estructuras anatómicas con un número arbitrario de regiones, cada una de ellas con diferentes características y complejidad. También le permite especificar algunos parámetros geométricas como el ángulo mínimo de los triángulos que evita tener triángulos planos, no deseados para propósitos de simulación. Los resultados experimentales muestran que el enfoque propuesto puede funcionar de manera efectiva al modelar estructuras heterogéneas con diferentes configuraciones.
- **Eficiencia.** El enfoque presente en el método RAM está bien adaptado a los grandes conjuntos de datos que se encuentran en las imágenes médicas, ya que utiliza una estructura de datos recursiva de árboles octree que dividen el espacio para adaptarse a los detalles de la superficie local. Por lo tanto, el tiempo de cálculo depende de la complejidad de la superficie, en lugar del tamaño de la imagen. Además, el error de aproximación geométrica puede ser controlado por el usuario y limitado.

Estas características y los resultados de los experimentos sugieren la validez del método RAM, y de todo el enfoque en general, para modelar tanto regiones anatómicas individuales como estructuras anatómicas heterogéneas.

El trabajo de investigación sobre el modelado 3D de estructuras anatómicas heterogéneas para aplicaciones en áreas de la ingeniería está lejos de estar terminado. El enfoque presentado en este libro demostró solo una alternativa para el modelado

de estas estructuras. Para aplicaciones prácticas, las funcionalidades proporcionadas por el método RAM son todavía muy limitadas. Estas aplicaciones incluyen simulación quirúrgica, exploración virtual del cuerpo humano, diseño y fabricación de prótesis a la medida. Por lo tanto, todavía hay mucho espacio para mejorar las funcionalidades y crear nuevos desarrollos prácticos y teóricos. Las siguientes son algunas direcciones de trabajo futuro:

- **Una implementación de GPU.** Las unidades modernas de procesamiento de gráficos (GPU) se encuentran entre los chips de procesamiento más poderosos que existen en la actualidad. Las GPU de última generación son capaces de más de 12 teraflops de operaciones aritméticas de punto flotante y tienen más de 250 GB de memoria, mucho más que las actuales CPU de múltiples núcleos. El enfoque de *dividir y conquistar* del método RAM tiene suficiente paralelismo de datos para funcionar bien en las GPU. La implementación de GPU del método RAM con suficientes aceleraciones de rendimiento podría reconstruir modelos 3D de estructuras anatómicas en tiempo real.
- **Un sistema integrado de asignación de materiales.** Los modelos obtenidos en el capítulo 4 pueden medirse, modificarse y usarse en diferentes áreas de la ingeniería; sin embargo, no pueden registrar la composición local de los materiales. Esta información se tiene en las imágenes médicas no segmentadas o podría adicionarse mediante modelos de ingeniería asistida por computadora (CAE). El desarrollo de un sistema de asignación de materiales integrado con el método RAM podría contribuir a realizar un proceso de modelado más completo de las estructuras anatómicas heterogéneas.
- **Un sistema de simulación basado en realidad virtual.** Este libro se ha centrado en la creación de modelos tridimensionales de estructuras anatómica que solo pueden visualizar los usuarios mediante un proceso de renderizado. Sin embargo, las técnicas para interactuar con los modelos tridimensionales también son importantes para brindar a los usuarios experiencias más ricas e intuitivas. El desarrollo de un sistema de simulación con base en realidad virtual permitiría que interactúen de manera realista con los modelos 3D obtenidos mediante el método RAM.

# Apéndice A

---

## Conceptos relevantes

En este anexo se revisan una variedad de conceptos relevantes para el modelado tridimensional de estructuras anatómicas heterogéneas. Para un mejor entendimiento se ha organizado por secciones: Conceptos topológicos (sección A.1), Curvas y superficies (sección A.2), Reconstrucción tomográfica (sección A.3) y Estructuras de datos espaciales (sección A.4). Estos conceptos y definiciones de ninguna manera son exhaustivos, pero sí importantes para comprender mejor el área del modelado de objetos sólidos heterogéneos en general y del método RAM en particular.

### A.1. Conceptos topológicos

**Definición 4.** Dado un conjunto  $X$ , una *topología* en  $X$  consiste en una familia de subconjuntos  $T$ , llamados *conjuntos abiertos*, que cumplen las siguientes condiciones:

- El conjunto vacío  $\emptyset$  es abierto y todo el espacio  $X$  es abierto,  $\{\emptyset, X\} \subset T$ .
- $U \in T$  y  $V \in T \Rightarrow U \cap V \in T$
- Si  $V_i | i \in I$  es cualquier familia de conjuntos abiertos, entonces su unión  $\bigcup_{i \in I} V_i$  es abierta.

#### A.1.1. Espacio topológico

**Definición 5.** Un par  $(X, T)$ , donde  $T$  es una topología en  $X$ , se denomina *espacio topológico*. Los elementos del espacio subyacente  $X$  se conocen generalmente como

puntos. Dado  $x, y \in X$ , donde  $X$  es un espacio topológico, se dice que  $x$  e  $y$  pueden separarse por vecindarios si existe un vecindario  $U$  de  $x$ , y un vecindario  $V$  de  $y$  tal que  $(U \cap V) = \emptyset$ .

Un homeomorfismo es un isomorfismo especial entre espacios topológicos que respeta las propiedades topológicas. Dos espacios homeomórficos son idénticos desde un punto de vista topológico. Uno puede ver un homeomorfismo entre dos espacios como un estiramiento y combinación continuo de un espacio en el otro. Por ejemplo, un cuadrado y un círculo son homeomórficos. Intuitivamente, un homeomorfismo mapea puntos en el primer espacio que están muy cerca de puntos en el segundo espacio, y viceversa.

**Definición 6.** Una función  $f$  entre dos espacios topológicos  $X$  e  $Y$  se llama homeomorfismo si tiene las siguientes propiedades:

- $f$  es una biyección
- $f$  es continuo
- $f^{-1}$  es continuo

Por ejemplo, el intervalo abierto  $(-1, 1)$  es homeomorfo a los números reales  $\mathbb{R}$ .

**Definición 7.** Un espacio topológico  $X$  se denomina *espacio de Hausdorff* si dos puntos distintos de  $X$  se pueden separar por vecindarios.

### A.1.2. Variedad

Bernhard Riemann fue el primer matemático que extendió sistemáticamente la noción de superficie a los objetos de mayores dimensiones, a los que llamó *Mannigfaltigkeit*. De este término procede el inglés *manifold*. Riemann ofrece una descripción intuitiva de variedad, considerando una variedad de dimensión  $n$  como un apilamiento continuo de variedades de dimensión  $n-1$ .

Intuitivamente, una variedad es un espacio topológico que es localmente euclíadiano. Localmente, estos espacios se ven tal como los describe la geometría euclíadiana, pero pueden tener una estructura más complicada cuando se ven como un todo. Un ejemplo de esto es la superficie de la tierra. Para una persona de pie en la superficie parece plana, pero cuando se ve desde el espacio, se ve esférica. Se puede construir una variedad al pegar espacios euclidianos separados entre sí. Por ejemplo, se puede construir un mapa mundial pegando muchos mapas de regiones locales.

**Definición 8.** Una *variedad*, en matemática, es el objeto geométrico estándar que generaliza la noción intuitiva de curva (1-variedad) y de superficie (2-variedad) a cualquier dimensión y sobre cuerpos diversos. Se puede decir que una variedad de dimensión  $n$  ( $n$ -variedad) es un espacio topológico de Hausdorff en el que cada punto tiene un vecindario homeomorfo a una  $n$ -bola euclíadiana abierta. Un punto  $x = (x_1, x_2, \dots, x_n)$  está dentro de una  *$n$ -bola euclíadiana* si y solo si:

$$B^n = \{(x_1, x_2, \dots, x_n) | x_1^2 + x_2^2 + \dots + x_n^2 < 1\} \quad (\text{A.1})$$

Todas las variedades con una misma dimensión  $n$  o  $n$ -variedades tienen la misma topología local. Así, una pequeña porción de la curva es análoga a una recta y una pequeña porción de superficie es análoga a un plano. Una superficie es un ejemplo de una variedad-2. El carácter bidimensional de una superficie proviene del hecho de que alrededor de cada punto hay un parche de coordenadas sobre el que se define un sistema bidimensional de estas. En general, no es posible extender este parche a toda la superficie, por lo que será necesario definir múltiples parches que cubran colectivamente la superficie.

### A.1.3. Partición

Debido a que es imposible almacenar en el computador todo el conjunto de puntos que pueden constituir un objeto continuo, normalmente se opta por discretizar el espacio ocupado por este utilizando un conjunto de celdas y manteniendo información concerniente a cada una de ellas (Mantyla, 1998). El tipo de celdas utilizadas y la forma en que se combinan determinan las diferentes estrategias llevadas a cabo sobre esta idea. Sin embargo, las diferentes estrategias tienen un requisito en común: las celdas deben estar disjuntas o tener solo caras, bordes o vértices en común. Es decir, deben formar una cubierta sólida sin superposición espacial entre celdas o, lo que es lo mismo, una partición espacial.

**Definición 9.** Sea  $I$  un conjunto de índices de la forma  $I = \{1, \dots, n\}$ . Se dice que una familia de subconjuntos  $A_i \subseteq A \mid i \in I$  de un conjunto  $A$  es un *recubrimiento* de  $A$  si se cumplen las siguientes condiciones:

$$\begin{aligned} A_i &\neq \emptyset, \quad \forall i \in I \\ \bigcup_{i \in I} A_i &= A \end{aligned} \quad (\text{A.2})$$

**Definición 10.** Sea  $I$  un conjunto de índices de la forma  $I = \{1, \dots, n\}$ . Se dice que una familia de subconjuntos  $A_i \subseteq A \mid i \in I$  de un conjunto  $A$  es una *partición* de  $A$  si la familia es un recubrimiento y es cierto que:

$$A_i \cap A_j = \emptyset, \quad \forall i, j \in I, \quad i \neq j \quad (\text{A.3})$$

En otras palabras, las diferentes estrategias tienen en común que el espacio que representa el objeto se divide en celdas que no se superponen; por lo tanto, cualquier punto que pertenezca a él se puede identificar como dentro de una celda particular (o celdas, si se cae en la frontera de celdas vecinas).

#### A.1.4. Vecindad

La vecindad define la relación entre un píxel y su entorno (figura A.1).

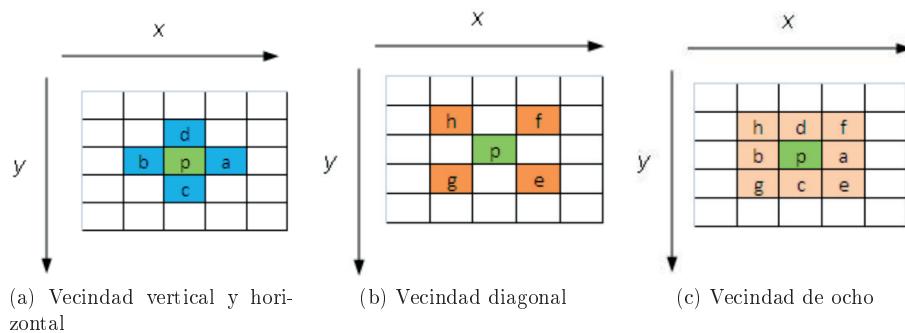


Figura A.1: Vecindades

El píxel  $p = (x, y)$  tiene dos vecinos verticales y dos vecinos horizontales, conocidos como vecindad-4  $V_4(p)$  o vecindad directa (figura A.1a):

$$V_4(p) = \begin{cases} a = & (x + 1, y) \\ b = & (x - 1, y) \\ c = & (x, y + 1) \\ d = & (x, y - 1) \end{cases} \quad (\text{A.4})$$

La vecindad diagonal (figura A.1b), denotada por  $V_d(p)$ , está definida por las siguientes coordenadas:

$$V_d(p) = \begin{cases} e = (x+1, y+1) \\ f = (x+1, y-1) \\ g = (x-1, y+1) \\ h = (x-1, y-1) \end{cases} \quad (\text{A.5})$$

La vecindad-8  $V_8(p)$  es la unión de las dos vecindades anteriores (figura A.1c). Naturalmente, existen las excepciones dadas cuando el píxel  $p = (x, y)$  es un punto del borde de la imagen, en cuyo caso algunos de los vecinos definidos anteriormente no existen.

### A.1.5. Conectividad

Un píxel  $p$  es contiguo a otro píxel  $q$  si están conectados. Sea  $C$  el conjunto de valores de intensidad de los píxeles que se permiten estar adyacentes. Si solo se desea que exista conectividad entre los píxeles con intensidades  $t_1$  y  $t_2$ , entonces  $C = \{t_1, t_2\}$ . Considérense dos tipos básicos de conectividad:

- Conectividad-4: dos píxeles  $p$  y  $q$  con valores de  $C$  están 4-conectados si  $q$  está en el conjunto  $V_4(p)$ .
- Conectividad-8: dos píxeles  $p$  y  $q$  con valores de  $C$  están 8-conectados si  $q$  está en el conjunto  $V_8(p)$ .

## A.2. Curvas y superficies

### A.2.1. Curvas implícitas en 2D

De forma intuitiva, una curva es un conjunto de puntos que puede ser dibujados en una hoja de papel sin levantar el lápiz. Una forma común de representar una curva es usando una ecuación implícita, que en dos dimensiones se expresa así:

$$f(x, y) = 0 \quad (\text{A.6})$$

La función  $f(x, y)$  retorna un valor real. Los puntos  $(x, y)$  donde este valor es 0 están en (pertenece a) la curva. Por ejemplo, digamos que:

$$f(x, y) = (x - x_c)^2 + (y - y_c)^2 - r^2 \quad (\text{A.7})$$

Donde  $(x_c, y_c)$  es un punto en 2D y  $r$  es un número real diferente de cero.

Si se toma  $f(x, y) = 0$ , los puntos donde se cumple esta función están en el círculo de radio  $r$  y centro  $(x_c, y_c)$ . La razón de que sea llamada *ecuación implícita* es que los puntos  $(x_c, y_c)$  en la curva no pueden ser calculados inmediatamente por la ecuación y, en cambio, debe ser determinados por la solución de esta. Así, los puntos en la curva no son generados explícitamente por la ecuación, sino que están ocultos implícitamente en algún lugar dentro de ella. Se debe notar que  $f(x, y)$  tiene valores para todos los puntos  $(x, y)$ .

El valor escalar devuelto por la función se puede interpretar como una distancia a la superficie del objeto descrita por aquella. Por ejemplo, la función implícita  $f(x, y) = x^2 + y^2 - 2^2$  define un disco de radio 2 centrado en el origen de las coordenadas. La figura A.2 representa una serie de valores escalares devueltos por esta función para diferentes puntos en un plano 2D.

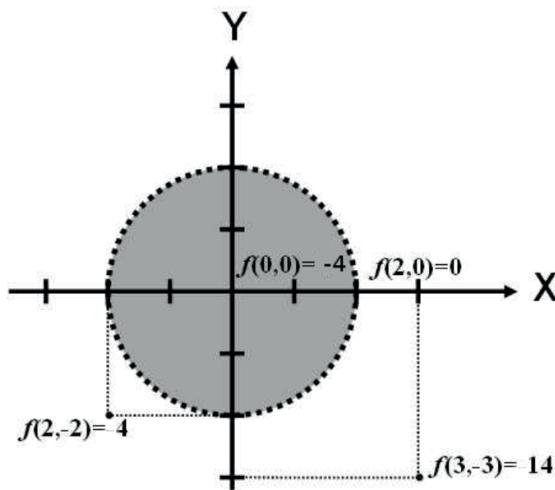


Figura A.2: Gráfico de la función implícita  $f(x, y) = x^2 + y^2 - 2^2$

El área gris es un subconjunto del plano 2D donde los valores devueltos por la función  $f(x, y)$  son negativos ( $f(0, 0) = -4$ ). El contorno punteado es un conjunto de puntos

donde la función  $f(x, y)$  devuelve 0 (el punto  $(2, 0)$  pertenece a la periferia de un círculo como  $f(2, 0) = 0$ ). Para cualquier otro punto en un plano 2D  $f(x, y)$  devuelve valores positivos ( $f(3, -3) = 14$ ). Por lo tanto, se puede observar que la curva parte el espacio en dos partes:  $f(x, y) \leq 0$  y  $f(x, y) > 0$ .

### A.2.2. Gradiente

Si se piensa la función  $f(x, y)$  como un campo escalar, por ejemplo un mapa de alturas donde altura =  $f(x, y)$ , entonces el vector gradiente indica la dirección en la cual el campo  $f(x, y)$  varía más rápidamente y su módulo representa el ritmo de variación de  $f(x, y)$  en la dirección de dicho vector gradiente.

**Definición 11.** El *gradiente* se define como el campo vectorial cuyas funciones coordenadas son las derivadas parciales del campo escalar y se representa con el operador diferencial nabla  $\nabla$  seguido de la función, esto es:

$$\nabla f(x, y) = \left( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) \quad (\text{A.8})$$

Se debe observar que el vector gradiente evaluado en un punto  $(x, y)$  de la curva implícita  $f(x, y) = 0$  es perpendicular al vector tangente de la curva en ese punto. Adicionalmente, como el gradiente apunta hacia afuera (arriba), él indica la dirección de la región  $f(x, y) > 0$ .

Como ejemplo, considérese el círculo implícito  $x^2 + y^2 - 1 = 0$  con el vector gradiente  $\nabla f(x, y) = (2x, 2y)$ , indicando que la parte externa del círculo es la región positiva de la función  $f(x, y) = x^2 + y^2 - 1$ . La longitud del gradiente puede variar dependiendo del multiplicador de la función implícita. El círculo de longitud 1, por ejemplo, puede ser descrito como  $Ax^2 + Ay^2 - A = 0$  para todo  $A$  diferente de cero. El gradiente de esta curva es  $(2Ax, 2Ay)$ , que es la normal perpendicular al círculo pero tendrá una longitud determinada por  $A$ . Para  $A > 0$ , la normal apunta hacia afuera del círculo y para  $A < 0$ , hacia adentro. De forma intuitiva, se puede pensar que la dirección del gradiente apunta hacia afuera y su magnitud mide qué tan afuera está el punto  $(x, y)$  en cuestión.

### A.2.3. Superficies en 3D

#### A.2.3.1. Superficie paramétrica

Una superficie paramétrica es una superficie cuyos límites están definidos por funciones paramétricas:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} f_x(u, v) \\ f_y(u, v) \\ f_z(u, v) \end{bmatrix} \quad (\text{A.9})$$

Las superficies paramétricas se basan en un mapeo desde un dominio planar (a trozos)  $D$  a  $\mathbb{R}^3$ , es decir,  $f : D \rightarrow \mathbb{R}^3$ . El número de mapeos necesarios para representar con precisión una superficie se correlaciona estrechamente con la complejidad de su superficie. Dado que los puntos en la superficie pueden obtenerse evaluando la función inversa  $f^{-1} : \mathbb{R}^3 \rightarrow D$ , las superficies paramétricas pueden renderizarse de manera bastante eficiente.

#### A.2.3.2. Superficie implícita

Así como las ecuaciones implícitas pueden ser usadas para definir curvas en dos dimensiones, también pueden usarse para definir superficies en tres dimensiones de la siguiente forma:

$$f(x, y, z) = d \quad (\text{A.10})$$

Donde  $d$  se puede considerar como la densidad de un campo escalar 3D en el punto  $(x, y, z)$ . Usualmente la ecuación es de la forma  $f(x, y, z) = 0$ .

Mediante la evaluación de  $f$  se puede determinar si un punto  $(x, y, z)$  está en la superficie o no. De igual forma se puede determinar de qué lado de la superficie está el punto solo con observar el signo de  $f$ . Si cada punto es representado por el vector  $\mathbf{p} = (x, y, z)$ , entonces la función implícita se puede escribir como  $f(\mathbf{p}) = 0$ .

**Definición 12.** Dada una función  $f : \Omega \mapsto \mathbb{R}$ , donde  $\Omega$  es un subconjunto abierto de  $\mathbb{R}^3$ , el conjunto  $S$  es una *superficie en forma implícita* si se cumple que:

$$S = \{\mathbf{p} = (x, y, z) \in \Omega : f(\mathbf{p}) = 0\} \quad (\text{A.11})$$

#### A.2.4. Vector normal a la superficie

Cada punto  $\mathbf{p}$  en la superficie  $S$  puede tener un vector normal  $\mathbf{n}$  diferente.

**Definición 13.** Un vector se dice *normal a una superficie* en un punto  $\mathbf{p}$  si es perpendicular al plano tangente en dicho punto de la superficie. El vector normal a la superficie en el punto  $\mathbf{p}$  es dado por el gradiente de la función implícita:

$$\mathbf{n} = \nabla f(\mathbf{p}) = \left( \frac{\partial f(\mathbf{p})}{\partial x}, \frac{\partial f(\mathbf{p})}{\partial y}, \frac{\partial f(\mathbf{p})}{\partial z} \right) \quad (\text{A.12})$$

Al igual que en 2D, el gradiente en 3D apunta en la dirección de incremento más rápido de  $f$  en la superficie, donde  $f(\mathbf{p}) > 0$ , siendo perpendicular a la dirección de la tangente en el punto  $\mathbf{p}$ . Para una función implícita, la *normal* se define como el gradiente de longitud uno en un punto, mientras que para las superficies paramétricas esta se calcula generalmente como el producto cruz de las tangentes de superficie en las dos direcciones paramétricas.

### A.3. Reconstrucción tomográfica

La comprensión de los conceptos relacionados con la reconstrucción tomográfica requiere de la revisión de algunos conceptos matemáticos claves, como la transformada de Radon (subsección A.3.1), que permite crear imágenes de objetos muy similares a las que ocurren en las radiografías. El problema inverso permite convertir las transformaciones de Radon en coeficientes de atenuación utilizando la transformada de Radon inversa. Se revisarán dos métodos para reconstruir el objeto a partir de un escaneo TAC: el teorema de cortes y la retroproyección, ambos con base en la transformada de Fourier (subsección A.3.2).

#### A.3.1. La transformada de Radon

En 1917 Johann Radon (1887-1956) demostró matemáticamente que se puede reconstruir un objeto bidimensional o tridimensional si se conocen todas sus proyecciones (Hounsfield, 1980; Deans, 2007). Sin embargo, sus trabajos permanecieron en el olvido aproximadamente cincuenta años, hasta que Cormack y Hounsfield usaron sus resultados para crear de manera independiente las bases de la tomografía computarizada.

Si se denota como  $f(x, y)$  al mapa de atenuación del objeto que se quiere reconstruir y  $f(p, \theta)$  la proyección de  $f(x, y)$  a un ángulo  $\theta$ , entonces se tiene que  $p = x \cos \theta + s \sin \theta$ . De donde resulta:

$$f(p, \theta) = \int_{-\infty}^{\infty} f(x, y) ds \quad (\text{A.13})$$

**Definición 14.** Dada una función  $f(t, \theta)$  definida en  $\mathbb{R}^2$  con soporte compacto, la transformada de Radon de  $f(p, \theta)$  en dos dimensiones denotada por  $\mathcal{R}_2 f(p, \theta)$  es definida como:

$$\mathcal{R}_2\{f(p, \theta)\} = \iint_{\mathbb{R}^2} f(x, y) \delta(p - x \cos \theta - y \sin \theta) dx dy \quad (\text{A.14})$$

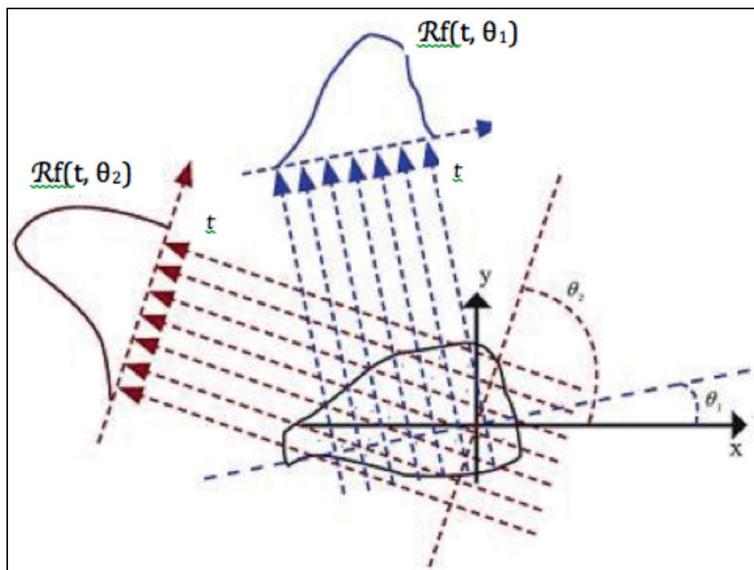
Donde  $\delta$  es la función delta de Dirac,  $p \in \mathbb{R}$  y  $\theta \in [0, 2\pi]$ .

Una función con soporte compacto es aquella que toma el valor cero en todas partes fuera de un conjunto compacto. Este es un requisito razonable para un problema de imágenes médicas porque solo se está tratando con áreas finitas (o cortes) de algún objeto. Recuérdese que el objetivo es determinar el coeficiente de atenuación del objeto que está relacionado con la densidad del propio objeto. Por lo tanto, dado que solo se está tratando con cortes finitos, habrá alguna región finita fuera de la cual el coeficiente de atenuación debe ser igual a cero.

La transformada de Radon ofrece un medio para determinar la densidad total de una determinada función  $f$  a lo largo de una línea dada  $L$ . Esta línea  $L$  está determinada por un ángulo  $\theta$  desde el eje  $x$  y una distancia  $t$  desde el origen, como se describe en la ecuación A.14. Como se observa en la figura A.3, si se toma la transformada de Radon a lo largo de múltiples líneas en ángulos variables  $\theta_1$  y  $\theta_2$ , se podrán determinar múltiples funciones de densidad para el objeto estudiado. Intuitivamente se puede interpretar la transformada de Radon como una versión borrosa del objeto inicial.

Supóngase que la región representada en la figura A.3 fuera una mancha de tinta; si se hubiera borrado esta mancha a lo largo de líneas variables en la dirección  $\theta_1$ , se podría esperar que las regiones más anchas corresponderían a una región más grande vs. las más pequeñas, que es exactamente lo que se ve.

Las proyecciones tomográficas pueden expresarse como la transformada de Radon del mapa de atenuaciones del objeto que se desea reconstruir, con lo cual el problema de la reconstrucción es equivalente a encontrar la inversa de la transformada de Radon que permita recuperar nuestra función inicial  $f$ .

Figura A.3: Transformada de Radon de  $\theta_1$  y  $\theta_2$ 

Además, se define el dominio natural de la transformada de Radon como el conjunto de funciones  $f$  sobre  $\mathbb{R}^2$ , tal que:

$$\int_{-\infty}^{\infty} |f(x(s), y(s))| ds < \infty \quad (\text{A.15})$$

Es decir, se restringe el dominio al conjunto de funciones absolutamente integrables.

La transformada de Radon está estrechamente relacionada con la transformada de Fourier, un método ampliamente estudiado cuya inversa está bien descrita por el teorema central del corte. A continuación se introducirá la transformada de Fourier antes de explorar esta relación más adelante.

### A.3.2. La transformada de Fourier

**Definición 15.** Dada una función absolutamente integrable  $f(x)$  en  $\mathbb{R}$ , la transformada de Fourier de  $f(x)$  es definida para cada numero real  $\xi$  como:

$$\mathcal{F}f(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx \quad (\text{A.16})$$

La transformada de Fourier se utiliza frecuentemente en el análisis de señales y ofrece un medio para cambiar una función del tiempo en una función de frecuencia. La variable  $x$  representa el tiempo en segundos y la variable  $\xi$ , la frecuencia de la función en Hertz. La transformada de Fourier de la ecuación A.16 es:

$$\mathcal{F}\{f(p, \theta)\} = \int_{-\infty}^{\infty} f(p, \theta) e^{-2\pi i t p} dp \quad (\text{A.17})$$

Si  $f = f(x, y)$ , entonces  $\mathcal{F}_2\{f\} = \mathcal{F}\{\mathcal{R}(f)\}$ . En efecto, desglosando el lado derecho de la igualdad anterior, se observa que:

$$\begin{aligned} \mathcal{F}\{\mathcal{R}(f)\} &= \int_{-\infty}^{\infty} \mathcal{R}(f) e^{-2\pi i t p} dp \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(p - x \cos \theta - y \sin \theta) dx dy e^{-2\pi i t p} dp \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i t (x \cos \theta + y \sin \theta)} dx dy \end{aligned} \quad (\text{A.18})$$

Haciendo  $k_x = k \cos \theta$ ,  $k_y = k \sin \theta$ , resulta:

$$\mathcal{F}\{\mathcal{R}(f)\} = \mathcal{F}_2\{f\} \quad (\text{A.19})$$

Este resultado es conocido como el teorema de cortes de Fourier, que establece que la transformada unidimensional de Fourier de una proyección  $f(p, \theta)$  de un objeto es igual a la transformada bidimensional de Fourier de la función  $f(x, y)$  evaluada en la recta de esta proyección. Esta idea se demuestra en la figura A.4. En la parte izquierda se empieza con una función  $F(x, y)$  y se aplica luego la transformada de Radon para obtener una proyección  $g$ , a la que luego se aplica la transformada de Fourier, recibiendo la función naranja  $G(w)$  en la parte superior de la figura A.4. En la parte derecha se toma la transformada de Fourier bidimensional de la porción naranja de  $F(x, y)$  paralela a la línea de proyección. Esta transformada de Fourier bidimensional también es igual a la función  $G(w)$ , representada en color naranja.

La transformada inversa de Fourier de  $\mathcal{F}_2\{f\}$  da como resultado la reconstrucción completa de  $f(x, y)$ , es decir:

$$f(x, y) = \mathcal{F}_2^{-1}\{\mathcal{F}\{\mathcal{R}(f)\}\} \quad (\text{A.20})$$

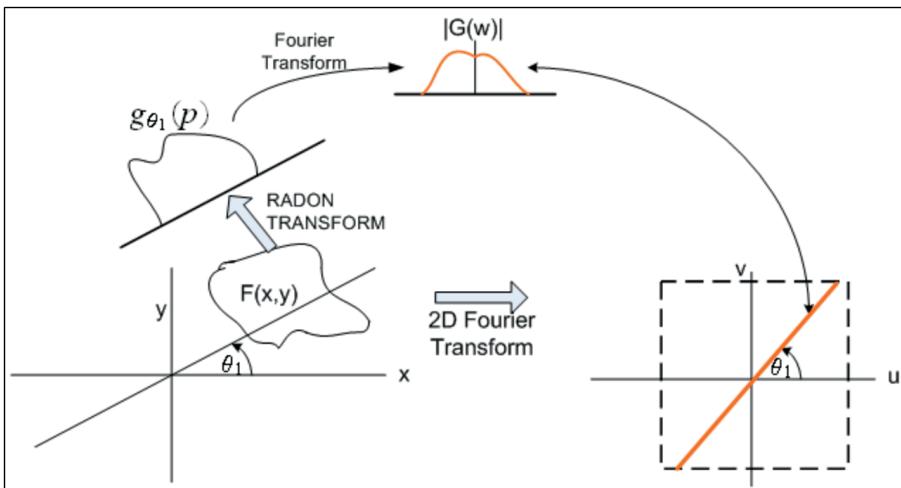


Figura A.4: Teorema del corte central

Como se ha visto, es posible recuperar a la función  $f$  usando el teorema de cortes de Fourier; sin embargo, implementar este método resulta muy costoso en cuestiones computacionales debido a la gran cantidad de proyecciones que se realizan. Para esto, se han investigado diferentes técnicas de implementación de este tipo de reconstrucción. Entre ellas destaca el algoritmo de retroproyección filtrada, que transforma las proyecciones simples en proyecciones filtradas aplicando un filtro, con el fin de obtener resultados más nítidos y con menos factores de ruido.

Como su nombre indica, hay dos pasos dentro del algoritmo de retroproyección filtrada: la parte de filtrado, que puede visualizarse como una ponderación de cada proyección en el dominio de frecuencia, y la parte de retroproyección, que es equivalente a encontrar las reconstrucciones correspondientes a cada una de las proyecciones filtradas. A partir de la siguiente relación:

$$\mathcal{R}(k_x, k_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i(xk_x + yk_y)} dx dy \quad (\text{A.21})$$

Se tiene que:

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathcal{R}(k_x, k_y) e^{2\pi i(xk_x + yk_y)} dk_x dk_y \quad (\text{A.22})$$

Pasando de coordenadas rectangulares a coordenadas polares mediante  $k_x = k \cos \theta$ ,  $k = k \sin \theta$ ,  $dk_x dk_y = k dk dk \theta$ , se puede reescribir la ecuación A.22 como:

$$f(x, y) = \int_0^{2\pi} \int_0^{\infty} \mathcal{R}(k, \theta) e^{2\pi i k(x \cos \theta + y \sin \theta)} k dk d\theta \quad (\text{A.23})$$

La ecuación A.23 puede separarse en dos sumandos considerando que  $\theta$  puede variar de 0 a  $\pi$  y de  $\pi$  a  $2\pi$ . Así, se tiene:

$$f(x, y) = \int_0^{2\pi} \int_0^{\infty} \mathcal{R}(k, \theta) e^{2\pi i k(x \cos \theta + y \sin \theta)} k dk d\theta + \int_0^{2\pi} \int_0^{\infty} \mathcal{R}(k, \theta + \pi) e^{2\pi i k(x \cos(\theta + \pi) + y \sin(\theta + \pi))} k dk d\theta \quad (\text{A.24})$$

Y usando la siguiente propiedad:

$$\mathcal{R}(k, \theta + \pi) = \mathcal{R}(-k, \theta) \quad (\text{A.25})$$

La expresión A.23 para  $f(x, y)$  puede expresarse como:

$$f(x, y) = \int_0^{\pi} \left[ \int_{-\infty}^{\infty} \mathcal{R}(k, \theta) |k| e^{2\pi i kp} dk \right] d\theta \quad (\text{A.26})$$

Donde se ha simplificado la expresión A.23 usando  $p = x \cos \theta + s \sin \theta$ . Ahora, se denota  $S_{\theta}(k)$  así:

$$S_{\theta}(k) = \mathcal{F}\{\mathcal{R}(f)\} = \mathcal{F}_2\{f\} \quad (\text{A.27})$$

Con lo que la expresión A.26 queda como:

$$f(x, y) = \int_0^{\pi} Q_{\theta}(x \cos \theta + s \sin \theta) d\theta \quad (\text{A.28})$$

Donde:

$$Q_{\theta}(p) = \int_{-\infty}^{\infty} S_{\theta}(k) |k| e^{2\pi i kp} dk \quad (\text{A.29})$$

La ecuación A.29 representa una operación de filtrado (convolución), donde la respuesta en frecuencia del filtro está dada por  $|k|$ ; por lo tanto,  $Q_{\theta}(p)$  es llamada una *proyección filtrada*. Las proyecciones resultantes para distintos ángulos  $\theta$  son superpuestas para formar la estimación de  $f(x, y)$ .

### A.3.3. Coeficiente de atenuación

La absorción de rayos X en un medio dado va a depender de la distancia que estos atraviesan y de su intensidad (Prince y Links, 2006). Considérese un rayo de intensidad  $I_0$  que penetra un objeto a lo largo de una trayectoria  $L$  en línea recta, con una distribución no homogénea de atenuaciones  $\mu(x)$  (figura A.5). La intensidad del rayo que alcanza el detector  $I(x)$  depende no solo de la distancia atravesada  $x$ , sino también de la atenuación  $\mu(x)$  de cada punto en su trayectoria. De acuerdo con la ley de Beer-Lambert:

$$I(x) = I_0 e^{-\mu(x)} \quad (\text{A.30})$$

Donde  $\mu(x)$  es el coeficiente de atenuación de masa característico del material e independiente del estado físico. El coeficiente de atenuación de masa está dado por el coeficiente lineal de atenuación y la densidad del material.

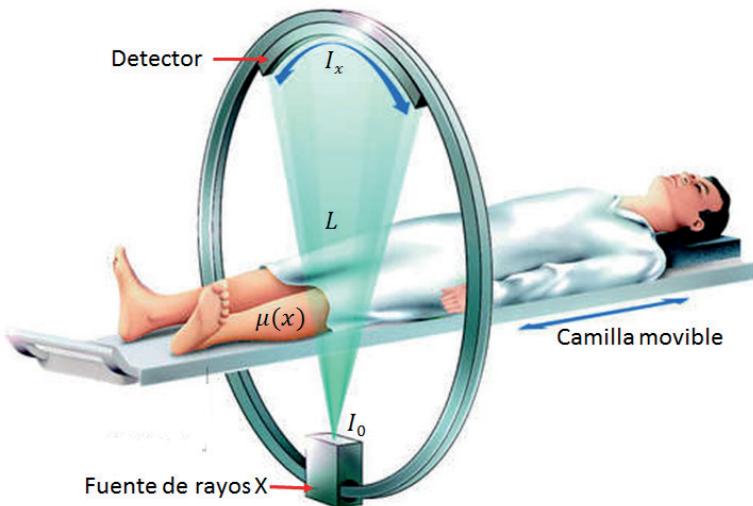


Figura A.5: Adquisición de las proyecciones tomográficas

Un objeto en el espacio tridimensional está determinado por una función de densidad  $f$  en el espacio  $\mathbb{R}^3$ , siendo  $f(x) = I_x$  la densidad en el punto  $x$ . Una imagen de rayos X tomada en la dirección  $\theta$  proporciona una función  $L_\theta f$  en el plano ortogonal a  $\theta$ , cuyo valor en un punto  $x$  es la masa total a lo largo de la recta a través de  $x$  en la dirección  $\theta$ , dado por:

$$L_\theta f = \int_{-\infty}^{\infty} f(x + t\theta) dt, \quad x \in \theta^\perp \quad (\text{A.31})$$

Técnicamente  $L_\theta f$  es la imagen o radiografía de  $f$  (Solomon, 1976). La imagen de rayos X se basa en este principio: un objeto absorberá o dispersará rayos X de una energía particular de una manera dependiente de su composición, cuantificada por el coeficiente de atenuación  $\mu(x)$ . El coeficiente de atenuación  $\mu(x)$  de una sustancia es una función en  $\mathbb{R}^3$  dependiente de una variedad de factores, pero principalmente refleja la densidad de electrones.

Por consiguiente, las sustancias más densas y las sustancias que contienen elementos con muchos electrones tendrán coeficientes de atenuación  $\mu(x)$  más altos. Esto ayuda a explicar por qué el hueso, que contiene altos porcentajes de calcio (20 electrones), potasio (19 electrones), fósforo (15 electrones) y magnesio (12 electrones), tiene un coeficiente de atenuación mucho mayor que el tejido blando, que está compuesto principalmente de carbono (6 electrones), nitrógeno (7 electrones) y oxígeno (8 electrones) (Epstein, 2007).

#### A.3.4. Adquisición de las proyecciones tomográficas

Para capturar cada proyección tomográfica, el tomógrafo realiza dos mediciones:  $I_0$ , la intensidad inicial de cada haz de rayos X en la fuente de radiación, y  $I(x)$ , la intensidad final de cada haz en el detector de radiación (figura A.5). Los cambios de intensidad para un solo haz dependen de la densidad interna del objeto a lo largo de la línea  $L$  por la que atraviesa el rayo X. Al cambiar la orientación de la fuente y el detector en una geometría circular, es posible obtener más información sobre la densidad interna mirando los cambios de intensidad correspondientes de un corte.

Es posible medir tanto  $I_0$  como la intensidad de salida  $I(x)$  en el detector del tomógrafo mediante la siguiente ecuación:

$$I(x) = I_0 e^{-\int_L \mu(x) dx} \quad (\text{A.32})$$

Sin embargo, resulta más conveniente escribirla como:

$$p(x) = -\ln \left( \frac{I(x)}{I_o} \right) = \int_L \mu(x) dx \quad (\text{A.33})$$

La ecuación A.33 constituye una integral de línea o transformada de Radon de los coeficientes de atenuación lineal a través del recorrido de los rayos X. La proyección  $p(x)$ , formada por las integrales de línea paralelas o en abanico, tiene implicaciones importantes. La primera, que el detector registra la integral de línea y esta depende de las atenuaciones en cada región del objeto en la trayectoria del rayo. La segunda, que aunque se usa información volumétrica, el detector registra la proyección  $p(x)$ , que es una señal unidimensional para cada ángulo  $\theta$ .

Cada corte axial se puede dividir en una cuadrícula de píxeles, o elementos de imagen que representan una pequeña parte de la imagen global. La imagen resultante se subdivide en cierta cantidad de elementos, llamados *píxeles*, para poder ser almacenada en forma de matriz (figura 2.2a). Alternativamente, los píxeles se llaman *vóxeles* porque representan volúmenes unitarios. Los tamaños habituales son 256 x 256, o 512 x 512.

Por convención, un corte se ve como si estuviera mirando desde la parte inferior del objeto (los pies en el caso de una persona) hacia la parte superior (la cabeza de una persona), por lo que la imagen TAC especifica el lado izquierdo  $L$  del objeto y el lado derecho  $R$ .

## A.4. Estructuras de datos espaciales

### A.4.1. Arreglo de etiquetas

Una de las primeras estructura de datos utilizada por los algoritmos de segmentación es el arreglo (*array*) de etiquetas (Suk y Chung, 1983; Nicol, 1995), que consiste en asociar a cada píxel de la imagen una etiqueta tal que todos los píxeles que la comparten pertenezcan a una misma región. Esta estructura de datos está bien adaptada a las operaciones que se aplican a toda la imagen, como son los operadores morfológicos.

Las fronteras de las regiones se definen implícitamente como los lados compartidos por píxeles con etiquetas diferentes. Esto hace que características vinculadas a la frontera de las regiones, como segmentos fronterizos o regiones vecinas, requieran una reconstrucción explícita de esta. Es posible codificar explícitamente la frontera, por ejemplo almacenando para cada región la secuencia de píxeles adyacente que no pertenece a ella. Tal frontera puede ser representada por uno de sus puntos y una secuencia de movimientos (figura A.6).

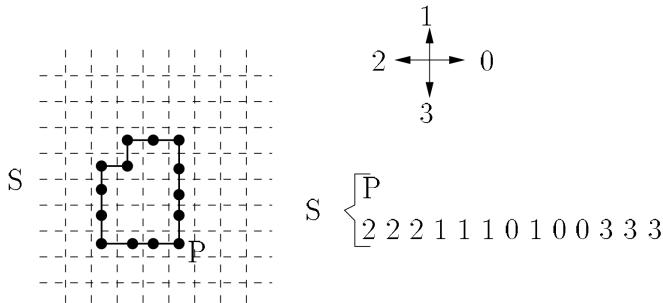


Figura A.6: Definición de una región en 2D con un camino 4-conectado

Un inconveniente importante de las fronteras definidas como secuencia de píxeles es que no son topológicamente consistentes. Además, dos regiones adyacentes no comparten elementos de frontera.

#### A.4.2. Estructuras jerárquicas

En algunos casos puede ser interesante procesar una imagen a diferentes niveles de resolución. Comenzando en un nivel aproximado de resolución, la partición de la imagen inicial puede refinarse de nivel a nivel, hasta el nivel de resolución de la imagen original. Las matriz-pirámides (Browning y Tanimoto, 1982; Tanimoto, 2014) o M-pirámides proporcionan una descripción de dicha multiresolución.

**Definición 16.** Una *M-pirámide*  $P$  es una secuencia  $\{M(n), M(n-1), \dots, M(0)\}$  de matrices, donde  $M(n)$  representa la imagen original,  $M(n-1)$  una versión de  $M(n)$  a la mitad de la resolución, y así hasta  $M(0)$ , que contiene un solo píxel.

Si un algoritmo particular tiene que acceder con frecuencia a varios niveles de resolución, puede ser más fácil definir las pirámides en términos de árboles en lugar de matrices. Los árbol-pirámides corresponden a la versión de árbol de M-pirámides. Su definición formal puede expresarse como sigue:

**Definición 17.** Un *árbol-pirámide*  $P$  es un conjunto de p-nodos  $(k, i, j)$  con  $0 \leq k \leq n$ ,  $0 \leq i \leq 2^n - 1$ ,  $0 \leq j \leq 2^n - 1$ . Cada p-nodo corresponde a un píxel de una M-pirámide: el p-nodo  $(k, i, j)$  es el píxel  $(i, j)$  de la imagen  $M(k)$  en la M-pirámide  $\{M(n), M(n-1), \dots, M(0)\}$ . El padre de un p-nodo  $(k, i, j)$  es igual a  $(k-1, [i/2], [j/2])$ , donde  $[x]$  denota el mayor entero inferior o igual a  $x$ . Cada hoja de un árbol-pirámide representa un píxel de la imagen original.

Si se etiqueta cada hoja con una etiqueta de región, todos los hijos de un nodo pueden compartir la misma etiqueta y ser representados por su padre sin pérdida de información. Así, se puede podar el árbol-pirámide hasta que ningún nodo tenga hijos con la misma etiqueta. El árbol-pirámide no es entonces más equilibrado y la estructura resultante se llama *quadtree* o *árbol cuádruple* (Rosenfeld, 1976; Samet, 1980; Dyer *et al.*, 1980) (figura A.7).

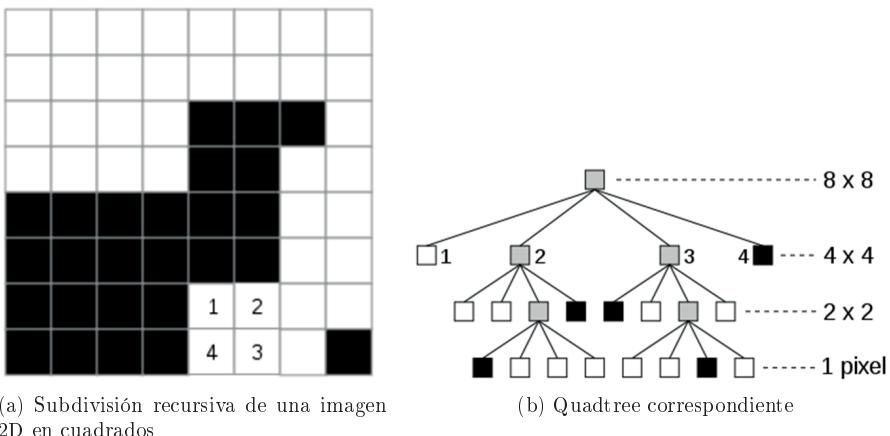
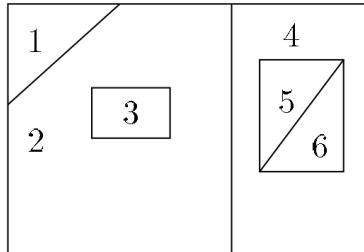


Figura A.7: Generación de un árbol cuádruple (quadtree)

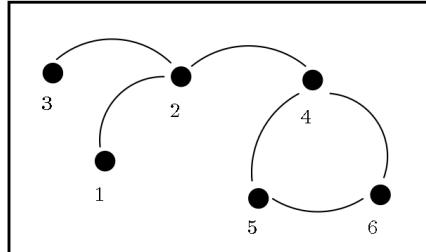
Cuando se trabaja en un espacio tridimensional, se utiliza un octree o árbol octal, que es una estructura de datos en la que cada nodo interno tiene exactamente ocho hijos; lo que permite particionar el espacio recursivamente en ocho octantes.

#### A.4.3. Grafo de adyacencia de regiones

Las estructuras jerárquicas (subsección A.4.2) están dedicadas a algoritmos de división basados en regiones y no proporcionan implementaciones eficientes para algoritmos de fusión. De hecho, calcular la adyacencia de las regiones con una estructura de árbol puede implicar un procesamiento complejo. Además, la fusión de dos regiones que comparten diferentes padres rompe la estructura del árbol. Una estructura de datos habitual para realizar operaciones de fusión es el grafo de adyacencia de regiones GAR (Region Adjacency Graph).



(a) Imagen segmentada en seis regiones

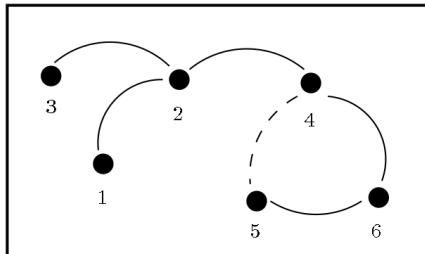


(b) Grafo de adyacencia del conjunto de regiones de la figura A.8a

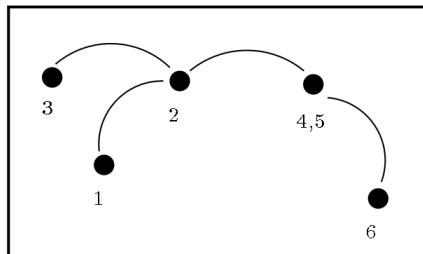
Figura A.8: Generación de un grafo de adyacencia de regiones

La figura A.8a muestra una imagen 2D segmentada en seis regiones. Cada nodo del grafo de adyacencia representa una región de la imagen segmentada, y dos regiones adyacentes comparten una arista en el grafo (figura A.8b).

La fusión de dos regiones  $r_1$  y  $r_2$  implica actualizar los nodos correspondientes en el GAR al contraer la arista que enlaza sus nodos. Ambos nodos  $r_1$  y  $r_2$  se fusionan, y el nodo resultante se enlaza a cada nodo de una región adyacente a  $r_1$  o  $r_2$ . Luego se eliminan las aristas múltiples, que son aquellas que unen los mismos dos nodos. La figura A.9a muestra las regiones 4 y 5 que serán fusionadas. La figura A.9b muestra el grafo de adyacencia actualizado.



(a) Regiones 4 y 5 que serán fusionadas



(b) Grafo de adyacencia actualizado

Figura A.9: Fusión de regiones en un grafo de adyacencia

El grafo de adyacencia de regiones se puede implementar como plexos o una lista de adyacencia o un medio arreglo de bits. Cuando se implementa como un medio arreglo, la fusión de dos nodos  $i$  y  $j$  requiere invalidar la línea y la columna  $i$  y actualizar la línea y la columna  $j$ . El grafo de adyacencia de regiones se puede usar junto con un

arreglo de sus descripciones. Cada entrada de este arreglo puede contener parámetros de una región, los cuales se actualizan durante fusiones sucesivas y se pueden utilizar para etiquetar las aristas del GAR. Por ejemplo, este arreglo puede contener el color medio de las regiones. Así, en cada paso del algoritmo de fusión se pueden fusionar las dos regiones  $r_1$  y  $r_2$  de manera que:

- Las regiones  $r_1$  y  $r_2$  son adyacentes en el GAR.
- La distancia media de color entre  $r_1$  y  $r_2$  es mínima.

Parámetros tales como la media, la varianza o el histograma pueden actualizarse eficazmente durante sucesivas operaciones de fusión. Por lo tanto, los grafos de adyacencia de regiones dan una descripción eficaz de las regiones adyacentes. Además, las relaciones de adyacencia y la mayoría de los parámetros de las regiones pueden actualizarse eficazmente durante sucesivas operaciones de fusión. Sin embargo, la división de una región  $r$  en un conjunto de regiones  $r_1, r_2, \dots, r_n$  implica:

- Suprimir la región  $r$  en el GAR.
- Recorrer las regiones  $r_1, r_2, \dots, r_n$  para insertar los nodos correspondientes y las relaciones de adyacencia en el GAR.

En consecuencia, la actualización de la estructura GAR después de una operación de división involucra demasiados cálculos. Por esto, la estructura de datos GAR se utiliza preferiblemente en algoritmos de fusión.

Todas las estructuras estudiadas se dedican principalmente a un tipo de operación de los datos. Los arreglos de etiquetas y las estructuras jerárquicas se dedican a operaciones de división, mientras que la estructura de datos de cadenas, la transformada de eje medio y el grafo de adyacencia de regiones se dedican a operaciones de fusión. Además, cada estructura es conveniente para calcular un subconjunto de parámetros, pero puede inducir un procesamiento complejo para otros casos. Por ejemplo, el cálculo de la frontera de una región dada usando una estructura de datos quadtree requeriría muchos cálculos.

Por otra parte, las estructuras de datos presentadas en la subsección 3.3.1 se pueden utilizar para operaciones de división y de fusión. Además, estas estructuras permiten cálculos eficientes de los parámetros requeridos por los algoritmos de segmentación.



## Bibliografía

---

- Amenta, N., Choi, S., Dey, T. K., y Leekha, N. (2000). A simple algorithm for homeomorphic surface reconstruction. En *Proceedings of the sixteenth annual symposium on Computational geometry*, pp. 213–222. Nueva York: ACM.
- Amenta, N., Choi, S., y Kolluri, R. K. (2001). The power crust. En *Proceedings of the sixth ACM symposium on Solid modeling and applications*, pp. 249–266. Nueva York: ACM.
- Anderson, J. C., Garth, C., Duchaineau, M. A., y Joy, K. I. (2010). Smooth, volume-accurate material interface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 16(5):802–814.
- Avila, L. S., Barre, S., Blue, R., Geveci, B., Henderson, A., Hoffman, W. A., King, B., Law, C. C., Martin, K. M., y Schroeder, W. J. (2010). *The VTK User's Guide*. Nueva York: Kitware.
- Ayachit, U. (2015). *The paraview guide: a parallel visualization application*. Nueva York: Kitware.
- Bankman, I. (2008). *Handbook of medical image processing and analysis*. Nueva York: Academic Press.
- Barbour, L. J. (2001). X-seed: A software tool for supramolecular crystallography. *Journal of Supramolecular Chemistry*, 1(4):189 – 191.
- Berg, M., Cheong, O., Kreveld, M., y Overmars, M. (2008). *Computational Geometry*. Berlín: Springer.
- Bloomenthal, J. y Ferguson, K. (1995). Polygonization of non-manifold implicit surfaces. En *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 309–316. Nueva York: ACM.

- Boltcheva, D., Yvinec, M., y Boissonnat, J.-D. (2009). Feature preserving delaunay mesh generation from 3d multi-material images. En *Computer Graphics Forum*, volumen 28, pp. 1455–1464. Wiley Online Library.
- Braid, I. C. (1974). Designing with volumes. *Cambridge: Cantab Press.*
- Browning, J. D. y Tanimoto, S. L. (1982). Segmentation of pictures into regions with a tile-by-tile method. *Pattern Recognition*, 15(1):1–10.
- Carr, J. C., Beatson, R. K., Cherrie, J. B., Mitchell, T. J., Fright, W. R., McCallum, B. C., y Evans, T. R. (2001). Reconstruction and representation of 3d objects with radial basis functions. En *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp. 67–76. Nueva York: ACM.
- Chen, S., Zhang, J., Zhang, H., Guan, Q., Du, Y., Yao, C., y Zhang, J. (2010). Myocardial motion analysis for determination of tei-index of human heart. *Sensors*, 10(12):11428–11439.
- Cohen, E., Riesenfeld, R. F., y Elber, G. (2001). *Geometric modeling with splines: an introduction*. Natick: AK Peters/CRC Press.
- Csikós, B. (2006). A schläfli-type formula for polytopes with curved faces and its application to the kneser-poulsen conjecture. *Monatshefte für Mathematik*, 147(4):273–292.
- De, S., Guilak, F., y Mofrad, M. R. (2010). *Computational modeling in biomechanics*. Berlín: Springer.
- Deans, S. R. (2007). *The Radon transform and some of its applications*. North Chelmsford: Courier Corporation.
- Dey, T. K. y Giesen, J. (2003). Detecting undersampling in surface reconstruction. En *Discrete and Computational Geometry*, pp. 329–345. Berlín: Springer.
- Dey, T. K., Janoos, F., y Levine, J. A. (2012). Meshing interfaces of multi-label data with delaunay refinement. *Engineering with Computers*, 28(1):71–82.
- Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.
- d'Otreppe, V., Boman, R., y Ponthot, J.-P. (2012). Generating smooth surface meshes from multi-region medical images. *International Journal for Numerical Methods in Biomedical Engineering*, 28(6-7):642–660.

- Dougherty, G. (2009). *Digital image processing for medical applications*. Cambridge: Cambridge University Press.
- Dyer, C. R., Rosenfeld, A., y Samet, H. (1980). Region representation: Boundary codes from quadtrees. *Communications of the ACM*, 23(3):171–179.
- Edelsbrunner, H. y Mücke, E. P. (1994). Three-dimensional alpha shapes. *ACM Transactions on Graphics (TOG)*, 13(1):43–72.
- Epstein, C. L. (2007). *Introduction to the mathematics of medical imaging*. Filadélfia: SIAM.
- Fabri, A. y Pion, S. (2009). Cgal: The computational geometry algorithms library. En *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*, pp. 538–539. Nueva York: ACM.
- Faridani, A. (2003). Introduction to the mathematics of computed tomography. *Inside Out: Inverse Problems and Applications*, 47:1–46.
- Farin, G. E. y Farin, G. (2002). *Curves and surfaces for CAGD: a practical guide*. San Francisco: Morgan Kaufmann.
- Field, D. A. (1988). Laplacian smoothing and delaunay triangulations. *International Journal for Numerical Methods in Biomedical Engineering*, 4(6):709–712.
- Fram, J. R. y Deutsch, E. S. (1975). On the quantitative evaluation of edge detection schemes and their comparison with human performance. *Computers, IEEE Transactions on*, 100(6):616–628.
- González, R. C. y Woods, R. E. (2002). *Digital image processing, vol. 2*. Nueva Jersey: Prentice-Hall.
- Hartmann, E. (1999). On the curvature of curves and surfaces defined by normal forms. *Computer Aided Geometric Design*, 16(5):355–376.
- Hausdorff, F. (2008). *Deskripte Mengenlehre und Topologie*, volumen 3. Berlín: Springer-Verlag.
- He, Y. y Qin, H. (2004). Surface reconstruction with triangular b-splines. En *Geometric Modeling and Processing, 2004. Proceedings*, pp. 279–287. Nueva Jersey: IEEE.

- Hege, H.-C., Seebass, M., Stalling, D., y Zockler, M. (1997). A generalized marching cubes algorithm based on non-binary classifications. *Berlín: Konrad-Zuse-Zentrum für Informationstechnik*, (SC-97-05).
- Heimann, T., Van Ginneken, B., y Styner, M. A. (2009). Comparison and evaluation of methods for liver segmentation from ct datasets. *IEEE transactions on medical imaging*, 28(8):1251–1265.
- Hoffmann, M. (2005). Numerical control of kohonen neural network for scattered data approximation. *Numerical Algorithms*, 39(1-3):175–186.
- Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., y Stuetzle, W. (1992). *Surface reconstruction from unorganized points*, volumen 26. ACM.
- Hounsfield, G. N. (1980). Computed medical imaging. *Medical physics*, 7(4):283–290.
- Hu, G. (2009). Survey of recent volumetric medical image segmentation techniques. En *Biomedical Engineering*. Londres: InTech.
- Jamin, C., Alliez, P., Yvinec, M., y Boissonnat, J.-D. (2015). Cgalmesh: a generic framework for delaunay mesh generation. *ACM Transactions on Mathematical Software (TOMS)*, 41(4):23.
- John, N. y McCloy, R. (2004). Navigating and visualizing three-dimensional data sets. *The British journal of radiology*, 77(suppl\_2):S108–S113.
- Johnson, H. J., McCormick, M. M., e Ibanez, L. (2015). *The ITK Software Guide Book: Design and Functionality*, volumen 4. Nueva York: Kitware.
- Ju, T., Losasso, F., Schaefer, S., y Warren, J. (2002). Dual contouring of hermite data. En *ACM transactions on graphics (TOG)*, volumen 21, pp. 339–346. ACM.
- Kaufman, A., Cohen, D., y Yagel, R. (1993). Volume graphics. *Computer*, 26(7):51–64.
- Kazhdan, M., Bolitho, M., y Hoppe, H. (2006). Poisson surface reconstruction. En *Proceedings of the fourth Eurographics symposium on Geometry processing*, volumen 7. Aire-la-Vile: The Eurographics Association.
- Kazhdan, M. y Hoppe, H. (2013). Screened poisson surface reconstruction. *ACM Transactions on Graphics (TOG)*, 32(3):29.
- Knopf, G. K. y Al-Naji, R. (2001). Adaptive reconstruction of bone geometry from serial cross-sections. *Artificial Intelligence in Engineering*, 15(3):227–239.

- Kobbelt, L., Campagna, S., y Seidel, H.-P. (1998). A general framework for mesh decimation. En *Graphics interface*, volumen 98, pp. 43–50.
- Kou, X. y Tan, S. (2007). Heterogeneous object modeling: A review. *Computer-Aided Design*, 39(4):284–301.
- Kumar, V. y Dutta, D. (1997). An approach to modeling multi-material objects. En *Proceedings of the fourth ACM symposium on Solid modeling and applications*, pp. 336–345. Nueva York: ACM.
- Lee, C.-H. (1986). Recursive region splitting at hierarchical scope views. *Computer Vision, Graphics, and Image Processing*, 33(2):237–258.
- Lin, C.-F., Yang, D.-L., y Chung, Y.-C. (2001). A marching voxels method for surface rendering of volume data. En *Computer Graphics International 2001. Proceedings*, pp. 306–313. Washington: IEEE.
- Liu, Y., Yang, H., y Wang, W. (2005). Reconstructing b-spline curves from point clouds—a tangential flow approach using least squares minimization. En *Shape Modeling and Applications, 2005 International Conference*, pp. 4–12. Washington: IEEE.
- Lorensen, W. E. y Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. En *ACM siggraph computer graphics*, volumen 21, pp. 163–169. ACM.
- Mantyla, M. (1998). An introduction to solid modeling. Nueva York: W. H. Freeman.
- Martin, K. y Hoffman, B. (2010). *Mastering CMake: a cross-platform build system*. Nueva York: Kitware.
- Moustakides, G., Briassoulis, D., Psarakis, E., y Dimas, E. (2000). 3d image acquisition and nurbs based geometry modelling of natural objects. *Advances in Engineering Software*, 31(12):955–969.
- Nicol, C. J. (1995). A systolic approach for real time connected component labeling. *Computer Vision and Image Understanding*, 61(1):17–31.
- Ohlander, R., Price, K., y Reddy, D. R. (1978). Picture segmentation using a recursive region splitting method. *Computer graphics and image processing*, 8(3):313–333.
- Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., y Seidel, H.-P. (2005). Multi-level partition of unity implicits. En *ACM SIGGRAPH 2005 Courses*, p. 173. ACM.

- Ohtake, Y. y Suzuki, H. (2013). Edge detection based multi-material interface extraction on industrial ct volumes. *Science China Information Sciences*, 56(9):1–9.
- Ozbolat, I. T. y Koc, B. (2011). Multi-directional blending for heterogeneous objects. *Computer-Aided Design*, 43(8):863–875.
- Pasko, A., Adzhiev, V., Sourin, A., y Savchenko, V. (1995). Function representation in geometric modeling: concepts, implementation and applications. *The Visual Computer*, 11(8):429–446.
- Pavarino, E., Neves, L. A., Machado, J. M., de Godoy, M. F., Shiyou, Y., Momente, J. C., Zafalon, G. F., Pinto, A. R., y Valêncio, C. R. (2013). Free tools and strategies for the generation of 3d finite element meshes: modeling of the cardiac structures. *Journal of Biomedical Imaging*, 2013:5.
- Prince, J. L. y Links, J. M. (2006). *Medical imaging signals and systems*. Nueva Jersey: Pearson.
- Reitinger, B., Bornik, A., y Beichel, R. (2005). Constructing smooth non-manifold meshes of multi-labeled volumetric datasets.
- Ricci, A. (1973). A constructive geometry for computer graphics. *The Computer Journal*, 16(2):157–160.
- Rogers, D. F. (2000). *An introduction to NURBS: with historical perspective*. San Francisco: Morgan Kaufmann.
- Rosenfeld, A. (1976). *Digital picture processing*. Nueva York: Academic press.
- Roussellet, V. (2018). *Implicit muscle models for interactive character skinning*. Tesis doctoral, Tolouse: Université Paul Sabatier-Toulouse III.
- Samet, H. (1980). Region representation: Quadtrees from boundary codes. *Communications of the ACM*, 23(3):163–170.
- Saye, R. (2015). An algorithm to mesh interconnected surfaces via the voronoi interface. *Engineering with Computers*, 31(1):123–139.
- Schroeder, C., Regli, W. C., Shokoufandeh, A., y Sun, W. (2005). Computer-aided design of porous artifacts. *Computer-Aided Design*, 37(3):339–353.
- Siu, Y. y Tan, S. (2002). Source-based heterogeneous solid modeling. *Computer-Aided Design*, 34(1):41–55.

- Skala, V. (2013). Fast interpolation and approximation of scattered multidimensional and dynamic data using radial basis functions.
- Smith, L. I. (2002). *A tutorial on principal components analysis*. Dunedin: Universidad de Otago.
- Solomon, D. C. (1976). The x-ray transform. *Journal of Mathematical Analysis and Applications*, 56(1):61–83.
- Stytz, M. R., Frieder, G., y Frieder, O. (1991). Three-dimensional medical imaging: algorithms and computer systems. *ACM Computing Surveys (CSUR)*, 23(4):421–499.
- Suk, M. y Chung, S.-M. (1983). A new image segmentation technique based on partition mode test. *Pattern recognition*, 16(5):469–480.
- Sun, W. y Hu, X. (2002). Reasoning boolean operation based modeling for heterogeneous objects. *Computer-Aided Design*, 34(6):481–488.
- Tanimoto, S. (2014). *Structured computer vision: machine perception through hierarchical computation structures*. Nueva York: Academic Press.
- Taubin, G. (1995). Estimating the tensor of curvature of a surface from a polyhedral approximation. En *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pp. 902–907. Cambridge: IEEE.
- Tsai, Y.-C., Huang, C.-Y., Lin, K.-Y., Lai, J.-Y., y Ueng, W.-D. (2009). Development of automatic surface reconstruction technique in reverse engineering. *The International Journal of Advanced Manufacturing Technology*, 42(1-2):152–167.
- van Rikxoort, E. M. (2009). *Segmentation of anatomical structures in chest CT scans*. Tesis doctoral, Postbus: Universidad de Utrecht.
- Vese, L. A. y Chan, T. F. (2002). A multiphase level set framework for image segmentation using the mumford and shah model. *International journal of computer vision*, 50(3):271–293.
- Wang, L. (2014). *Reconstruction and Deformation of Objects from Sampled Point Clouds*. Tesis doctoral, The Ohio State University.
- Wang, L., Yu, Y., Zhou, K., y Guo, B. (2011). Multiscale vector volumes. *ACM Transactions on Graphics (TOG)*, 30(6):167.

- Wang, M. Y. y Wang, X. (2005). A level-set based variational method for design and optimization of heterogeneous objects. *Computer-Aided Design*, 37(3):321–337.
- Wesseling, M., De Groote, F., Meyer, C., Corten, K., Simon, J.-P., Desloovere, K., y Jonkers, I. (2016). Subject-specific musculoskeletal modelling in patients before and after total hip arthroplasty. *Computer methods in biomechanics and biomedical engineering*, 19(15):1683–1691.
- Withey, D. J. y Koles, Z. J. (2007). Medical image segmentation: Methods and software. En *Noninvasive Functional Source Imaging of the Brain and Heart and the International Conference on Functional Biomedical Imaging, 2007. NFSI-ICFBI 2007. Joint Meeting of the 6th International Symposium on*, pp. 140–143. Hangzhou: IEEE.
- Wu, W., Lee, P. V., Bryant, A. L., Galea, M., y Ackland, D. C. (2016). Subject-specific musculoskeletal modeling in the evaluation of shoulder muscle and joint function. *Journal of biomechanics*, 49(15):3626–3634.
- Wu, X., Liu, W., y Wang, M. Y. (2008). A cad modeling system for heterogeneous object. *Advances in Engineering Software*, 39(5):444–453.
- Wu, Z. y Sullivan Jr, J. M. (2003). Multiple material marching cubes algorithm. *International Journal for Numerical Methods in Engineering*, 58(2):189–207.
- Xu, C. (2011). Mesh valmet: Validation metric for meshes. Recuperado de: <https://www.nitrc.org/projects/meshvalmet/>.
- Yoo, D.-J. (2013). Heterogeneous object modeling using the radial basis functions. *International Journal of Precision Engineering and Manufacturing*, 14(7):1133–1140.
- You, Y., Kou, X., y Tan, S. (2015). Adaptive meshing for finite element analysis of heterogeneous materials. *Computer-Aided Design*, 62:176–189.
- Yuan, Z., Yu, Y., y Wang, W. (2012). Object-space multiphase implicit functions. *ACM Transactions on Graphics (TOG)*, 31(4):114.
- Zhang, C. y Chen, T. (2001). Efficient feature extraction for 2d/3d objects in mesh representation. En *Image Processing, 2001. Proceedings. 2001 International Conference on*, volumen 3, pp. 935–938. Thessaloniki: IEEE.

Zhang, L., Kong, H., Liu, S., Wang, T., Chen, S., y Sonka, M. (2017). Graph-based segmentation of abnormal nuclei in cervical cytology. *Computerized Medical Imaging and Graphics*, 56:38–48.

Zhang, Y., Bajaj, C., y Xu, G. (2009). Surface smoothing and quality improvement of quadrilateral/hexahedral meshes with geometric flow. *Communications in Numerical Methods in Engineering*, 25(1):1–18.

Zhao, H.-K., Osher, S., y Fedkiw, R. (2001). Fast surface reconstruction using the level set method. En *Variational and Level Set Methods in Computer Vision, 2001. Proceedings. IEEE Workshop on*, pp. 194–201. Washington: IEEE.

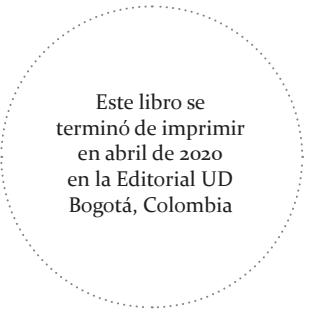
Zhou, K., Huang, J., Snyder, J., Liu, X., Bao, H., Guo, B., y Shum, H.-Y. (2005). Large mesh deformation using the volumetric graph laplacian. En *ACM Transactions on Graphics (TOG)*, volumen 24, pp. 496–503.



## **Autores**

### **Miller Gómez Mora**

Ingeniero en Computación de la Universidad Técnica de Georgia, especialista en Teleinformática y magíster en Ciencias de la información y las Comunicaciones de la Universidad Distrital Francisco José de Caldas, y doctor en Ingeniería de la Pontificia Universidad Javeriana de Bogotá. Es profesor Titular del programa de Ingeniería Telemática de la Universidad Distrital Francisco José de Caldas e investigador Junior del Sistema Nacional de Investigadores de Colombia (Colciencias). Los resultados de sus investigaciones se han publicado en diferentes revistas y presentado en congresos nacionales e internacionales, y ha formado parte de comités organizadores y científicos de varios de ellos. Sus líneas de investigación son computación gráfica, realidad virtual y sistemas telemáticos, que desarrolla principalmente en el grupo de investigación ImagiNet.



Este libro se  
terminó de imprimir  
en abril de 2020  
en la Editorial UD  
Bogotá, Colombia