



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

GENERACIÓN DE MODELOS DE
MALLAS DE TETRAEDROS PARA
SIMULADORES QUIRÚRGICOS

TESIS PROFESIONAL
QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

P R E S E N T A :
DAVID ARTURO SORIANO VALDEZ



DIRECTOR DE TESIS:
M. C. MIGUEL ÁNGEL PADILLA CASTAÑEDA

CIUDAD UNIVERSITARIA

2008



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

A mis padres.

Rosa María Valdez Ata y Arturo Soriano Baltazar.

Gracias por sus enseñanzas, sus consejos y su cariño. Sin duda ustedes son los que merecen el reconocimiento por haber enfrentado y superado hasta el día de hoy el enorme desafío que implica ser padre, pues esa es la tarea más difícil que uno puede realizar en esta vida. Gracias por nunca rendirse, y sobre todo gracias por no haberme permitido rendirme. Gracias por mostrarme el camino para alcanzar mis sueños, gracias por darme las herramientas para poder realizarlos y sobre todo por su apoyo incondicional.

A mi director de tesis.

Miguel Ángel Padilla Castañeda.

Le agradezco enormemente que haya depositado su confianza en mí, así como también le agradezco la enorme cantidad de tiempo que dedico a orientarme, corregirme y apoyarme para así poder concluir este trabajo, pero sobre todo le agradezco que me haya brindado su consejos para mi vida académica y profesional.

A mi familia.

Agradezco a mis abuelos el brindarme su cariño, así como haberme alentado siempre a ser mejor. También les doy gracias por compartir conmigo sus experiencias, ya que siempre que las escuchaba me sorprendía de saber todo lo que aun me falta por vivir.

Le doy las gracias a mi Tía Catalina por su apoyo a pesar de la distancia, gracias y le prometo seguir esforzándome.

Les doy las gracias a mis tíos Cirilo Y Marisol por su apoyo y tolerancia.

Gracias a todos, por su compañía, apoyo y consejo.

A mis amigos.

A mi hermano Gabriel, gracias por ser mi amigo, gracias por tu compañía, tus críticas y sobre todo tu sinceridad. Sin duda eres la persona que mejor me conoce y por lo mismo eres el mejor para hacerme ver mis errores. Espero sigas ayudándome a ser mejor y también espero poder ayudarte tanto como tú lo has hecho.

A mis amigos, gracias por su compañía, gracias por su sinceridad, gracias por su apoyo y sobre todo gracias por seguir siendo mis amigos.

Al Centro de Ciencias Aplicadas y Desarrollo Tecnológico

Agradezco al CCADET el haberme brindado la oportunidad y el apoyo para realizar esta tesis. También agradezco al Laboratorio de Análisis de Imágenes y Visualización por acogerme y brindarme su apoyo.

Índice

	Página
Resumen	1
1. Introducción	2
2. Interpolación de imágenes	6
2.1 Splines	7
2.1.1 Proceso de interpolación de imágenes por medio de splines	8
2.2 Morphing interpolation	13
2.2.1 Media morfológica	13
2.2.1.1 Dilatación	14
2.2.1.2 Erosión	15
3. Muestreo de imágenes	19
3.1 Muestreo regular con rejilla	19
3.2 Muestreo adaptivo	24
3.2.1 Mapa de distancia	24
4. Reconstrucción 3D	31
4.1 Tetraedrización	31
4.1.1 Tetraedrización de Delaunay	32
4.2 Filtrado de artefactos	35
4.3 Relajación	37
4.3.1 Modelo de masas y resortes	37
4.3.1.1 Método de Euler	48
5. Resultados	42
5.1 Modelos	42
5.1.1 Muestreo regular con <i>splines</i> sin relajación	43
5.1.2 Muestreo regular con <i>morphing</i> sin relajación	43
5.1.3 Muestreo adaptivo con <i>splines</i> sin relajación	44

5.1.4 Muestreo adaptivo con <i>morphing</i> sin relajación	44
5.1.5 Muestreo regular con <i>splines</i> con relajación	45
5.1.6 Muestreo regular con <i>morphing</i> con relajación	45
5.2 Análisis de resultados	46
Conclusiones	50
Bibliografía	52

Índice de figuras

	Página
Figura 1.1 Bordes de los cortes que componen el objeto.	4
Figura 2.1 Fenómeno de Runge.	7
Figura 2.2 Interpolación <i>splines</i> .	9
Figura 2.3 Ampliación de imagen interpolada por medio de splines.	9
Figura 2.4 Elemento estructurante (<i>disk</i>), con un radio de 2 píxeles.	10
Figura 2.5 Ejemplo de falla de interpolación debida a forma significativamente diferente.	11
Figura 2.6 Proceso de interpolación por splines.	12
Figura 2.7 (A) Dominio no convexo, (B) Distancia obtenida con píxeles D8.	14
Figura 2.8 Dilatación.	15
Figura 2.9 Erosión.	15
Figura 2.10 Media morfológica.	16
Figura 2.11 Esquema media morfológica.	18
Figura 3.1 Rejillas de muestreo.	19
Figura 3.2 (A) Región a muestrear.	20
Figura 3.3 Proceso de ajuste del contorno.	22
Figura 3.4 Proceso de ajuste de puntos.	23
Figura 3.5 Mapa de distancia con función de probabilidad moduladora del muestreo.	25
Figura 3.6 Mapa de distancia normalizado con tres regiones identificadas.	26
Figura 3.7 Validación de puntos.	27
Figura 3.8 Muestreo adaptivo de bordes.	27
Figura 3.9 Funciones de calculo de radio.	29
Figura 3.10 Muestreo adaptivo.	30
Figura 4.1 Estructura de la rejilla 3D.	32

Figura 4.2 Triangulación de Delaunay.	33
Figura 4.3 Carcasa convexa.	34
Figura 4.4 Carcasa convexa del volumen.	34
Figura 4.5 Validación de elementos por medio de sus etiquetas.	35
Figura 4.6 Objeto generado después de realizar el filtrado por medio de las etiquetas.	36
Figura 4.7 Conexión de nodos para relajado.	38
Figura 4.8 Comparación entre modelos.	40
Figura 4.9 Relajado (corte).	41
Figura 5.1 Número de nodos.	46
Figura 5.2 Volumen generado.	47
Figura 5.3 Distribución del volumen de los tetraedros.	48
Figura 5.4 Composición de las mallas.	49

Resumen

En este trabajo de tesis se presenta un método automático para generar mallas de tetraedros tanto regulares como adaptivas, las cuales serán empleadas para construir modelos 3D de tetraedros de estructuras anatómicas para aplicaciones de Cirugía Asistida por Computadora, como por ejemplo en sistemas de simulación quirúrgica que requieran modelos de elemento finito (FEM) de mallas con la menor complejidad posible pero visualmente aceptables.

Este método permite generar mallas a partir de una pila de imágenes segmentadas o anotadas, que en conjunto aproximan la forma del objeto en 3 dimensiones y permite al usuario controlar la resolución de salida tanto fija como adaptiva.

El método aquí presentado consta de tres procesos fundamentales, las cuales se ejecutan de modo secuencial, estos procesos son: interpolación de las imágenes, muestreo del volumen aproximado por las imágenes y reconstrucción 3D a partir de los puntos muestreados.

Como resultado de esta tesis se presenta la construcción de varios modelos 3D de la próstata para un simulador de Resección Transuretral de la Próstata. Se comparan los modelos obtenidos a diferente resolución tanto adaptivos como regulares, construidos con base en las técnicas descritas a lo largo de esta tesis y se observa que gracias al método adaptivo propuesto es posible construir mallas de menor complejidad, pero con suficiente calidad visual. Finalmente se presenta un análisis de los resultados obtenidos y en las conclusiones se presentan algunas perspectivas de mejoras del método propuesto.

Capítulo 1. Introducción

Al generar modelos en 3D para simulaciones, se discretiza el volumen por medio de figuras geométricas, dependiendo del método empleado para resolver el problema de simulación se pueden emplear triángulos o tetraedros. Dos de los métodos más populares para la resolución de estos problemas son el método de elementos finitos (FEM) y el método de elementos de frontera (BEM); cada uno de ellos posee sus ventajas y desventajas, aun así dependiendo del tipo de problema que se desee resolver un método es más apropiado que el otro.

BEM es un método en el que la representación del modelo solo se realiza sobre la frontera del volumen, de modo que no se emplean elementos de volúmenes sino de superficie, para el caso tridimensional son triángulos. Un problema que presenta este método es la dificultad para su implementación, esto se debe a la complejidad de las ecuaciones que lo modelan. [Drother]

FEM es un método que consta de dividir el dominio del problema en pequeños elementos, ya sean triángulos en el caso de dominios bidimensionales o tetraedros para el caso de dominios tridimensionales, a estos dominios les denominamos elementos y a los vértices que conforman estos elementos les llamamos nodos. La implementación de este método no es tan compleja como el caso del BEM, además permite representar geometrías complejas [Drother].

Este método es muy efectivo cuando deseamos representar geometrías sumamente irregulares, no homogéneas, con propiedades anisotrópicas de los materiales y condiciones de frontera. Por este motivo FEM es un método muy empleado en análisis de estructuras [AmeshGen], sin embargo se requiere de mucho esfuerzo para generar modelos que sean útiles. Al referirnos a útil hablamos que el modelo debe tener una buena calidad visual (ser lo parecido al objeto real) y que no sea muy denso (el número de elementos que lo conforman no sea muy grande). Por esta razón tener un método robusto para la generación automática de mallas de tetraedros es de suma importancia.

Existen muchos campos que requieren de la creación de mallas, cada una con sus necesidades propias, por ejemplo, en el caso de transferencia de calor, flujo de fluidos y cuerpos sólidos, la malla se deforma muy poco o no se deforma; en el caso de simulaciones de tejidos suaves para sistemas de simulación quirúrgica, la malla experimenta muchas deformaciones [TetMesh].

Los simuladores quirúrgicos son herramientas que se pueden emplear en el entrenamiento de médicos residentes, reduciendo los costos y los riesgos que representan una cirugía real, además de que se puede incrementar la práctica reduciendo así el tiempo de entrenamiento.

Cuando se generan mallas para simuladores quirúrgicos se desea que la malla sea capaz de simular las deformaciones en tiempo real, para esto puede ser muy útil contar con métodos de generación de mallas adaptivas, las cuales poseen elementos más pequeños en las regiones donde se desea obtener mayor resolución, con lo que se logran calcular deformaciones en menor tiempo que con modelos no adaptivos. Sin embargo, lograr esto es complicado, por lo que se opta por generar mallas con resoluciones fijas a lo largo de todo el modelo, estos modelos tienen el inconveniente de que para que funcionen de la manera deseada la resolución debe ser muy alta, lo que repercute en la velocidad de la simulación.

Uno de los desafíos más grandes en estos simuladores es lograr que los modelos empleados simulen el comportamiento del tejido o los órganos, pues esta simulación se debe poder realizar en tiempo real, de lo contrario es inútil. Para lograr el efecto de una simulación en tiempo real se deben desplegar animaciones a una velocidad de 10 a 30 cuadros por segundo, una manera de conseguir esto es sacrificando precisión para obtener velocidad. Esto se logra al reducir la calidad del modelo, sin embargo las técnicas actuales para la generación de modelos tienen énfasis en la representación precisa del objeto, lo cual las hace inapropiadas para algunas aplicaciones en simulaciones médicas [Morten].

Por esta razón desarrollar técnicas para generar modelos de mallas de tetraedros que permitan simulaciones médicas en tiempo real, sin sacrificar de manera significativa el aspecto visual, es de suma importancia. De este modo se pueden obtener simulaciones de buena calidad sin tener que recurrir a otras técnicas para incrementar la velocidad del tiempo de cómputo del simulador quirúrgico.

La mayoría de los trabajos reportados en la literatura se basan en hacer reconstrucciones de mallas no adaptivas, como el trabajo reportado en [TetMesh], donde se hace hincapié en que los métodos basados en Delaunay no son tan buenos cuando se aplican a problemas tridimensionales en comparación a cuando se aplican a problemas bidimensionales. En lugar de ello, inician con la discretización de los bordes y a partir de ellos generan elementos hacia dentro del volumen. Sin embargo, aunque estos métodos parecen más avanzados, no es tan

claro como se pueden obtener mallas adaptivas con resolución variable, además de que presentan problemas con bordes con una gran curvatura.

El objetivo de este trabajo de tesis es presentar un método adaptivo para generar mallas de tetraedros con resolución variable de manera automática, estas mallas se emplearan en simuladores quirúrgicos, de modo que las mallas generadas deben poder modelar una gran cantidad de deformaciones.

El método parte del hecho de que se cuenta con una serie de imágenes, las cuales son cortes del objeto a representar, además en ellas se han anotado las regiones de interés, definiendo así el objeto del cual se desea obtener el modelo. En la figura 1.1 se muestran los bordes anotados en las imágenes que se proporcionan, así como la disposición con la que se ordenan estos cortes para obtener el modelo.

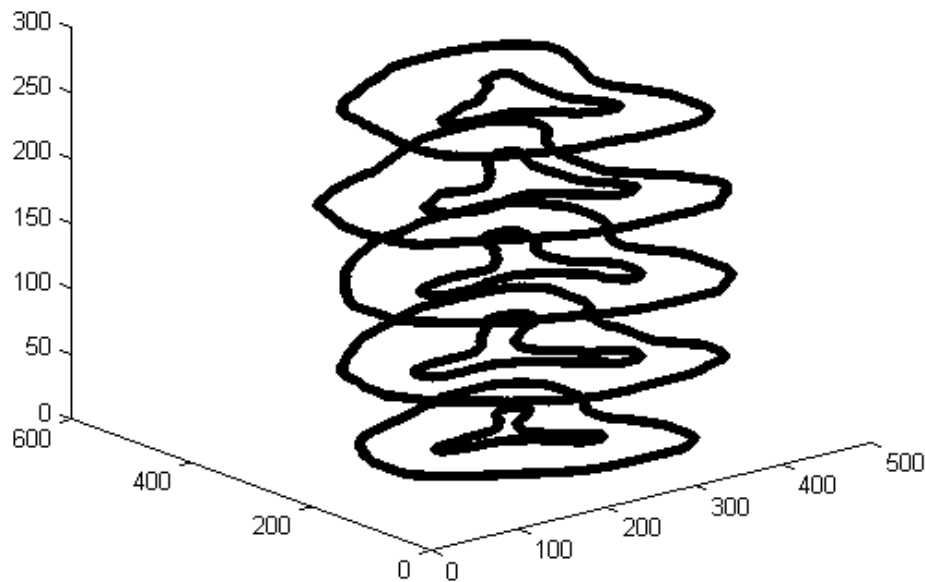


Figura 1.1 Bordes de los cortes que componen el objeto.

El método propuesto consiste de tres pasos esenciales. El primero de ellos es la interpolación de imágenes, esto es necesario puesto que la información que uno puede obtener por medio de imágenes en ocasiones es muy escasa, por lo que si uno tomará tal cual esa información el modelo generado tendría una mala calidad visual. Para realizar la interpolación se presentan dos técnicas, una de ellas emplea

splines para realizar la interpolación de contornos y la otra emplea la media morfológica para la obtención de regiones.

El siguiente paso es el muestreo del volumen; en esta parte se discretiza el objeto, con lo que se obtiene una nube de puntos, la cual posteriormente se empleará para generar el modelo de mallas de tetraedros. Las técnicas empleadas para realizar el muestreo fueron: el muestreo regular basado en el uso de una rejilla de muestreo y el muestreo adaptivo que se basa en la obtención de mapas de distancia para controlar la densidad del muestreo.

Por último se realiza la reconstrucción del modelo por medio de la tetraedrización de Delaunay. En esta parte del proceso se deben filtrar los artefactos generados al aplicar los métodos de Delaunay. Una vez filtrados estos artefactos se realiza una relajación del modelo por medio de un sistema de masas y resortes para redistribuir de manera uniforme la posición de los vértices de la malla, a fin de reducir posibles anisotropías en aplicaciones posteriores debidas a la geometría de la malla. Una vez realizado esto se cuenta con un modelo de mallas de tetraedros que puede ser incluido en un simulador quirúrgico.

Este método permite generar mallas adaptivas y regulares, las cuales poseen una buena calidad visual al mismo tiempo que no son demasiado densas. También cumple con el objetivo de proponer un método que permita generar las mallas de manera automática. Sin embargo, el método tiene sus limitaciones en cuanto a la variedad de modelos que se pueden generar, por lo que al final se presentan en las conclusiones algunas ideas y perspectivas futuras para mejorar las técnicas de interpolación, las técnicas de muestreo y por último el filtrado de artefactos.

Capítulo 2. Interpolación de imágenes

La interpolación de imágenes es un termino empleado para nombrar a un conjunto de técnicas que nos permiten crear imágenes intermedias a partir de las imágenes que poseemos. En esta tesis se trabajo con la generación de un modelo de mallas de tetraedros de una próstata, para ello se emplearon imágenes anotadas manualmente, en vez de emplear imágenes de ultrasonido; esto se debe a que el proceso para anotar imágenes de ultrasonido de manera automática es un proceso muy complejo y por lo tanto se partió de la premisa de que se contaba con las imágenes anotadas.

Cuando generamos modelos de mallas en 3D a partir de imágenes segmentadas o anotadas nos vemos limitados al número de imágenes con que contamos. En casos en los que el objeto a generar no tiene mucho detalle o este no interesa, el problema no es tan serio, pero cuando el objeto que deseamos modelar es pequeño o con muchos detalles, lo más posible es que contemos con pocas imágenes a partir de las cuales podamos generar el modelo, como ocurre en ocasiones en imágenes médicas. Para evitar que a causa de falta de información, en este caso imágenes segmentadas o anotadas, se afecte de manera significativa la calidad del modelo generado se recurren a técnicas de interpolación con el fin de obtener imágenes adicionales, las cuales ayuden a mejorar la resolución del modelo.

Dentro de las técnicas que podemos emplear se encuentran la: interpolación lineal y la no lineal. La interpolación lineal que empleamos en este trabajo de tesis fue la media morfológica, esta forma parte de las técnicas de interpolación morfológica, pero ya que sólo empleamos la media morfológica es que es catalogada como interpolación lineal. La otra técnica que empleamos es una que emplea *splines* de manera transversal a los contornos que poseemos, esta técnica fue implementada de manera muy simple y requiere de más análisis para mejorar su desempeño.

2.1 Splines

En el campo del análisis numérico una *spline* es una función especial definida por varios polinomios, cada uno de ellos definido en un subintervalo, los cuales se unen entre sí siguiendo ciertas condiciones de continuidad [WSpline]. Cuando se resuelven problemas de interpolación la técnica más utilizada es la que emplea *splines* debido a que los resultados que entrega son muy similares a los que entregaría la interpolación por medio de polinomios, con la ventaja de que con el uso de *splines* se evitan las oscilaciones que generan las interpolaciones con polinomios de grado elevado, propiciando que la aproximación a la curva empeore, este fenómeno es conocido como el fenómeno de Runge [Runge]. (Figura 2.1)

Puesto que las *splines* no generan oscilaciones es que son tan socorridas en la aproximación de formas complicadas, además de que son fácilmente resolubles computacionalmente hablando; esto las hace ideales para aplicaciones de computación gráfica, donde se deseen generar gráficos suavizados de manera eficiente.

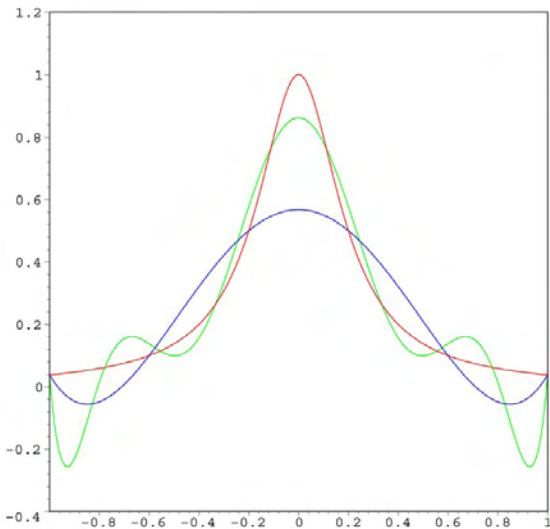


Figura 2.1 Fenómeno de Runge.
La curva en rojo es la función de Runge.
La curva en azul es la función interpolada con un polinomio de 6º orden.
La curva en verde es la función interpolada con un polinomio de 9º orden.
Se observan oscilaciones las cuales empeoran la aproximación [WRunge].

La forma más común de representar una *spline* S es

$$S : [a, b] \rightarrow \mathbf{R} \quad \text{.....(2.1)}$$

la cual consiste de partes de polinomios definidas por

$$P_i : [t_i, t_{i+1}) \rightarrow \mathbf{R} \quad \text{.....(2.2)}$$

Donde

$$a = t_0 < t_1 < \dots < t_{k-2} < t_{k-1} = b$$

tal que

$$S(t) = \begin{cases} P_0(t), t_0 \leq t < t_1 \\ P_1(t), t_1 \leq t < t_2 \\ \vdots \\ P_{k-2}(t), t_{k-2} \leq t \leq t_{k-1} \end{cases} \dots(2.3)$$

Los k puntos de t se llaman *nodos*. Además de esto se define un orden $n \geq 0$. Entonces decimos que tenemos una función *spline* de grado n con nudos en t_0, t_1, \dots, t_{k-1} , siempre y cuando la función S satisfaga las condiciones:

- i. En cada uno de los intervalos $[t_{i-1}, t_i]$, S es un polinomio de grado igual o menor a n .
- ii. S tiene una derivada de orden $(n-1)$ continua en $[t_0, t_{k-1}]$.

2.1.1 Proceso de interpolación de imágenes por medio de splines

A partir del conjunto C de los contornos que limitan las regiones de las cuales se desea generar el modelo se interpolan los demás contornos a fin de obtener una mayor cantidad de información. El primer paso es obtener el contorno de mayor perímetro, de este modo aseguramos que se interpolará una mayor cantidad de puntos. A este contorno lo llamaremos contorno de referencia (C_r).

Sea Q_i^r el conjunto de puntos pivote que forman C_r , se busca Q^k y Q^r para $k \neq r$ tal que $Q^k = \{Q_i^r \mid \min\}$.

Debido a la morfología de la próstata este proceso genera muchos huecos en los contornos que son interpolados dadas las diferencias de forma entre los contornos, lo que ocasiona problemas de correspondencia; lo cual se puede ver en la figura 2.2 y 2.3.

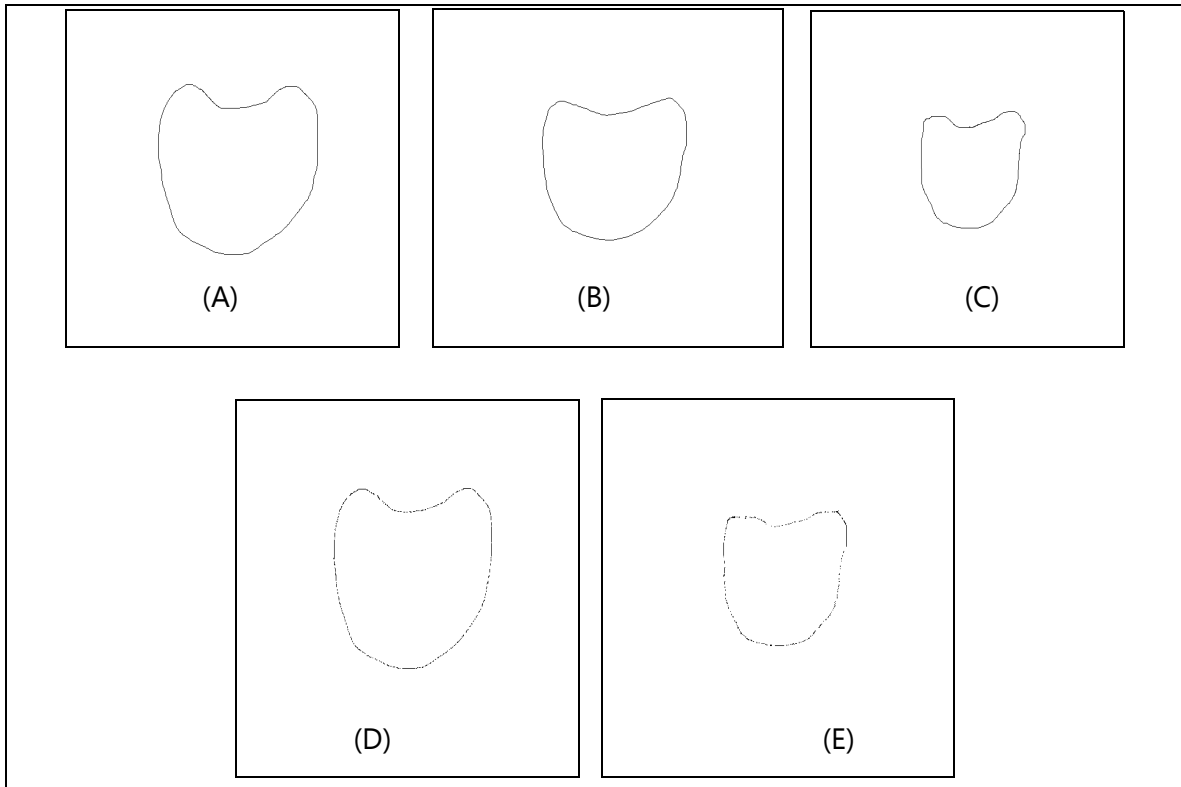


Figura 2.2 Interpolación *splines*
 (A) (B) (C) Imágenes base para realizar la interpolación;
 (D) (E) Imágenes interpoladas, en ellas se observan huecos debidos al método empleado para interpolar los contornos por *splines*.

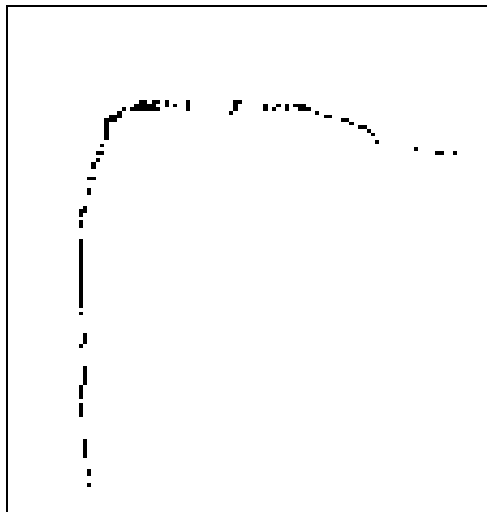


Figura 2.3 Ampliación de imagen interpolada por medio de *splines*.

Las imágenes resultantes de la interpolación por medio de *splines* son inservibles tal y como son generadas, pues necesitamos contornos continuos, esto se debe a que el criterio para asignar las correspondencias en los *splines* no es el óptimo; para poder mejorar el método se tienen que vectorizar y remuestrear los

contornos, además se deben agregar más criterios para determinar las correspondencias; aun así el método propuesto no es del todo inservible, pero necesita de procesos adicionales para ser útil. Entonces para poder aproximar el contorno que hemos interpolado primero debemos cerrar las discontinuidades, para ello recurrimos a la operación morfológica de cerradura [DImgPro].

La operación de cerradura de un conjunto A por un elemento estructurante B se define como

$$A \bullet B = (A \oplus B) \ominus B \quad \dots(2.4)$$

"...el cerrado de A por B es simplemente la dilatación de A por B , seguido de la erosión del resultado por B ".¹

Como se observa arriba la operación de cerradura requiere de dos cosas, un conjunto de elementos sobre los cuales se realizará el proceso, y un elemento estructurante el cual será el encargado de llevar a cabo el proceso. Para nuestro caso el conjunto de elementos lo conforman los puntos generados con la interpolación, mientras que el elemento estructurante lo definimos como una aproximación a un disco de radio fijo, como el que se muestra en la figura 2.4.

0	0	1	0	0
0	1	1	1	0
1	1	1	1	1
0	1	1	1	0
0	0	1	0	0

Figura 2.4 Elemento estructurante (*disk*), con un radio de 2 pixeles.

En la figura 2.4 se observa un elemento estructurante que tiende a asemejar un disco, el cual entre mayor sea su radio más será su semejanza. Claro que entre mayor sea el tamaño del elemento estructurante más tiempo tomara realizar la operación de cerradura, pero al mismo tiempo se mejora la exactitud de dicha operación.

Para cerrar el contorno necesitamos de un elemento estructurante de radio relativamente grande, lo cual produce artefactos dentro del contorno, que debemos eliminar. Antes de hacer esto extraemos el contorno generado gracias a la operación de cerrado, posteriormente llenamos la región de interés, esto es posible ya que el contorno se encuentra completamente cerrado. Una vez que se

1 Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Second Edition, pp. 528.

han eliminado los artefactos procedemos a extraer nuevamente los contornos de la imagen (figura 2.6).

La manera más simple para obtener los contornos es usando un detector de bordes, para ello empleamos los operadores de Sobel [DImgPro]. El tamaño del operador propuesto es de 3X3. La forma de los operadores empleados es la siguiente.

1 0 -1	-1 0 1	1 2 1	-1 -2 -1
2 0 -2	-2 0 2	0 0 0	0 0 0
1 0 -1	-1 0 1	-1 -2 -1	1 2 1
Sobel X1	Sobel X2	Sobel Y1	Sobel Y2

Usamos dos tipos de operadores en X y en Y debido a que tenemos una región a la que queremos extraer el borde. Cada uno de los operadores nos entrega una parte distinta del borde, la cual tenemos que unir al final para generar el borde completo.

Como resultado de todo el proceso anterior obtenemos la región de interés, así como el contorno perfectamente cerrado y sin ruido. Con este tipo de interpolación se obtuvieron buenos resultados con imágenes con contornos que son muy similares y conservan una forma bastante parecida las unas con las otras; pero si las imágenes no tuvieran esa similaridad el método empleado fallaría, puesto que el criterio para seleccionar los nodos para la spline se basa en buscar el punto mas cercano, sin tomar en cuenta las variaciones en la forma de los contornos del objeto (problema de correspondencia). (figura 2.5)

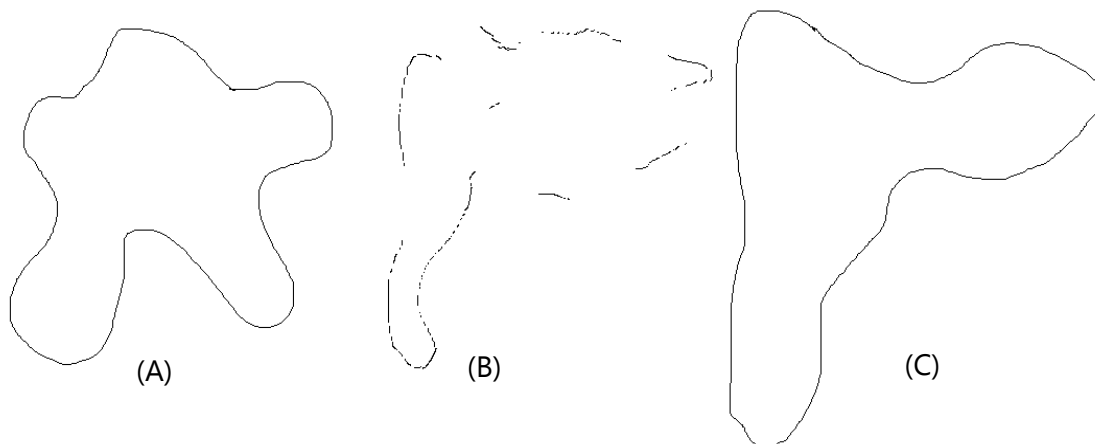


Figura 2.5 Ejemplo de falla de interpolación debida a forma significativamente diferente, (A) imagen 1, (B) imagen interpolada, (C) imagen 2.

El método funciona debido a que la forma de los contornos de la próstata son muy similares, pero no es lo suficientemente robusto para interpolar contornos con variaciones en sus formas muy significativas. En el siguiente diagrama (figura 2.6) observamos como obtenemos la región y el contorno de la imagen interpolada.

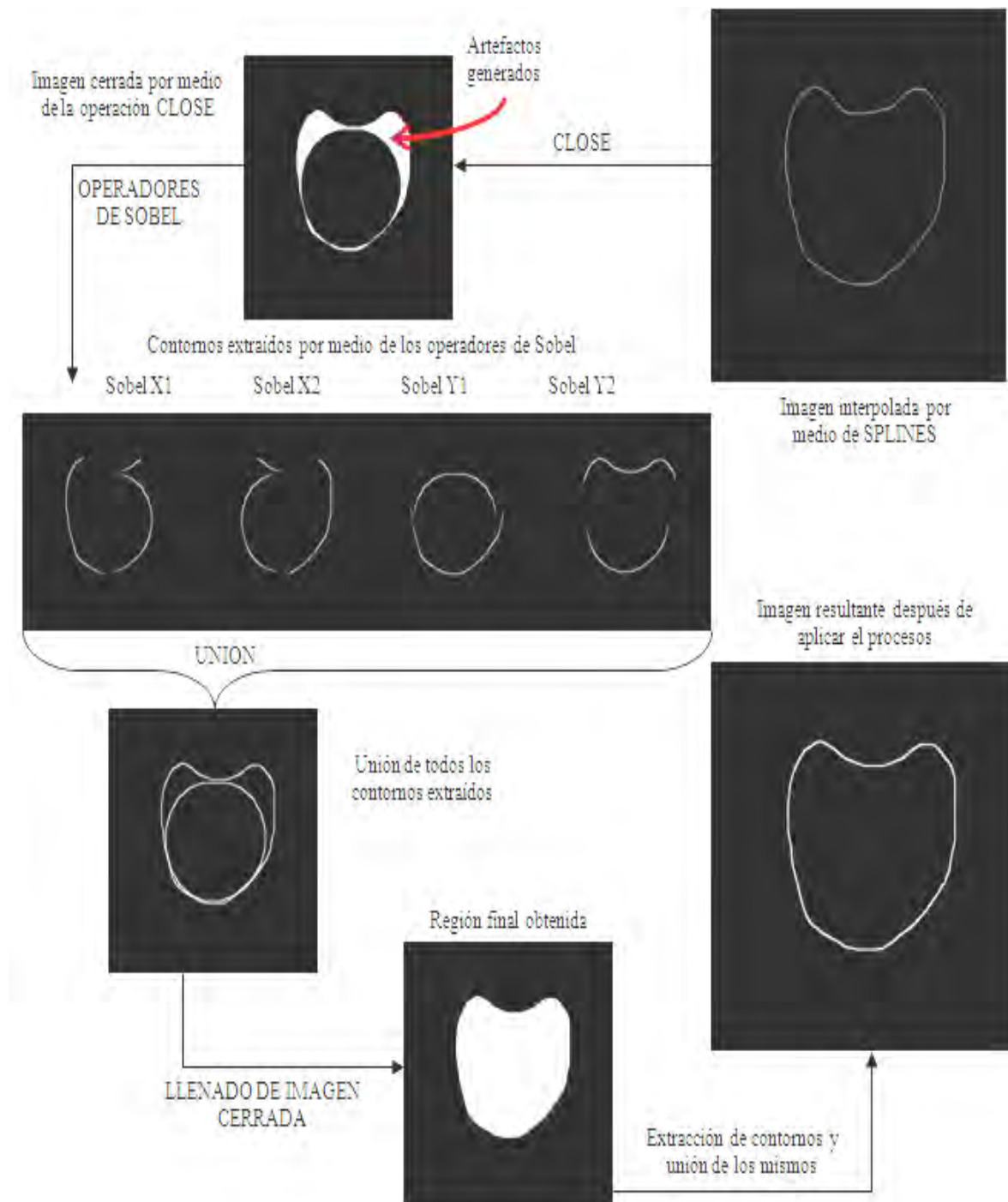


Figura 2.6 Proceso de interpolación por splines.

2.2 Interpolación morfológica

En la sección anterior empleamos la interpolación por medio de *splines*, con la técnica mostrada en esta tesis se observan buenos resultados siempre y cuando los objetos no tengan problemas de correspondencia, pero en el caso en que las formas sean significativamente diferentes no se obtienen los resultados deseados. Otras técnicas a las que se puede recurrir son las de interpolación morfológica. Estas técnicas están basadas en morfología matemática, para nuestro caso emplearemos una herramienta llamada Media Morfológica [MorInter].

2.2.1 Media morfológica

Para poder explicar lo que es la media morfológica antes debemos definir algunos conceptos extras. El primero de ellos es conocido como *zonas de influencia*.

Sean los conjuntos P_1, P_2, \dots, P_n los cuales son disjuntos, entonces la zona de influencia del conjunto P_i es el lugar en el que los puntos están más cerca al conjunto P_i que a cualquier otro conjunto. Si esto lo extendemos al caso en el que solo hay dos conjuntos entonces podemos decir lo siguiente:

Considerando los conjuntos P y Q , con Q incluida en P ($Q \subset P$), la zona de influencia ($IZ_P(Q)$) de Q con respecto a P' todavía es llamada la zona de influencia de P dentro de Q ,

$$IZ_P(Q) = \{x: d(x, Q) < d(x, P')\} \quad \dots(2.5)$$

Donde $d(x, A)$ denota a la *distancia geodesica*² entre el punto 'x' y el conjunto A. La distancia geodesica entre dos puntos p y q está definida como la longitud del camino más corto de p a q . Esta distancia es obtenida a partir de contar el número de vecinos de los píxeles que hay que recorrer hasta alcanzar el píxel deseado, esto se observa en la figura 2.7.

2 Geodesic Distance [MorInter]

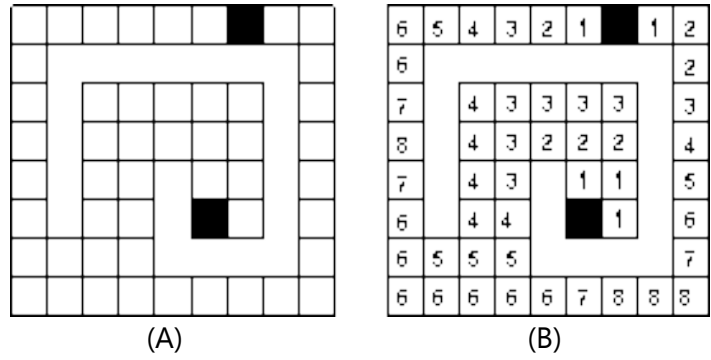


Figura 2.7 (A) Dominio no convexo, (B) Distancia obtenida con píxeles D8.

El contorno de la zona de influencia es un *SKIZ geodesico*³ de Q en P . La zona de influencia definida por la ecuación 2.5 representa el conjunto de media de los dos conjuntos, con la particularidad de que uno debe estar incluido en el otro.

$$M(P,Q) = IZ_P(Q) \quad \dots\dots(2.6)$$

La ecuación 2.6 tiene la siguiente representación, la cual puede corroborarse en el trabajo de Serra [Serra].

$$M(P,Q) = \cup \{(Q \oplus \lambda B) \cap (P \ominus \lambda B)\}, \quad \forall \lambda \quad \dots\dots(2.7)$$

Donde $\oplus \lambda B$ y $\ominus \lambda B$ son la dilatación y la erosión respectivamente de tamaño λ por el elemento estructurante B . Las operaciones de dilación y erosión son operaciones morfológicas, las cuales se definen de la siguiente manera.

Dados A y B definidos como conjuntos de Z^2 [DImgPro]:

2.2.1.1 Dilatación

La dilatación de A por B se denota como $A \oplus B$ y se define como

$$A \oplus B = \{z \mid [(B^\wedge)_z \cap A] \subseteq A\} \quad \dots\dots(2.8)$$

Esta ecuación se basa en la obtención de B^\wedge , que es la reflexión de B sobre su origen y de $(B^\wedge)_z$ que es la traslación de esta por z . La dilatación de A por el elemento estructurante B es el conjunto de todos los desplazamientos z , tales que B y A se superpongan por al menos en un elemento. En la figura 2.8 se observa el proceso de dilatación.

3 Geodesic SKIZ es el esqueleto de la zona de influencia. [MorInter]

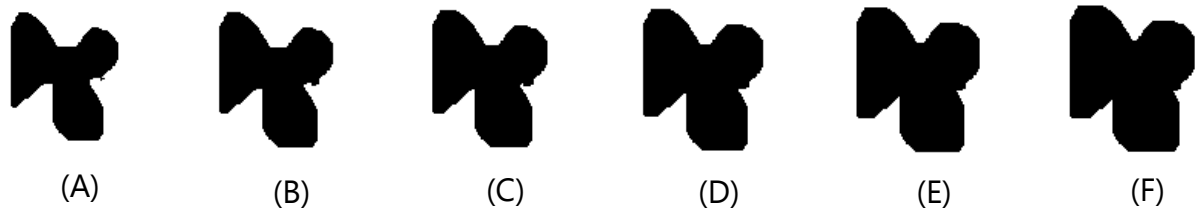


Figura 2.8 Dilatación. (A) Figura original, (B) (C) (D) (E) (F) proceso de dilatación aplicado consecutivamente.

2.2.1.2 Erosión

La erosión de A por B se denota como $A \ominus B$ y se define como

$$A \ominus B = \{z \mid (B)_z \subseteq A\} \quad \dots(2.9)$$

Esta ecuación indica que la erosión de A por B es el conjunto de todos los puntos z , tal que B trasladado por z está contenido en A . En la figura 2.9 se observa el proceso de erosión.

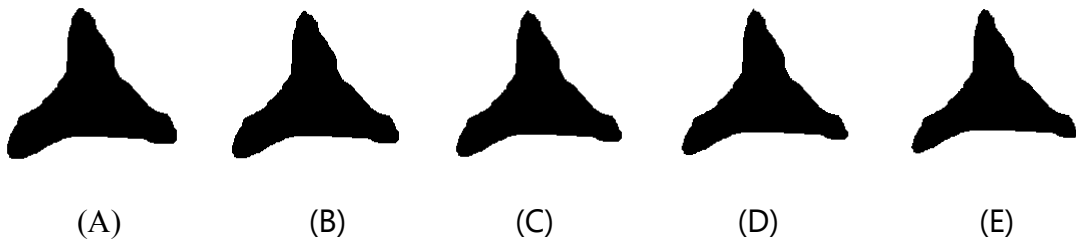


Figura 2.9 Erosión. (A) Figura original, (B) (C) (D) (E) proceso de erosión aplicado consecutivamente.

El método descrito en la ecuación (2.6) se aplica cuando la intersección de los conjuntos es el conjunto vacío, pero si existe una inclusión parcial de un conjunto con respecto al otro, entonces el conjunto de media de estos se define como la zona de influencia de la conjunción de estos en la disyunción de los mismos, lo que es:

$$M(P,Q) = I_{Z(P \cup Q)}(P \cap Q) \quad \dots(2.10)$$

La ecuación 2.10 describe un nuevo conjunto ubicado entre los dos conjuntos iniciales, el contorno de este nuevo conjunto representa el esqueleto de la conjunción en la disyunción de los conjuntos. Al final obtenemos la siguiente ecuación:

$$M(P,Q) = \cup \{[(Q \cap P) \oplus \lambda B] \cap [(Q \cup P) \ominus \lambda B]\}, \quad \forall \lambda \dots\dots(2.11)$$

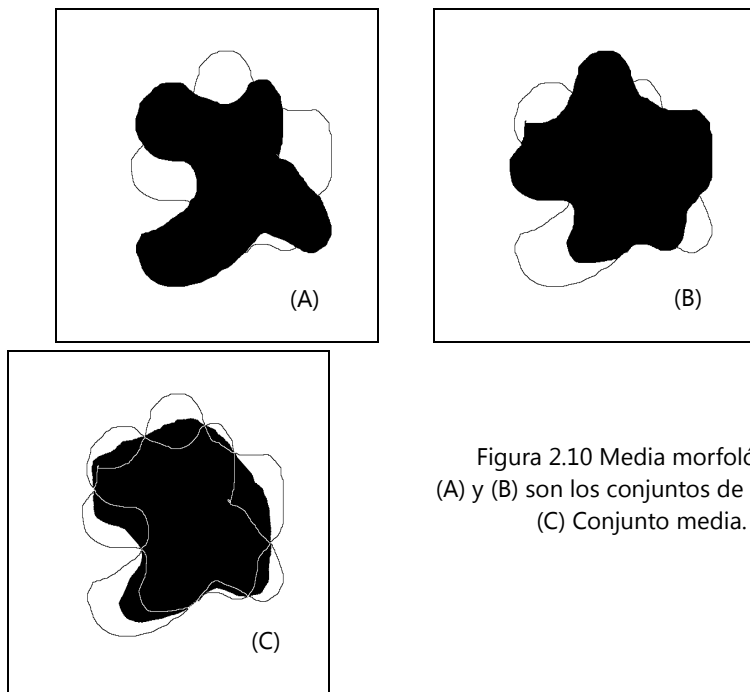


Figura 2.10 Media morfológica
(A) y (B) son los conjuntos de referencia
(C) Conjunto media.

La figura 2.10 muestra el resultado de aplicar la ecuación 2.11 sobre dos conjuntos. Puesto que es un método iterativo en el que la dilatación y erosión de la conjunción y disyunción respectivamente se repite hasta encontrar la zona de influencia común a ambos, se puede implementar mediante el siguiente algoritmo.

Algoritmo media morfológica

- (1) Obtener 2 imágenes para realizar la interpolación (Q, P)
- (2) Generar una nueva imagen con la intersección de Q y P (QyP)
- (3) Generar una nueva imagen con la unión de Q y P (QoP)
- (4) Generar una nueva imagen de la intersección de QyP y QoP (QP)
- Hacer {
- (5) Contar el numero de píxeles de valor 0 en QyP
- (6) Dilatar una vez QyP (EQyP)
- (7) Erosionar una vez QoP (DQoP)
- (8) Realizar la intersección de EQyP y DQoP (ED)
- (9) Realizar la unión de ED y QP, el resultado toma el lugar de QP
- (10) Contar la numero de píxeles de valor 0 en EQyP
- } repetir mientras la cuenta de píxeles al inicio sea distinta que la del final
- (11) El resultado es QP

El algoritmo de arriba produce como resultado la imagen intermedia a las dos imágenes de entrada. Si es que deseamos generar más de una imagen intermedia a nuestras imágenes solo debemos seguir repitiendo el algoritmo anterior con todas las imágenes que tengamos. Dado que al generar las imágenes interpoladas buscamos que la distancia entre ellas sea la misma, el algoritmo solo puede generar series con la siguiente correspondencia.

$$n + k = 2n - 1 \quad \forall n \in \mathbb{N} \quad \dots\dots(2.12)$$

Donde n es la cantidad de imágenes que se proporcionan al algoritmo y k es la cantidad de imágenes generadas por medio de la técnica de interpolación. Entonces, si le proporcionamos al inicio 5 imágenes, obtendremos como resultado 9 imágenes, 4 de las cuales fueron generadas por el algoritmo; si estas 9 las volvemos a proporcionar obtendremos 17 imágenes; si se quieren obtener aún más imágenes se deben seguir esa misma secuencia para asegurar que las imágenes tengan la correspondencia deseada en el modelo.

Por lo tanto, si partimos de dos imágenes cualesquiera podemos generar imágenes intermedias a estas dos con al siguiente correspondencia.

$$|T^k| = l_0, l_1=2(l_0)+1, l_2=2(l_1)+1, \dots, l_j = 2(l_{j-1})+1 \quad \dots\dots(2.13)$$

Donde T^k es el conjunto de imágenes interpoladas en j número de ejecuciones del algoritmo (interpolaciones) y l_0 es el numero de imágenes al iniciar el proceso, por lo tanto sí no seguimos la regla de correspondencia las imágenes no estarán separadas a la misma distancia una de las otras en el modelo. En la figura 2.11 se ilustra lo anterior.

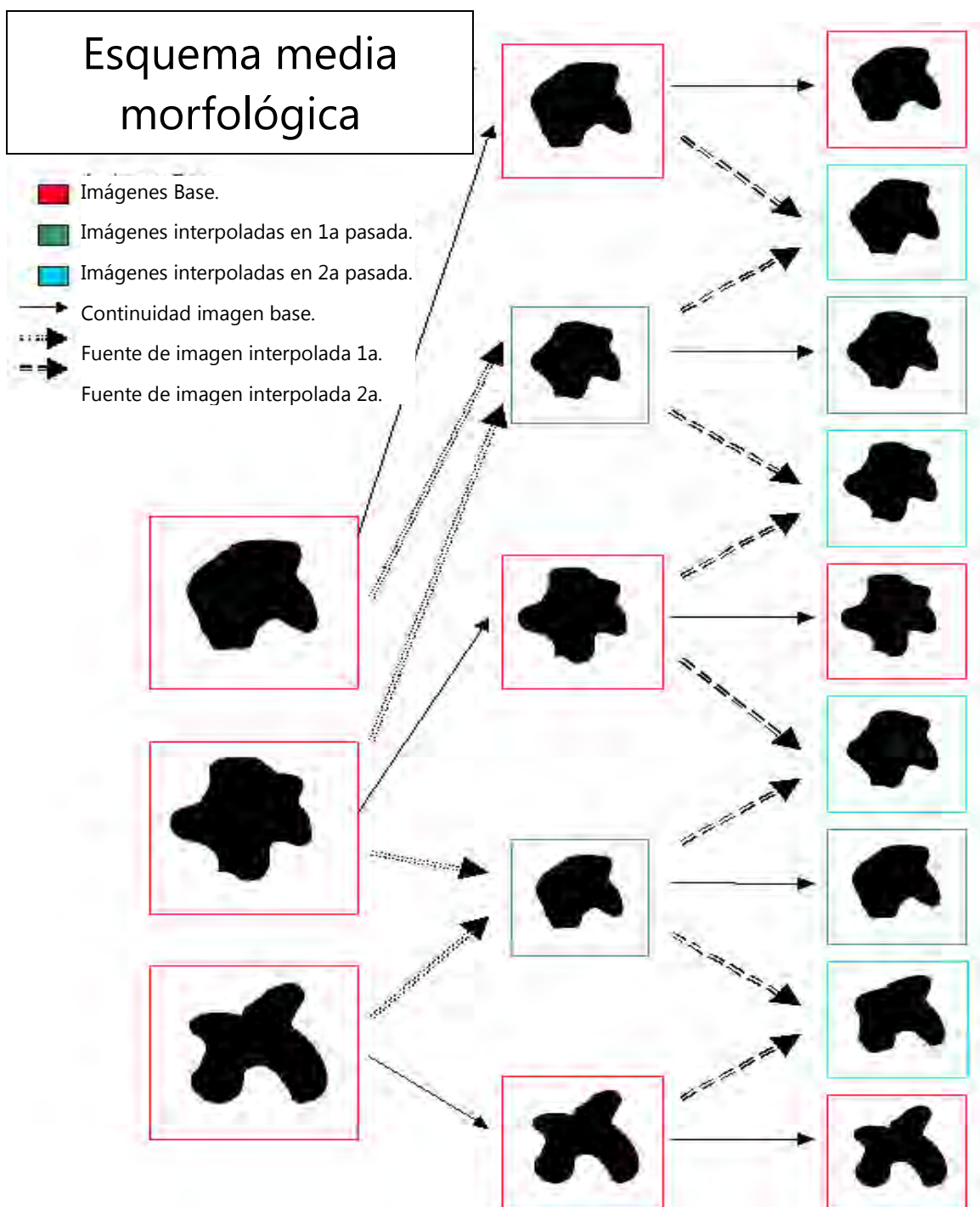


Figura 2.11 Esquema media morfológica.

Capítulo 3. Muestreo de imágenes

En el capítulo anterior se explicaron algunas de las técnicas empleadas para mejorar la forma externa del objeto mediante la interpolación de los contornos que lo aproximan. En este capítulo se explicará como muestrear la información interna del objeto a partir de los contornos generados, para así poder construir posteriormente el modelo de volumen de mallas de tetraedros en 3D.

Es importante que el muestreo del objeto permita tener el control sobre la resolución del modelo y el tamaño de los tetraedros generados; esto es de suma importancia ya que existe un compromiso entre la calidad del modelo (mayor resolución) y el tiempo de cálculo numérico sobre la malla (menor resolución). Por ejemplo en estudios de elemento finito (FEM en inglés), el modelo de mallas de tetraedros en 3D no debe contener demasiados tetraedros, si no sería imposible calcular deformaciones en tiempo real.

Los métodos empleados en esta tesis para realizar el muestreo son los siguientes: 1) muestreo regular con rejilla (grid); 2) muestreo adaptativo modulado por mapas de distancia.

3.1 Muestreo regular con rejilla

El muestreo regular con rejilla es un método muy simple y rápido, y consiste en hacer un muestreo con cierto espaciamiento horizontal y cierto espaciamiento vertical igual en ambas direcciones en 2D (Figura 3.1) y en profundidad para 3D.

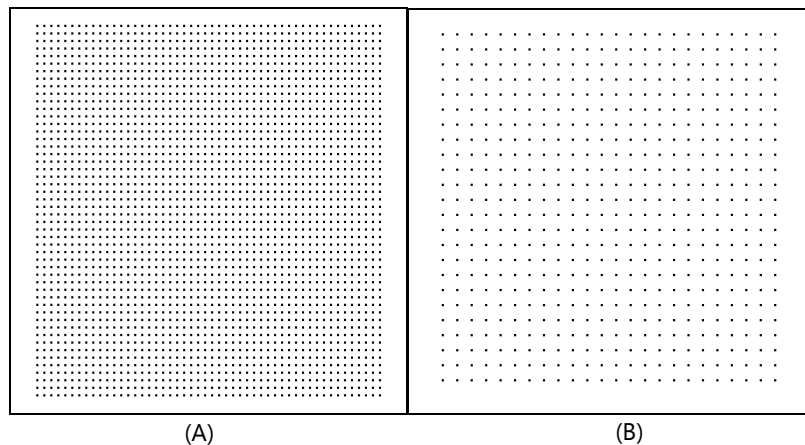


Figura 3.1 Rejillas de muestreo. (A) Rejilla para muestreo con espaciamiento horizontal de 10 píxeles y espaciamiento vertical de 10 píxeles (10X10); (B) Rejilla para muestreo con espaciamiento horizontal de 20 píxeles y espaciamiento vertical de 20 píxeles (20X20).

Como ya se mencionó este tipo de muestreo es muy rápido y sencillo, pero si tomamos las muestras tal cual, necesitaremos muestreos a muy baja resolución para aproximar los detalles más significativos de los objetos, lo cual incrementaría el número de puntos.

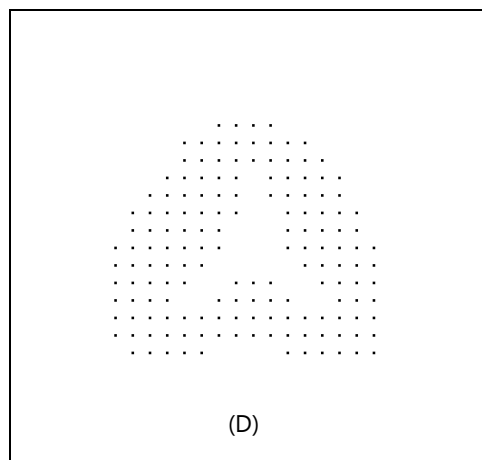
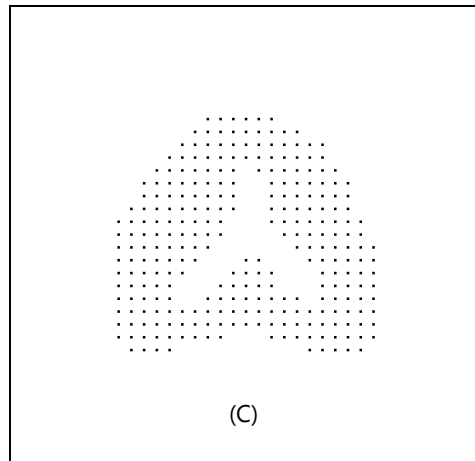
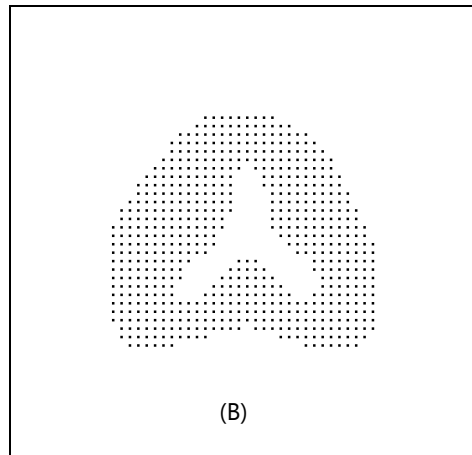
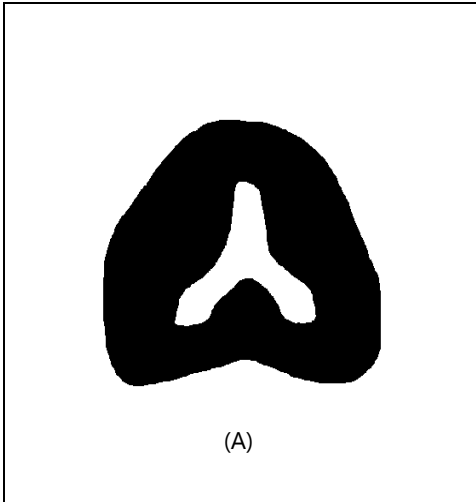


Figura 3.2 (A) Región a muestrear.
(B) Región muestreada con rejilla 10X10;
(C) Región muestreada con rejilla 15X15;
(D) región muestreada con rejilla 20X20.

El proceso "crudo" de muestreo con el cual generamos las figuras 3.2 B, C y D, es muy sencillo y rápido pues solo se trata de realizar la operación de producto lógico (AND) entre la imagen con la información del objeto correspondiente y la imagen de la rejilla, considerando el negro como 1 lógico y el blanco como 0 lógico.

Como se observa en la figura 3.2 entre menor sea el espaciamiento en la rejilla mejor es la representación de la región que obtenemos, pero aunque empleemos una rejilla con un espaciamiento muy pequeño seguimos sin muestrear de manera aceptable los bordes de dicha región. Para lograr un mejor muestreo de los bordes tendremos que agregar un paso de ajuste de posición de los nodos. El ajuste consiste en desplazar los puntos que fueron muestreados en el paso anterior y sean los más cercanos a los bordes hasta que se encuentren sobre estos últimos; de manera similar se desplazan los puntos que estén fuera de la región que forma el objeto, pero que al mismo tiempo estén lo suficientemente cerca de la región de interés. Para poder saber si un punto está lo suficientemente cerca es necesario definir una región para cada punto de la rejilla; estas regiones deben ser mutuamente excluyentes, pues de este modo el muestreo es más estructurado y controlable. Paralelamente también llevamos a cabo el proceso de etiquetado de puntos, de modo tal que identificamos los puntos pertenecientes a la región de interés y cuales forman parte de los bordes del objeto.

Formalmente el proceso anterior queda expresado de la siguiente manera:

Sea P el conjunto de los puntos de la rejilla y R el conjunto de puntos que conforman la región de interés, entonces

$$P_m = P \cap R \quad \text{..... (3.1)}$$

Y sea P_m' el complemento de P_m definido por

$$P_m' = P \cap R' \quad \text{..... (3.2)}$$

Donde R' es el complemento de la región R . Por otro lado,

$$P^K = (A(P_m, R_m) \cup A(P_m', R_m)) \quad \text{..... (3.3)}$$

Donde P^K es el conjunto de puntos ajustados y etiquetados, y A es la operación de ajuste a los contornos realizada sobre la región R_m de cada punto del conjunto. Los puntos se ajustan por medio del siguiente criterio:

$$A = (P_n(x,y) | R_n \cap R \neq 0, \min(P_n(x,y))) \quad \forall P_m \quad \dots\dots(3.4)$$

$$A = (P_n(x,y) | R_n \cap R' \neq 0, \min(P_n(x,y))) \quad \forall P_m' \quad \dots\dots(3.5)$$

Donde $P_n(x,y)$ es un punto de P_m o P_m' con una posición definida (x,y) , R_n es la región asignada al punto para realizar su ajuste y $\min(P_n(x,y))$ es la distancia mínima entre P_n y R .

Además de esto se puede mejorar el ajuste al definir un umbral en cuanto a la distancia, si la distancia es mayor a cierto parámetro definido por el usuario entonces se puede generar un punto adicional el cual se coloca sobre el borde en lugar de solo mover el punto al borde más próximo. En la figura 3.5 se ilustra el proceso de ajuste.

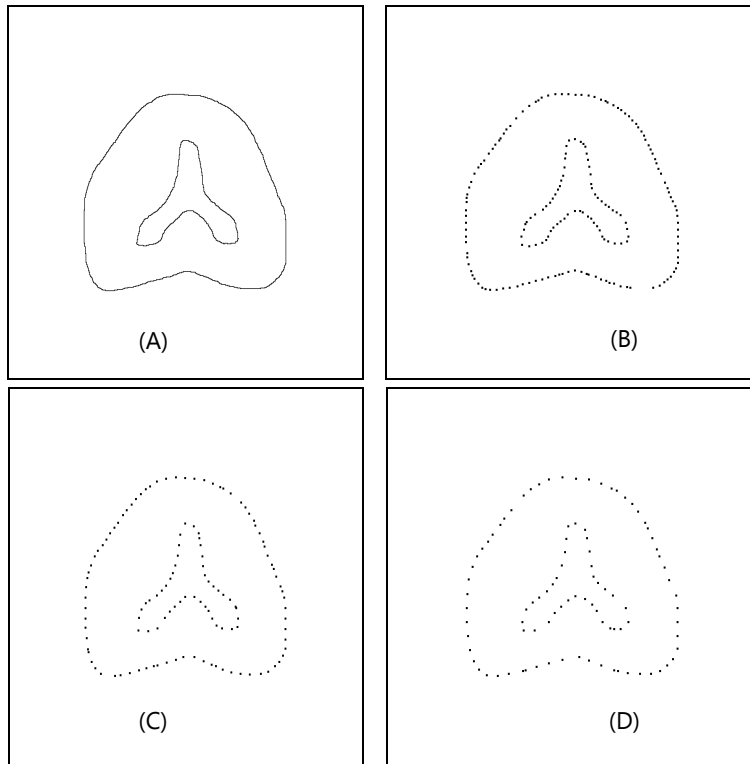


Figura 3.3 Proceso de ajuste del contorno. (A) Contorno Muestreado; (B) Contorno muestreado con rejilla 10X10; (C) Contorno muestreado con rejilla 15X15; (D) Contorno muestreado con rejilla 20X20.

En la figura 3.3 se presentan los resultados obtenidos al muestrear el contorno con distintos tamaños de rejillas, se puede observar que al realizar dicho muestreo se obtiene una buena representación del contorno.

El último paso es la unión de los puntos muestreados al inicio con la rejilla y los puntos que conforman el contorno; de este modo finaliza el proceso de muestreo del objeto. Este proceso se repite para cada una de las imágenes generadas en el proceso de interpolación.

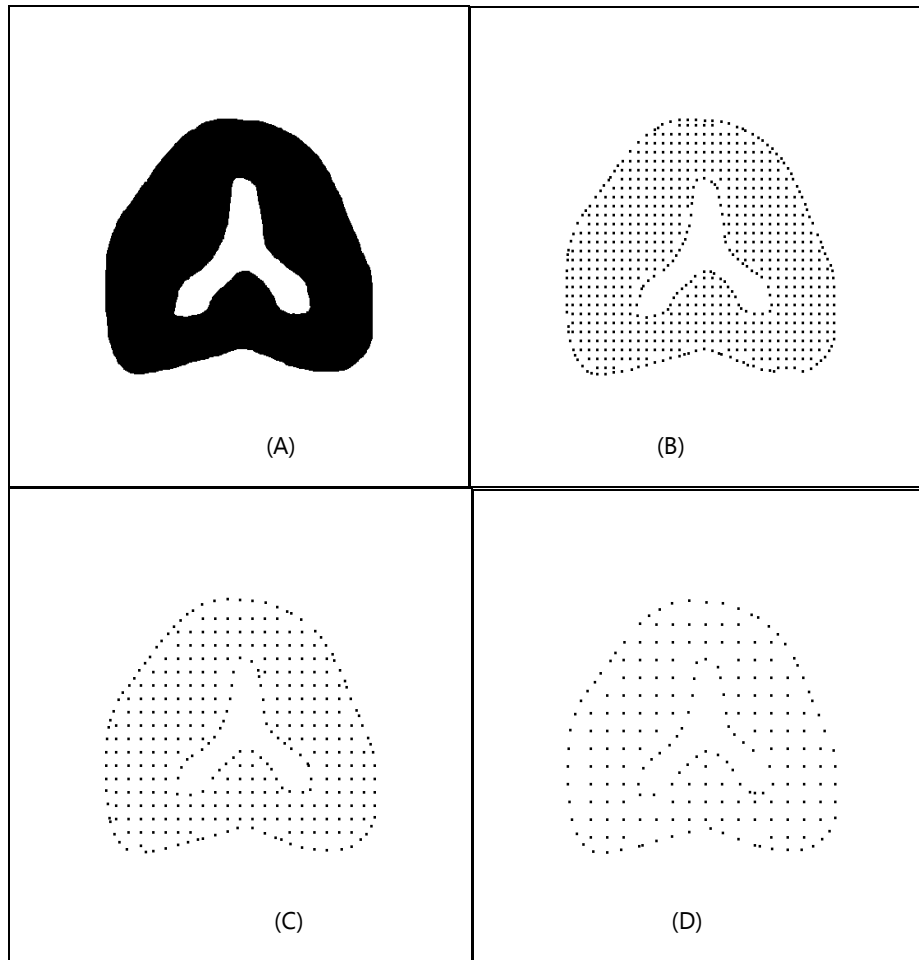


Figura 3.4 Proceso de ajuste de puntos. (A) Objeto a muestrear;
(B) Objeto muestreado con rejilla 10X10;
(C) Objeto muestreado con rejilla 15X15;
(D) Objeto muestreado con rejilla 20X20.

Como se mostró anteriormente el muestreo uniforme con rejilla es muy útil ya que de ese modo podemos controlar la resolución del modelo que deseamos generar, pero si queremos una buena resolución para representar el exterior del objeto también tenemos que incrementar la resolución en el interior del modelo; esto nos genera finalmente un modelo que es muy detallado, pero también muy denso. Para evitar este problema se puede recurrir al muestreo adaptivo, como se verá en la siguiente sección.

3.2 Muestreo adaptivo

En la sección anterior se explico la manera en que llevamos a cabo el muestreo regular, ahora analizaremos como realizar el muestreo adaptivo con base en mapas de distancia para controla la densidad del muestreo; cabe mencionar que es posible combinar ambos métodos, pero en esta tesis hemos decido no conjuntarlos.

En este tipo de muestreo se puede controlar la resolución del modelo de modo tal que se obtenga una mejor representación del objeto en las partes que así lo requieran, que para nuestro caso son los bordes, y una menor resolución para las demás partes, que en nuestro caso es el interior del objeto a modelar.

Para poder controlar la resolución al momento de realizar el muestreo del objeto empleamos una herramienta conocida como mapa de distancia, este permite que controlemos la densidad del modelo en función de la distancia a los bordes del objeto. La manera en que podemos controlar esta densidad está dada por una función de transferencia en función del mapa de distancia obtenido

3.2.1 Mapa de distancia

El mapa de distancia de una imagen es el conjunto de valores asignados a los pixeles de una imagen, donde estos representan la distancia mínima del pixel al borde de interés (contorno o forma). El mapa de distancia normalizado puede representar la asignación de un valor de probabilidad a los pixeles pertenecientes a una región válida; dicha región forma parte de una imagen binaria, de modo que solo existan dos regiones, una valida y otra no valida. Así, el valor asignado a cada pixel está en función inversa a la distancia, si la distancia al borde disminuye entonces el valor de probabilidad aumenta.

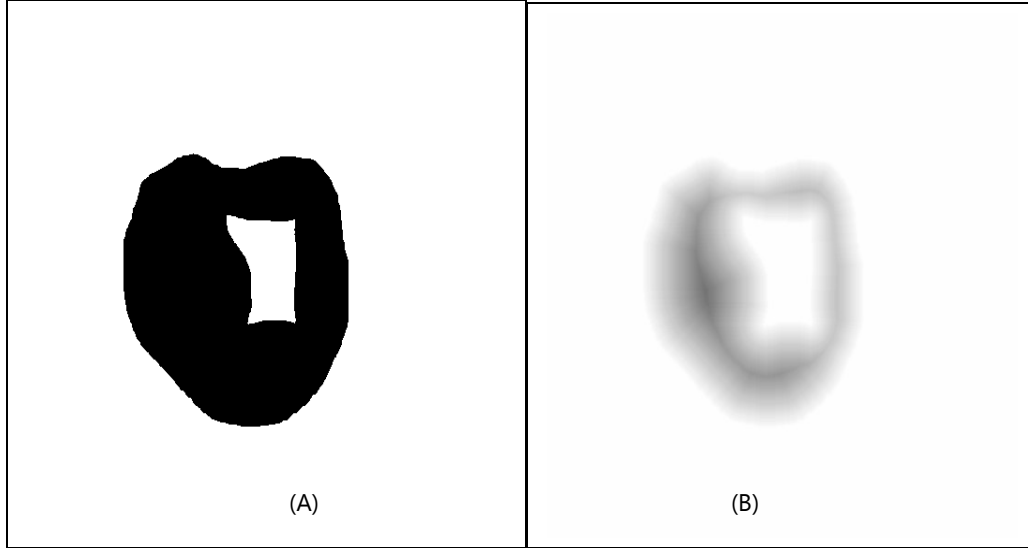


Figura 3.5 Mapa de distancia con función de probabilidad moduladora del muestreo.
(A) Objeto a obtener Mapa de Distancia (Región válida en negro); (B) Mapa de Distancia del objeto.

En la figura 3.5 (B) se observa el mapa de distancia obtenido sobre una imagen binaria, en el que se aprecia que entre más alejado esté un pixel de la región del borde, menor será el valor asignado, de modo que el pixel será más oscuro, y entre más cerca esté de la región el pixel será más blanco. A partir de estos valores podemos definir una resolución dependiente de la probabilidad al momento realizar el muestro del objeto.

Antes de comenzar con el proceso de muestreo adaptivo hay que realizar algunos ajustes al mapa de distancia: 1) agregar el borde de la imagen; 2) normalización del mapa de distancia para poderlo utilizar como una función de distribución de probabilidad de muestreo.

El intervalo de valores que deseamos es de $[0,1]$ en términos de probabilidad y $[0,255]$ en escala de grises de 8 bits; el nivel 255 es el borde, el nivel 0 el fondo y la región correspondiente al mapa de distancia normalizado tiene el intervalo de $[1,254]$; formándose así tres regiones. Esto es así porque requerimos identificar tres tipos de regiones: una no válida y dos válidas. Las regiones válidas son conformadas por el borde del objeto y el interior del objeto. (figura3.6)

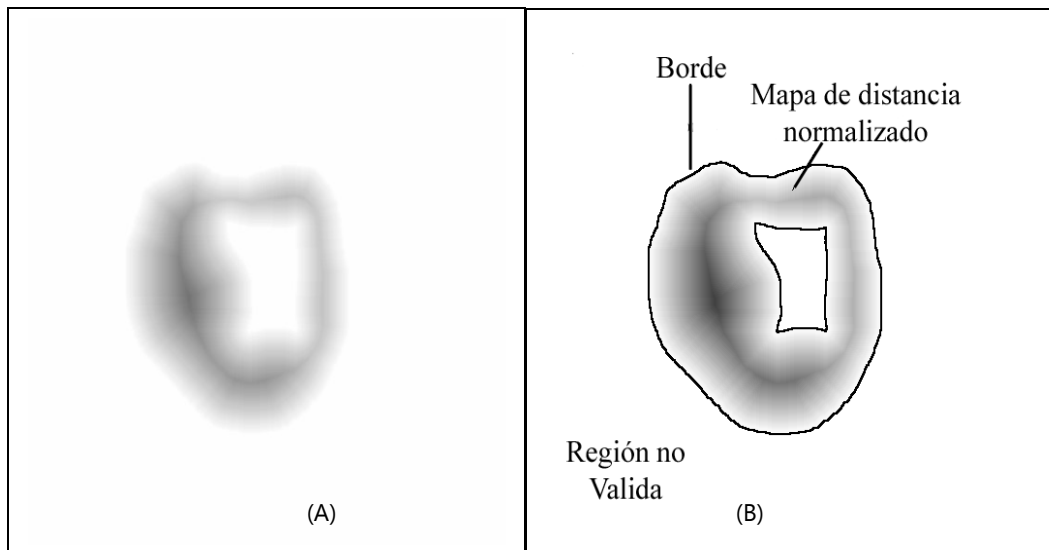


Figura 3.6 Mapa de distancia normalizado con tres regiones identificadas.

(A) Mapa de Distancia del objeto de interés;

(B) Mapa de Distancia normalizado con bordes (negativo).

Una vez que hemos generado el mapa de distancia normalizado con bordes ya podemos realizar el proceso de muestreo adaptivo. Este proceso al igual que el muestreo uniforme con rejilla debe darnos la posibilidad de etiquetar los puntos muestreados con el fin de poderlos identificar posteriormente.

El primer paso en este tipo de muestreo es definir el intervalo de muestreo que queremos manejar, esto lo logramos al definir dos radios, un radio mínimo de muestreo (r_{min}) y un radio máximo de muestreo (r_{max}). Estos radios nos definirán la validez de un punto tal como lo hacía la rejilla. Los radios definen una región para cada punto R_i ; en esta región no puede haber ningún otro punto, solo puede estar el punto central, entonces para que se pueda validar un punto su región no debe incluir a ningún punto previo; los puntos solo poseen una región para poder ser validados, una vez validados esta región deja de tomarse en cuenta. (figura3.7)

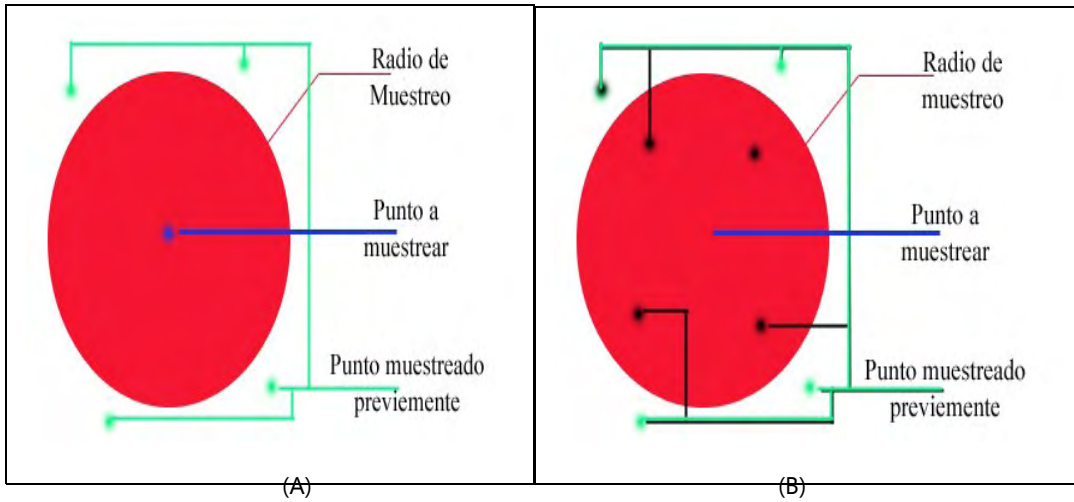


Figura 3.7 Validación de puntos.
 (A) Punto a muestrear válido; (B) Punto a muestrear no valido.

Una vez definidos los radios de muestreo el siguiente paso es muestrear el contorno (figura 3.8). Para evitar problemas posteriores este es muestreado usando r_{min} y la manera como se muestrea se define formalmente de la siguiente manera:

Sea P el conjunto de puntos muestreados y etiquetados con coordenadas (x,y) , y sea $p(x,y)$ un punto que pertenezca al contorno C del objeto, entonces

$$P = \{p(x,y) \mid (x+dx,y+dy) \cap P = \emptyset\} \quad \forall dx,dy = [-r_{min}, r_{min}] \quad \dots(3.6)$$

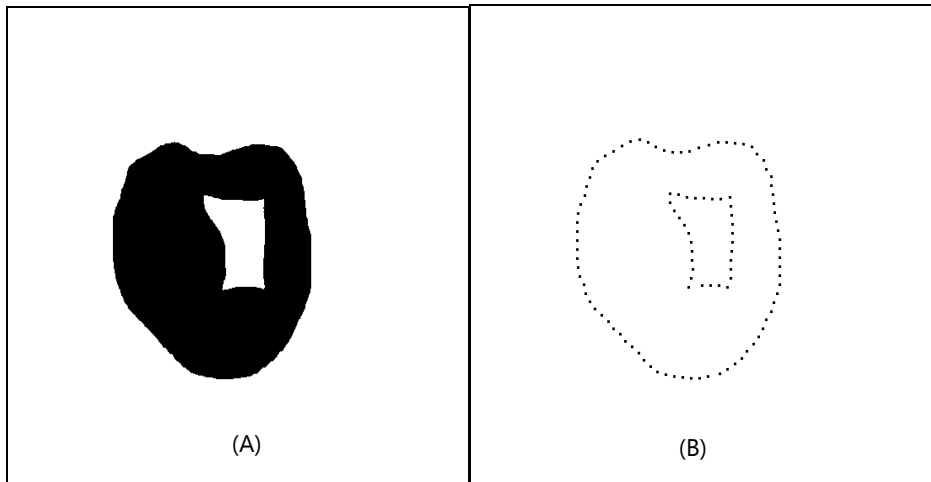


Figura 3.8 Muestreo adaptivo de bordes.
 (A) Objeto a muestrear; (B) Borde muestreado.

Ya que hemos muestreado los puntos pertenecientes al contorno lo siguiente es muestrear los puntos internos. Como ya mencionamos antes en el muestreo adaptivo la resolución varía, y esa resolución la variamos con ayuda del mapa de distancia: el radio de muestreo del píxel depende de su valor de probabilidad obtenido del mapa de distancia normalizado. Para asignar estos valores a cada píxel emplearemos una tabla de búsqueda, en la cual almacenaremos los distintos radios que usaremos dependiendo del valor de probabilidad, estos radios los calculamos empleando una función de transferencia.

Las funciones de transferencia que probamos en esta tesis son las siguientes:

$$\text{Función logarítmica } f(d) = \log(d) + b \quad \dots\dots(3.7)$$

$$\text{Función exponencial } f(d) = (1 - e^{-d}) + b \quad \dots\dots(3.8)$$

$$\text{Función lineal } f(d) = md + b \quad \dots\dots(3.9)$$

Donde d es el valor del píxel dado por el mapa de distancia normalizado y b es el radio mínimo, cabe mencionar que el dominio de estas funciones es de 1 a 254, y las funciones están normalizadas para que su valor máximo sea el radio máximo. Las gráficas que modelan los radios de muestreo se muestran en la figura 3.9.

La forma de realizar el muestreo es casi la misma que para el contorno con la diferencia de que aquí el radio está dado por la tabla de búsqueda precalculada, esto lo podemos expresar de la siguiente manera:

Sea P el conjunto de puntos muestreados y etiquetados con coordenadas (x,y) , sea $q(x,y)$ un punto que pertenezca a la región válida del objeto, y sea $V(q(x,y))$ el valor del radio de muestreo obtenido de la tabla de búsqueda de dicho punto, entonces

$$P = \{q(x,y) \mid (x+dx,y+dy) \cap P = \emptyset\} \quad \forall dx,dy=[-V(q(x,y)), V(q(x,y))] \quad \dots\dots(3.10)$$

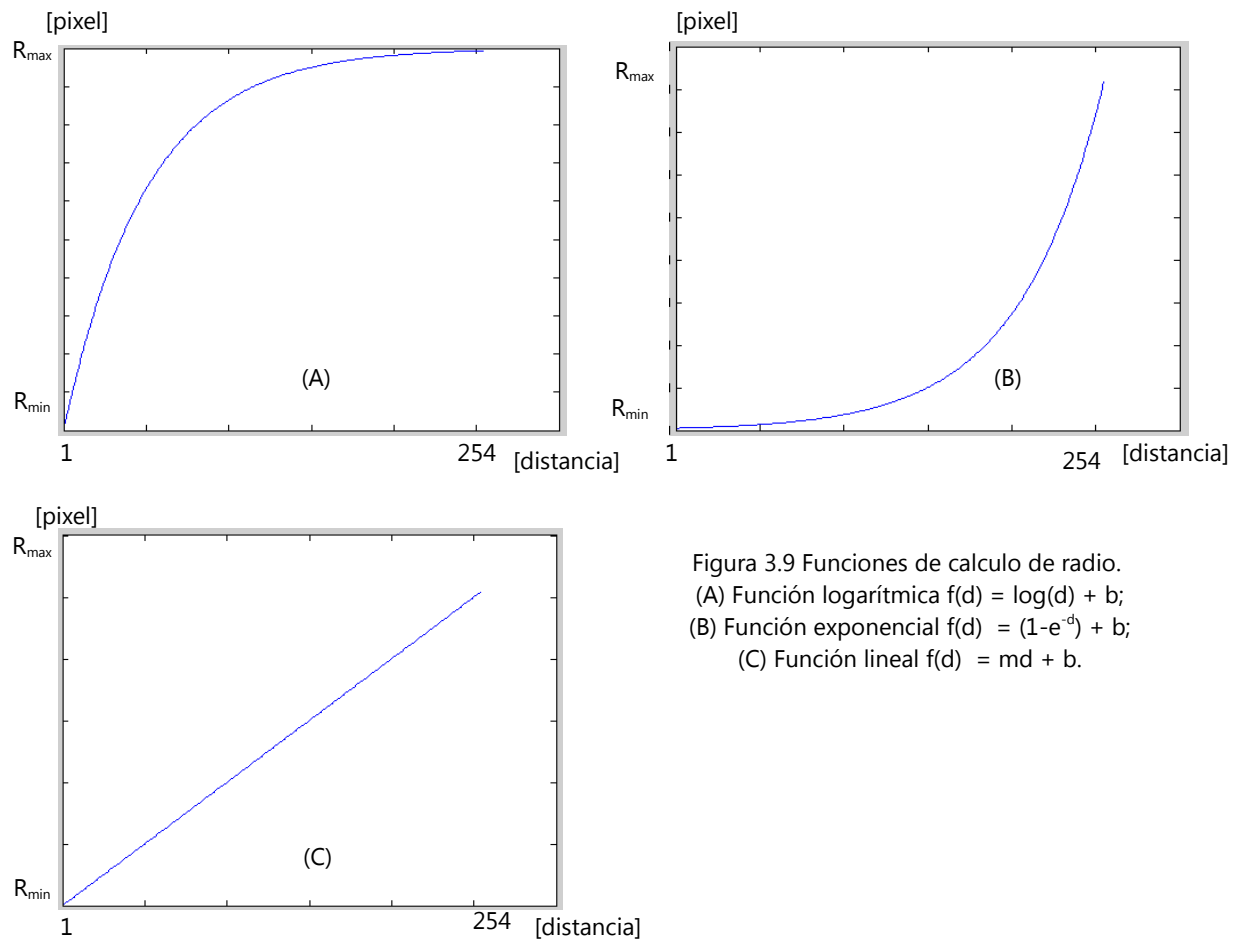


Figura 3.9 Funciones de calculo de radio.
 (A) Función logarítmica $f(d) = \log(d) + b$;
 (B) Función exponencial $f(d) = (1 - e^{-d}) + b$;
 (C) Función lineal $f(d) = md + b$.

Considerando que el mapa de distancia normalizado que se muestra en la figura 3.6 es el negativo, el valor de distancia máximo al borde calculado con el mapa de distancia es 254 y el valor de distancia mínimo al borde es 1, entonces entre más cerca se encuentre uno del borde el radio de muestreo será más cercano al radio mínimo y conforme uno se aleje del borde mas se acercara al radio de muestreo máximo.

En la figura 3.10 se observan los resultados obtenidos con las tres distintas funciones de transferencia, de entre estas la función que mejor se ajusta a nuestros requerimientos es la exponencial, debido a que el radio de muestreo cerca de los bordes es muy cercano al radio mínimo y conforme se aleja del borde este radio es más parecido al radio máximo, con lo que aseguramos que en los bordes tengamos suficiente resolución y en el resto del objeto el muestreo se realice a una resolución mucho menor.

En el caso de la función de transferencia lineal el problema que se observa es que es incapaz de muestrear zonas muy angostas o con bordes muy cercanos, pero si es una zona más amplia no tiene ese problema. Por otro lado la función de transferencia logarítmica entrega más muestras al interior del cuerpo que cerca de los bordes del objeto.

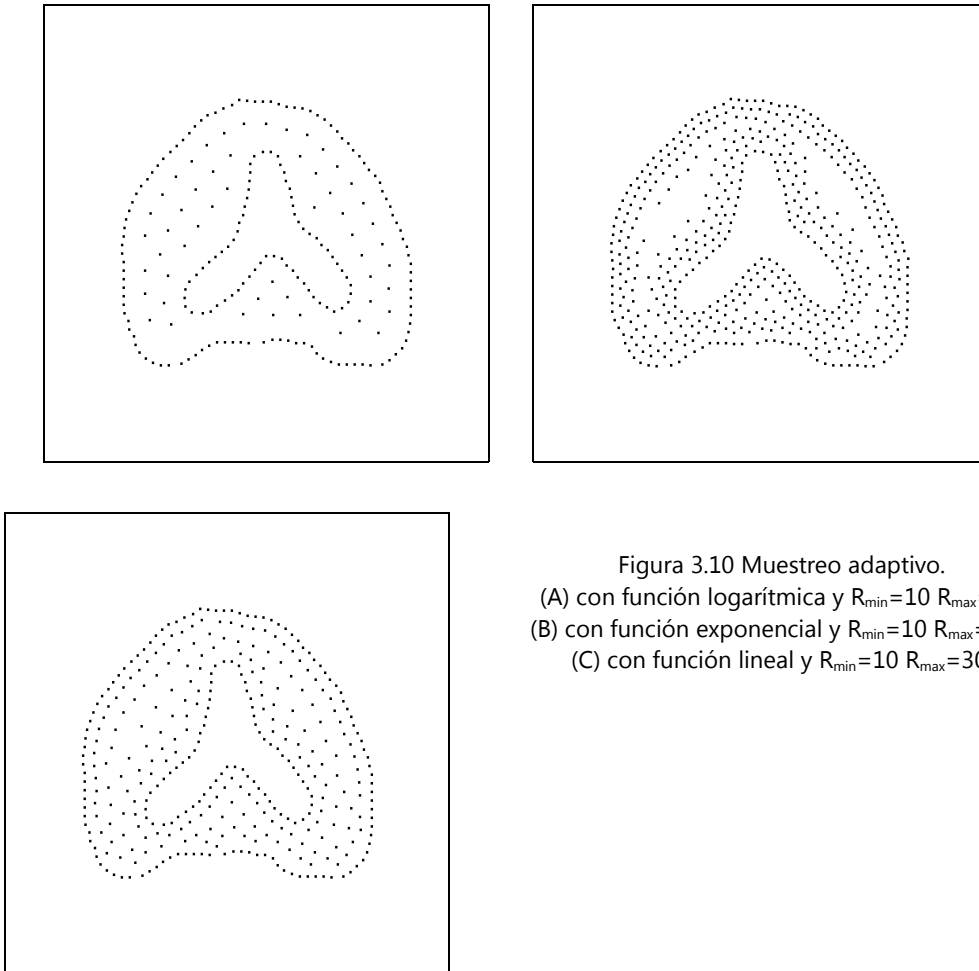


Figura 3.10 Muestreo adaptivo.
(A) con función logarítmica y $R_{\min}=10$ $R_{\max}=30$;
(B) con función exponencial y $R_{\min}=10$ $R_{\max}=100$;
(C) con función lineal y $R_{\min}=10$ $R_{\max}=30$.

Empleando este muestreo reducimos la cantidad de vértices que el modelo tendrá, haciéndolo menos denso y por lo tanto es más apto para realizar simulaciones en tiempo real, su único inconveniente es que para que realmente se aprovechen sus ventajas el objeto debe tener regiones grandes sin detalles, de lo contrario el proceso no resulta de gran ayuda, puesto que tiende a generar la misma cantidad de muestras que el muestreo uniforme con rejilla.

Capítulo 4. Reconstrucción 3D

En este capítulo se describirá el proceso mediante el cual se genera la malla tetraédrica que conforma el modelo del objeto. Esta malla se forma a partir de una nube de puntos la cual obtuvimos al muestrear cada una de las imágenes que generamos por medio de las técnicas de interpolación.

El proceso de reconstrucción en esta tesis consta de tres partes principales, las cuales son: tetraedrización, filtrado de artefactos y relajación del modelo. El proceso de relajado solo se aplica a los modelos generados mediante muestreo regular (rejilla), esto se debe a que al estar los nodos tan ordenados el modelo puede presentar anisotropías; esto no se presenta en el caso de muestreo adaptivo ya que al regular la densidad de muestreo mediante los campos de distancia es como si aplicáramos un relajado al mismo tiempo que muestreamos.

4.1 Tetraedrización

Antes de iniciar con la descripción del proceso de tetraedrización hay que realizar un reacomodamiento de los puntos que hemos muestreado, este reacomodamiento consiste en agrupar los puntos en una rejilla en forma de lattice 3D, este reacomodo solo se aplica a los nodos que se obtuvieron por medio del muestreo regular, en el caso de los nodos muestreado de manera adaptiva tan solo se hace una asignación de etiquetas. Este proceso de reacomodamiento nos será de gran utilidad no solo para el modelado sino también para la relajación.

Una vez armada la rejilla en forma de lattice en 3D (figura 4.1) se asigna a cada nodo de esta rejilla un punto del conjunto P que obtuvimos por medio del muestreo del capítulo 3 y cada nodo se etiqueta nuevamente de acuerdo a la información que obtuvimos del proceso de muestreo. En caso de que a algún nodo no se le pueda asignar un punto, este nodo se etiqueta como un punto no existente; de este modo solo tomaremos en cuenta los nodos que formen parte del volumen que nos interesa generar.

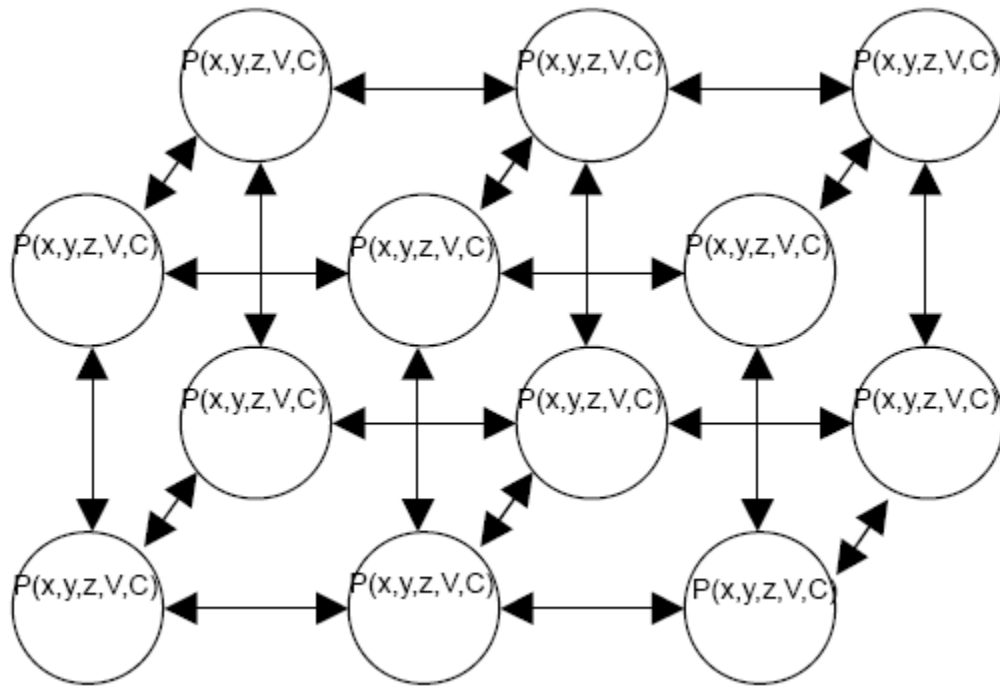


Figura 4.1 Estructura de la rejilla 3D. Cada nodo P tiene asignada una posición dada por x, y, z y dos etiquetas. La etiqueta V determina si el nodo existe en el volumen y la etiqueta C determina si forma parte del contorno del objeto. Las flechas son resortes que unen a los nodos.

En este lattice las flechas indican las uniones entre nodos, estas uniones son mediante resortes como se observa en la figura 4.7. Una vez construida la rejilla 3D, podemos proceder a realizar la Tetraedrización. Para generar los tetraedros que conforman la malla empleamos una herramienta de geometría computacional conocida como Tetraedrización de Delaunay.

4.1.1 Tetraedrización de Delaunay

La Tetraedrización de Delaunay nos es más que la expansión de la Triangulación de Delaunay para \mathbf{R}^3 . Este proceso nos asegura que cada nodo será empleado como mínimo en una ocasión para generar un tetraedro, también nos asegura que los tetraedros no se intersecan unos con otros. En este trabajo utilizamos la tetraedrización de Delaunay implementada en las cajas de herramientas de Matlab [Matlab].

La triangulación de Delaunay para un conjunto de puntos P en un plano es el conjunto de triángulos $TD(P)$ tales que se maximice el ángulo mínimo de los ángulos de todos los triángulos. Además para cada uno de los triángulos solo debe existir un círculo circunscrito el cual únicamente contenga los vértices de ese

triángulo como se observa en al figura 4.2, si llegara contener algún otro vértice entonces no es una triangulación de Delaunay [GeoTCG].

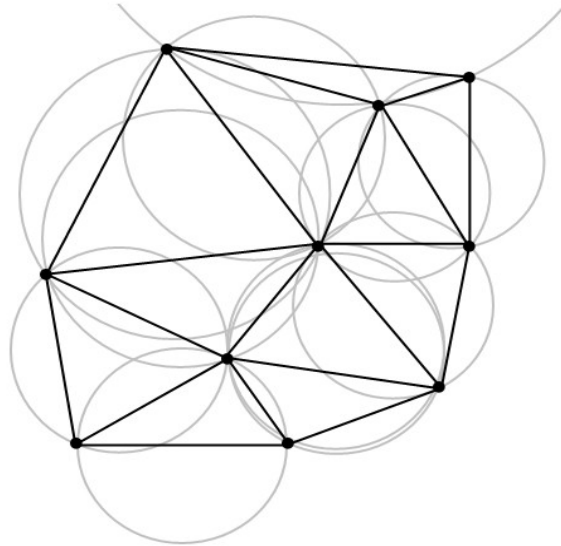


Figura 4.2 Triangulación de Delaunay.

La triangulación de Delaunay puede generalizarse para espacios de n dimensiones; para poder generar el modelo debemos emplear la triangulación de Delaunay para un espacio tridimensional también conocida como tetraedrización de Delaunay. Para resolver esta triangulación se debe encontrar la carcasa convexa de un conjunto de puntos de un espacio dimensional de orden $n+1$, donde n es el orden del espacio dimensional del conjunto P . Se agregará a cada punto de p una coordenada extra equivalente a $|p|^2$. Después se toma la parte inferior de la carcasa convexa y se mapea en un espacio de diemnsión n ignorando la última coordenada de los puntos.

La carcasa convexa de un conjunto de puntos P en n dimensiones es la intersección de todos los conjuntos convexos contenidos en P (figura 4.3). Para N puntos p_1, \dots, p_N , la carcasa convexa está dada por [MWCHull]:

$$C \equiv \left\{ \sum_{j=1}^N A_j p_j : A_j \geq 0 \text{ para toda } j \text{ y } \sum_{j=1}^N A_j = 1 \right\} \dots (4.1)$$

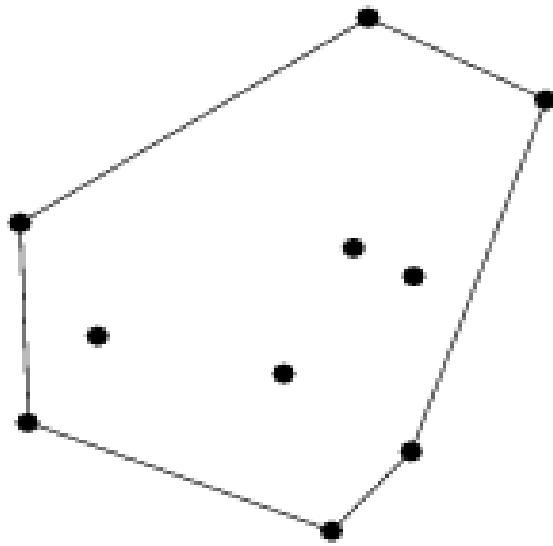


Figura 4.3 Carcasa convexa. [MWCHull]

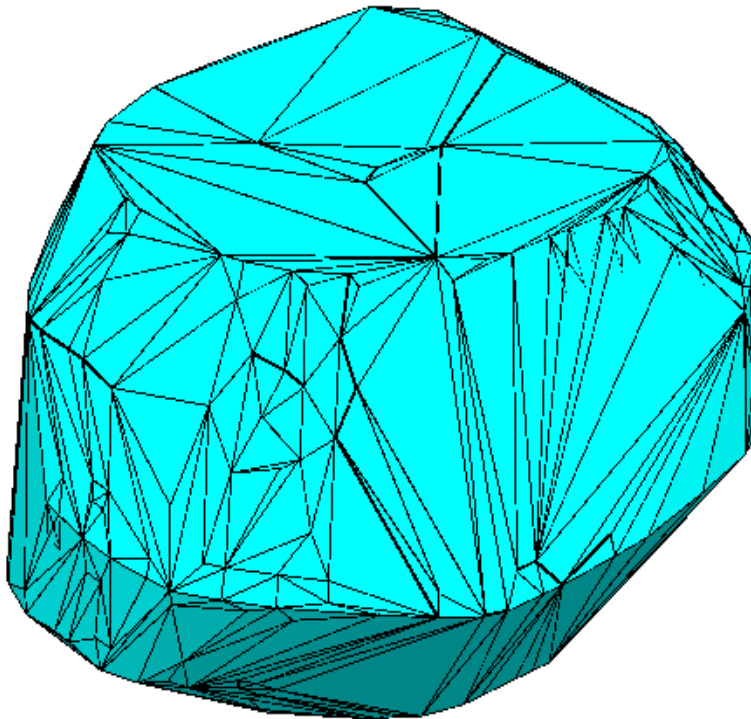


Figura 4.4 Carcasa convexa del volumen. Objeto generado a partir de la Tetraedrización de Delaunay.

4.2 Filtrado de artefactos

Como se observa en la figura 4.4, la Tetraedrización de Delaunay nos genera una malla en 3D conformada de tetraedros, en la cual se emplean todos los vértices que le proporcionamos, pero como esta tetraedrización toma como parámetro un casco convexo entonces nos genera un volumen completamente cerrado. Para lograr que el modelo generado sea consistente al objeto real entonces hay que filtrar los tetraedros de modo tal que al final solo tengamos aquellos que pertenecen al objeto.

Los tetraedros que no pertenecen al volumen que conforma al objeto se denominan artefactos, estos surgen debido a que se emplea una carcasa convexa para llevar a cabo la tetraedrización.

El filtrado de los artefactos se basa en las etiquetas de cada nodo, previamente identificadas en el proceso de muestreo del capítulo anterior. En el filtrado por medio de etiquetas se evalúa con qué tipo de puntos está formado cada tetraedro, es decir si el tetraedro está formado por puntos del contorno o puntos internos del objeto. Este criterio considera que si un tetraedro está formado exclusivamente por puntos del contorno entonces es un artefacto que debe ser eliminado del volumen, como se ilustra en 2D en la figura 4.5.

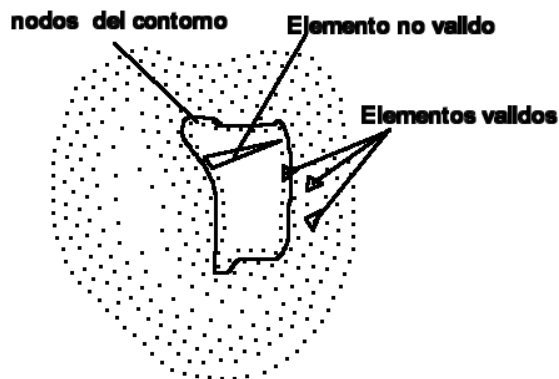


Figura 4.5 Validación de elementos por medio de sus etiquetas (los elementos encerrados fueron etiquetados como bordes).

Los puntos que se encuentran encerrados en la figura 4.5 pertenecen al borde del objeto y fueron etiquetados como tal, de modo que si se genera algún tetraedro empleando únicamente puntos etiquetados como borde entonces ese elemento no es válido y debe ser eliminado, si al menos uno de los puntos que forman el tetraedro no es borde entonces el elemento es válido. En la figura 4.6 se aprecia el resultado de filtrado de los artefactos por etiqueta.

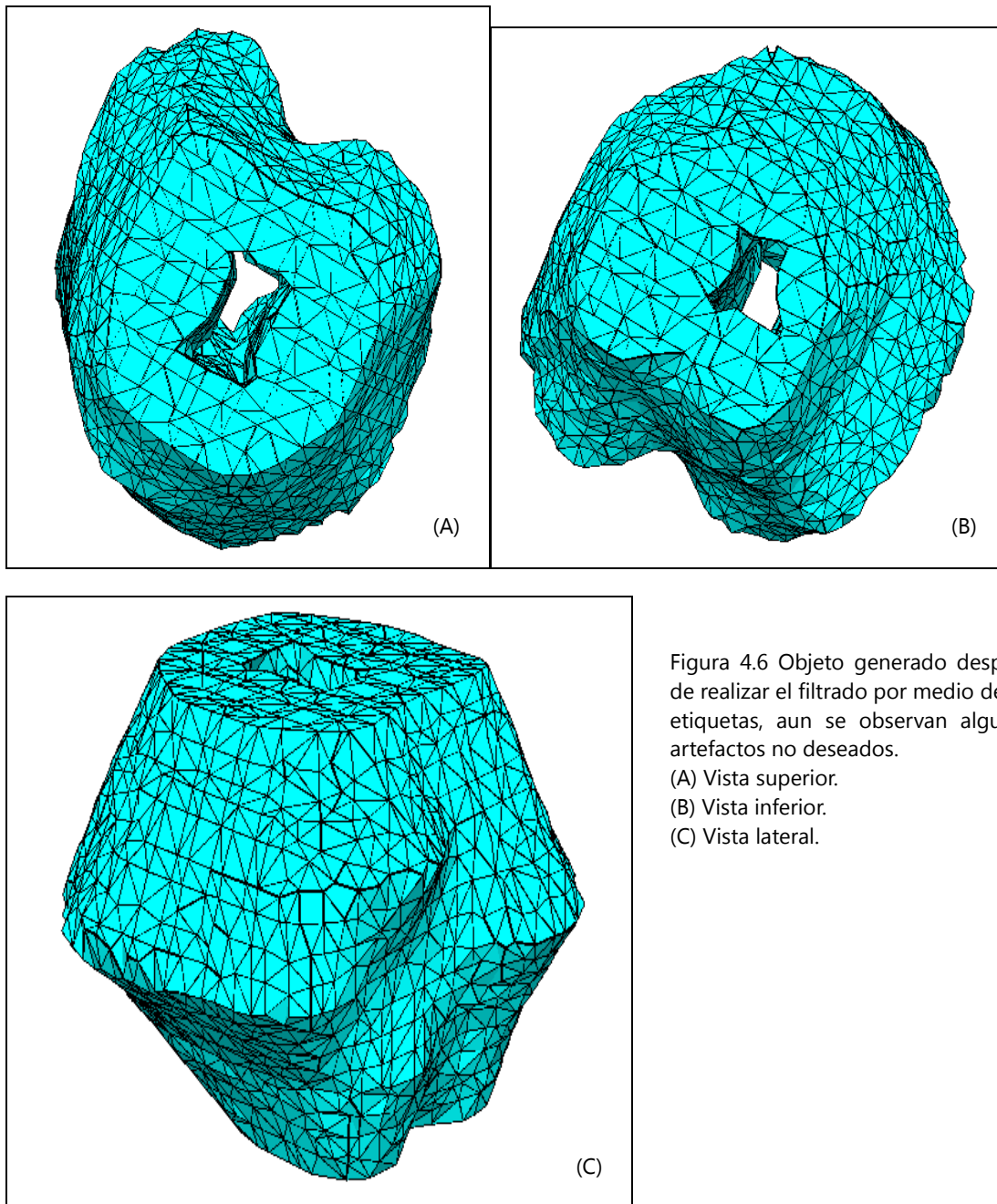


Figura 4.6 Objeto generado después de realizar el filtrado por medio de las etiquetas, aun se observan algunos artefactos no deseados.

- (A) Vista superior.
- (B) Vista inferior.
- (C) Vista lateral.

Al aplicar el filtrado podemos observar que nuestro volumen generado se asemeja más al objeto que deseamos modelar, pues ya no es una malla exclusivamente convexa, en este caso se eliminaron casi todos los artefactos no deseados. Sin embargo, aun existen artefactos que son muy difícil de eliminar de manera automática, por lo tanto para obtener un mayor éxito al eliminar estos artefactos se puede recurrir al empleo de mas criterios de eliminación, pero si estos criterios no se diseñan de manera adecuada puede terminar eliminando tetraedros que no sean artefactos.

4.3 Relajación

Una vez construido el modelo procedemos a realizar un reacomodo de los nodos que conforman el modelo, hay que recordar que este proceso solo se lleva a cabo con los modelos que fueron obtenidos mediante el muestreo regular, esto es con el fin de que los puntos se distribuyan de manera más uniforme sobre el modelo y así se eviten anisotropías; además de este modo los tetraedros sean más uniformes.

Puesto que el objetivo de este proceso es solo distribuir de mejor manera los nodos sin afectar la forma del objeto entonces el relajado se realiza tomando en cuenta los bordes de nuestro modelo como puntos de atracción. Para ellos utilizamos un modelo deformable conocido como masas y resortes. Este proceso consiste en aplicar fuerzas externas a los nodos de modo que estos se ajusten a los bordes.

En la figura 4.8 se aprecia cómo se distribuyen los nodos de manera más uniforme sobre el modelo y al mismo tiempo la forma de los tetraedros se normaliza; por otro lado la forma externa del objeto no cambia de modo que solo se ajusta la estructura interna.

4.3.1 Modelo de masas y resortes

Para realizar la relajación fue elegido un método langragiano de cálculo de deformaciones conocido como modelo de masas y resortes, ya que su implementación es muy sencilla. Con el modelo generado podemos asociar masa, rigidez y viscosidad a cada uno de los nodos que conforma el modelo. Cada nodo representa un punto de masa la cual ésta interconectada con sus vecinos por medio de resortes, los cuales se encuentran en un medio viscoso. La ecuación empleada en este trabajo y que ha sido utilizada anteriormente con éxito en [DefMTurp] es la siguiente:

$$m_i \frac{d^2 x_i}{dt_i^2} + \gamma_i \frac{dx_i}{dt_i} + g_i(t, x_i) = f_i(t, x_i) \quad \dots (4.2)$$

Donde m_i es la masa del nodo i en la malla, con coordenadas cartesianas x_i ; γ_i es el coeficiente de viscosidad del medio; g_i es el vector de fuerza elástica interna; y f_i representa todas las fuerzas externas que actúan sobre el nodo i . En la figura 4.7 se muestra la forma en la que conectamos los nodos del modelo [NIMDOS].

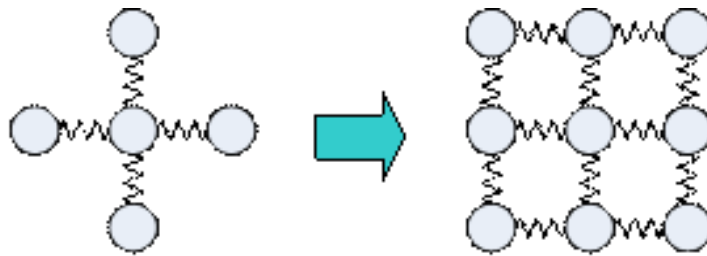


Figura 4.7 Conexión de nodos para relajado [NIMDOS]

Para poder realizar la relajación necesitamos resolver la ecuación 4.2, para resolverla recurrimos al método de Euler.

4.3.1.1 Método de Euler

El método de Euler es una de las técnicas más simples para aproximar la solución de una ecuación diferencial. Se utiliza para obtener una solución para un problema de valor inicial.

Supongase $y' = F(x, y)$, con $y(x_0) = y_0$ y supóngase $x_{n+1} = x_n + h$, donde h es un incremento relativamente pequeño.

La solución del método es:

$$y_n = y_{n-1} + h * F(x_{n-1}, y_{n-1}) \quad \dots (4.3)$$

Para resolver el cálculo de fuerzas internas se programo el sistema de fuerzas en base a las siguientes ecuaciones:

$$g_i = \sum s_k \quad \dots (4.4)$$

$$s_k = \frac{c_k \cdot e_k}{\|r_k\|} \cdot r_k \quad \dots (4.5)$$

$$e_k = \left\| r_k \right\| - l_k \quad \text{.....(4.6)}$$

$$r_k = x_j - x_i \quad \text{.....(4.7)}$$

Donde s_k es la fuerza interna en el nodo i causada por el resorte k , c_k denota el coeficiente de elasticidad del resorte k , e_k es el valor absoluto de la deformación desde su posición inicial (l_k), r_k es el vector de distancia entre el nodo i y su vecino j , $|r_k|$ es el valor absoluto de la distancia del resorte k , y x_j denota la posición cartesiana del vector para el nodo j , el cual está conectado por el resorte k con el nodo i .

El método explícito de Euler es inestable para modelar materiales con una rigidez muy alta, para evitar esta inestabilidad se puede optar por el método implícito de Euler o por reducir el tamaño de h en la ecuación 4.8. Reducir h tiene un impacto muy grande en la velocidad de simulación, ya que implica que se realicen más cálculos. Por otro lado la implementación del método implícito de Euler es bastante complicado, por lo que el método explícito es la mejor opción para simulaciones en tiempo real de tejidos, esto fue demostrado por Kühnaphel et al. [Kühnaphel]. La solución de la ecuación 4.2 se obtiene por medio del método explícito y el tamaño de h es determinado por el criterio de Courant. [DefMTurp]

$$h \geq \frac{1}{2\pi^2 \mu \gamma_d} \quad \text{.....(4.8)}$$

Donde μ es el coeficiente de rigidez del sistema y γ es el coeficiente de viscosidad.

En la figura 4.9 se observa el resultado del proceso de relajado en 2D, mientras que en la figura 4.8 se observa la comparación entre los modelos no relajados, relajados y adaptivos. Este proceso nos permite una mejor distribución de los nodos, de modo que se evitan anisotropías.

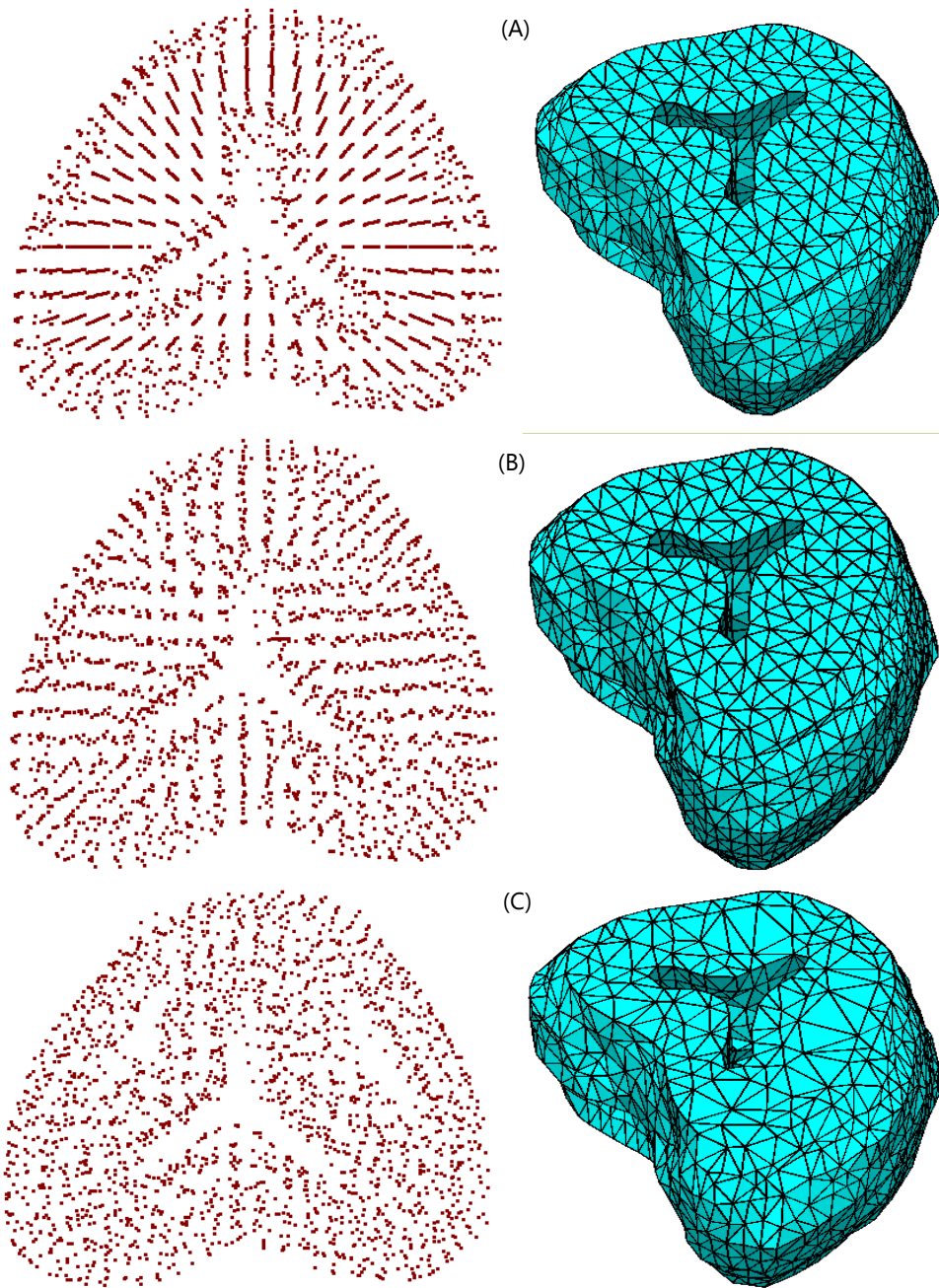
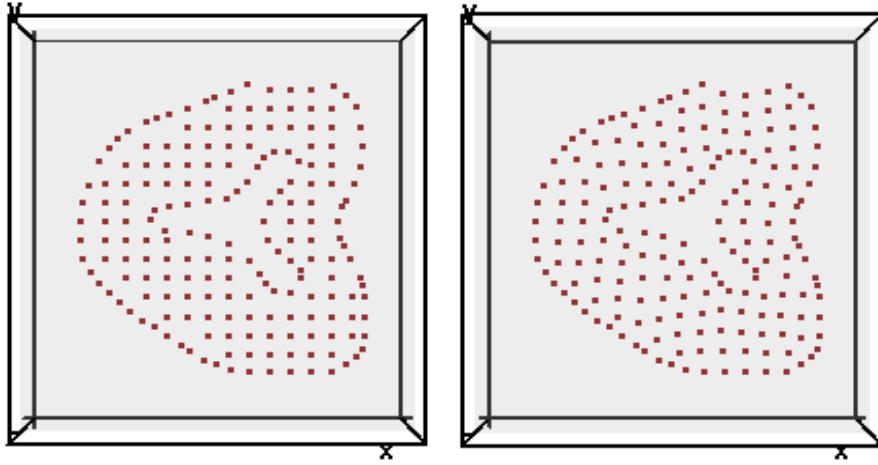


Figura 4.8 Comparación entre modelos.
(A) Modelos con muestreo regular sin relajar (2222 nodos),
(B) Modelo don muestreo regular relajado (2222 nodos),
(C) Modelos con muestreo adaptivo (1926 nodos).



(A)

(B)

Figura 4.9 Relajado (corte)
(A) Nodos sin relajar (B) Nodos relajados

Capítulo 5. Resultados

A partir de las técnicas de interpolación, muestreo, reconstrucción y relajado, que se describieron en los capítulos anteriores de esta tesis, se generaron modelos de mallas tetraédricas empleando las siguientes combinaciones de técnicas:

- (A) Muestreo regular con *splines* sin relajación;
- (B) Muestreo regular con *media morfológica* sin relajación;
- (C) Muestreo adaptivo con *splines* sin relajación;
- (D) Muestreo adaptivo con *media morfológica* sin relajación;
- (E) Muestreo regular con *splines* con relajación;
- (F) Muestreo regular con *media morfológica* con relajación.

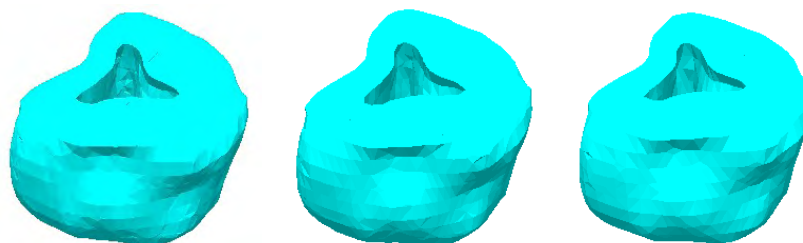
De estos modelos se consideraron los siguientes parámetros de comparación:

- (A) Número de nodos;
- (B) Número de tetraedros;
- (C) Volumen total generado;
- (D) Volumen promedio de los tetraedros;
- (E) Desviación estándar del volumen de los tetraedros;
- (F) Tamaño del Grid (Horizontal[rmin,rmax], Vertical).

5.1 Modelos

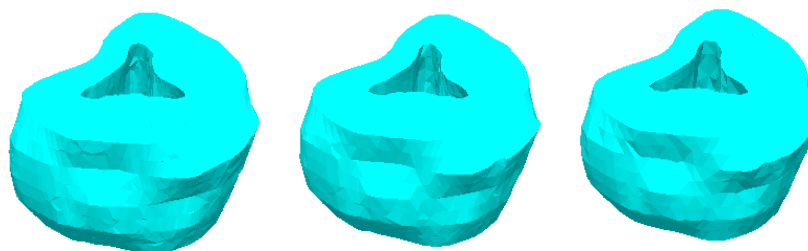
Los modelos se generaron a partir de imágenes de 512x512 pixeles, y se considero una profundidad de 256 pixeles.

5.1.1 Muestreo regular con *splines* sin relajación



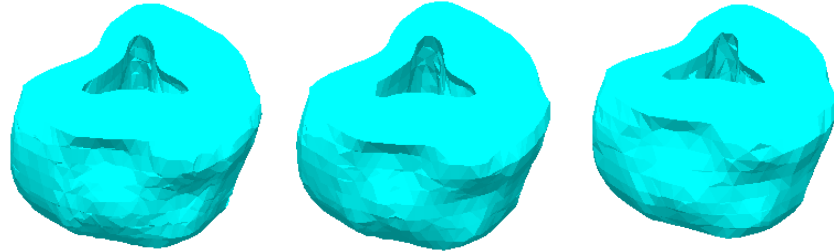
<i>Rejilla (X,Y,Z)</i>	(10,10,32)	(15,15,32)	(20,20,32)
# nodos	8252	3902	2308
# elementos	38548	17417	9803
<i>Volumen Total (voxel)</i>	20223270	20219690	20181510
<i>Vol. Promedio (voxel)</i>	524.6256	1160.917	2058.707
σ	172.3244	374.3942	688.1102

5.1.2 Muestreo regular con *media morfológica* sin relajación



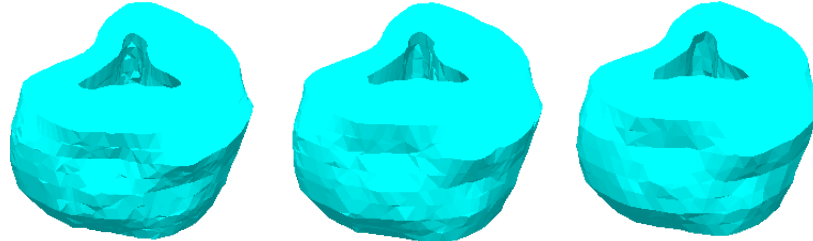
<i>Rejilla (X,Y,Z)</i>	(10,10,32)	(15,15,32)	(20,20,32)
# nodos	7956	3772	2222
# elementos	37018	16868	9430
<i>Volumen Total (voxel)</i>	19520900	19511620	19529110
<i>Vol. Promedio (voxel)</i>	527.3354	1156.724	2070.956
σ	174.2474	375.3159	669.2989

5.1.3 Muestreo adaptivo con *splines* sin relajación



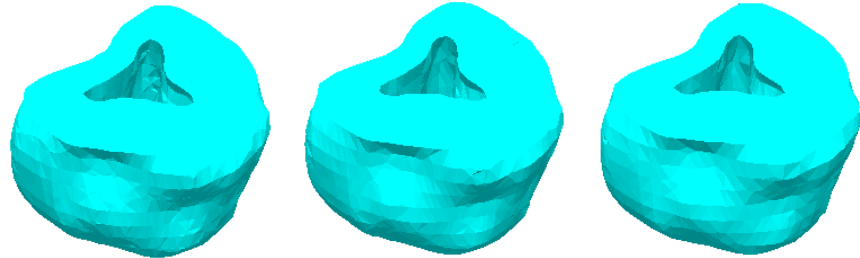
<i>Rejilla (X,Y,Z)</i>	(10,10,32)	(15,15,32)	(20,20,32)
# nodos	7390	3367	1993
# elementos	38095	16390	9098
<i>Volumen Total (voxel)</i>	20238060	20233690	20162700
<i>Vol. Promedio (voxel)</i>	531.2524	1234.514	2216.168
σ	404.0659	848.4244	1324.118

5.1.4 Muestreo adaptivo con *media morfológica* sin relajación



<i>Rejilla (X,Y,Z)</i>	(10,10,32)	(15,15,32)	(20,20,32)
# nodos	7093	3243	1926
# elementos	36533	15831	8882
<i>Volumen Total (voxel)</i>	19538430	19534460	19548430
<i>Vol. Promedio (voxel)</i>	534.8158	1233.937	2200.904
σ	399.4261	847.5083	1316.281

5.1.5 Muestreo regular con *splines* con relajación



<i>Rejilla (X,Y,Z)</i>	(10,10,32)	(15,15,32)	(20,20,32)
# nodos	8252	3902	2308
# elementos	38548	17417	9803
Volumen Total (voxel)	20086830	20061970	20002880
Vol. Promedio (voxel)	521.0863	1151.861	2040.485
σ	167.6403	382.5616	693.5050

5.1.6 Muestreo regular con *media morfológica* con relajación



<i>Rejilla (X,Y,Z)</i>	(10,10,32)	(15,15,32)	(20,20,32)
# nodos	7956	3772	2222
# elementos	37018	16868	9430
Volumen Total (voxel)	19329890	19290770	19334690
Vol. Promedio (voxel)	522.1755	1143.631	2050.338
σ	170.3404	381.6283	682.8954

5.2 Análisis de resultados

Como se observa en la figura 5.1, el muestreo adaptivo reduce el número de nodos de un 10% hasta un 14%. Aparentemente no es una ganancia muy significativa, pero se logra sin afectar negativamente la calidad visual del modelo, pero hay que considerar que en este trabajo el muestreo adaptivo se realizó sólo en 2D por corte, por lo que se piensa que hacerlo en 3D reducirá significativamente la complejidad de la malla; por otro lado dado que tiende a incrementar el muestreo en los bordes se mejora la calidad visual.

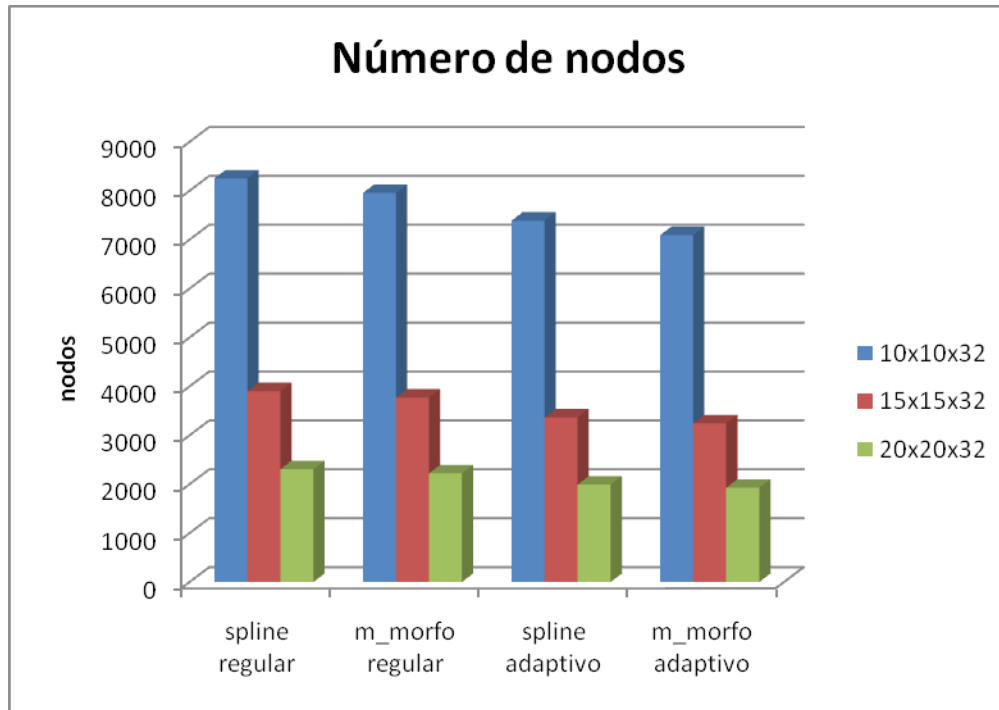


Figura 5.1 Número de nodos.

Por otro lado, en la figura 5.2 observamos que los volúmenes generados no son iguales. Los modelos generados con las imágenes interpoladas por medio de *splines* poseen un mayor volumen que los que fueron generados con la media morfológica. Esto se debe a que las *splines* generan un modelo más suavizado que el generado por la media morfológica, lo cual se lo atribuimos al hecho de que la media morfológica la calculamos empleando un elemento estructurante cuadrado; el resultado de la media morfológica se mejoraría si empleáramos un campo de distancia euclidiano.

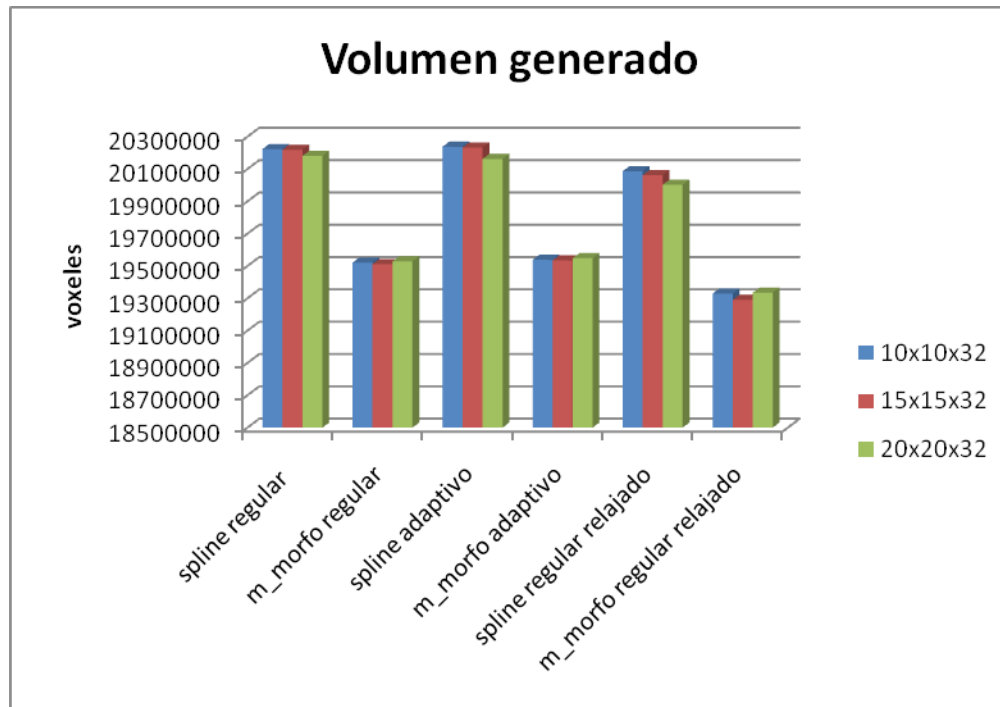


Figura 5.2 Volumen generado.

Por ultimo los tetraedros son mas uniformes cuando se emplean los métodos de muestreo regular y esta uniformidad incrementa conforme se incrementa la resolución de muestreo. En el caso del muestreo adaptivo es de esperarse que los tetraedros no sean uniformes y que esto se vea más acentuado conforme se incremente el radio de muestreo.

Cuando relajamos el modelo redistribuimos los nodos para que se ajusten a la forma de los contorno, en el caso del muestreo regular con un ancho de 10 pixeles el proceso efectivamente modifica los tetraedros haciéndolos más uniformes, pero conforme se incrementa el ancho de la rejilla de muestreo el relajado ya no genera tetraedros más uniformes, sino todo lo contrario. En el caso del muestreo adaptativo no es necesario aplicar el relajado, dado que el método de muestreo distribuye los nodos a lo largo del volumen del objeto, lo cual reduce anisotropías indeseadas en aplicaciones como elemento finito, debidas a la geometría del objeto.

Las curvas de la figura 5.3 muestran la distribución del volumen de los tetraedros para los distintos modelos generados. Se observa que el factor determinante en cuanto a la conformación de la malla se debe en gran medida al método de muestreo, así como a la resolución empleada del mismo. Por otro lado se observa que al aplicar el relajado, solo en los casos en que se empleó una resolución alta con un muestreo regular se obtuvieron tetraedros mas uniformes.

Para resoluciones mayores empleando la misma técnica de muestreo, el relajado no vuelve más uniformes los tetraedros, pero tampoco hace lo contrario, como ocurre en los modelos generados empleando el muestreo adaptivo.

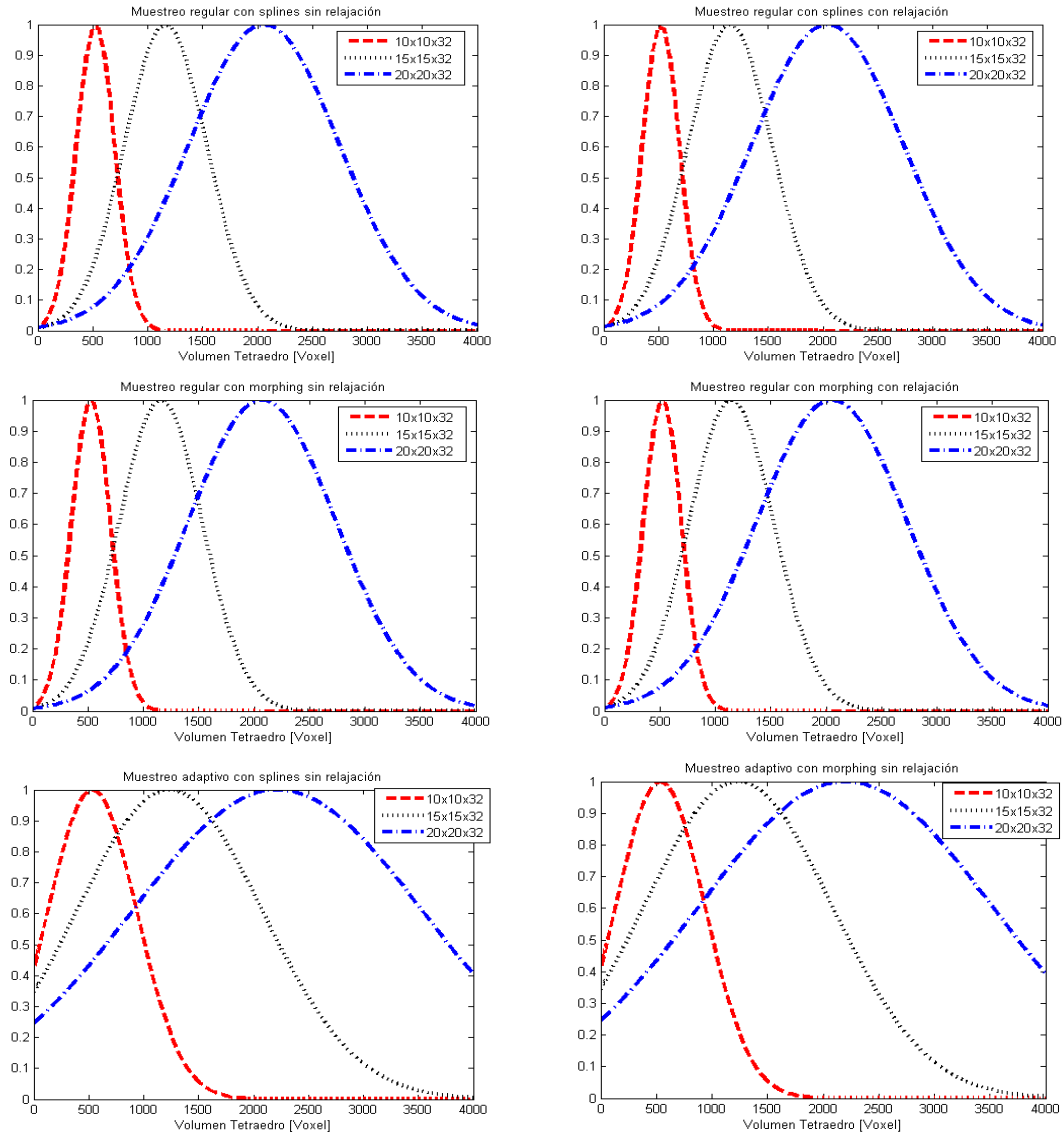


Figura 5.3 Distribución del volumen de los tetraedros.

En la figura 5.4 (A) se muestra la composición de los modelos generados empleando interpolación morfológica y muestreo regular, primero se muestra la diferencia entre un muestreo con resolución de 10 pixeles contra otro de 20 pixeles, posteriormente en la figura 5.4(B) se observan los mismos modelos después de haber sido relajados. Por último en la figura 5.4 (B) se muestra el mismo ejemplo pero empleando el muestreo adaptivo.

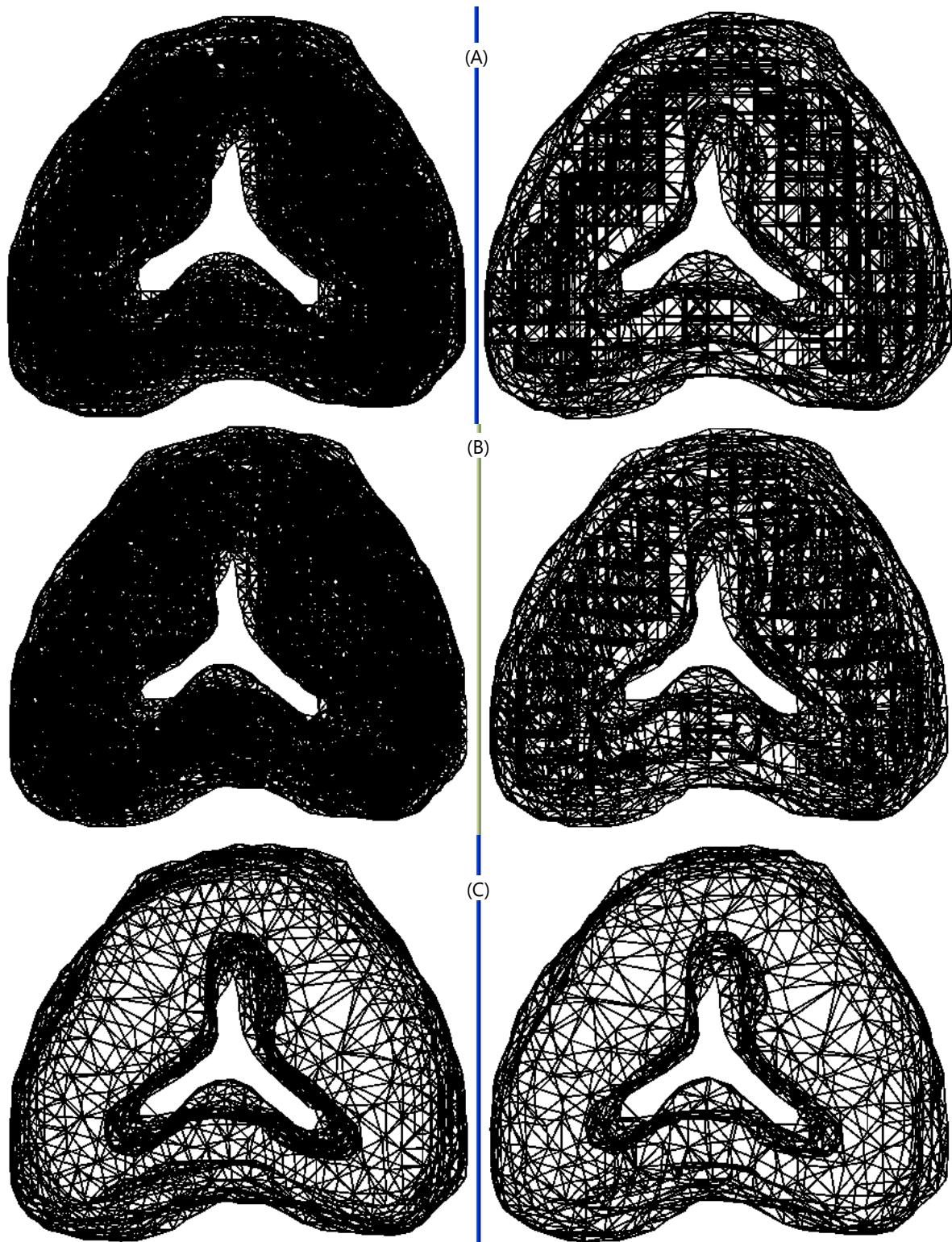


Figura 5.4 Composición de las mallas. (A) Interpolación morfológica con muestreo regular con rejilla de 10 y 20 píxeles, (B) Interpolación morfológica con muestreo regular con rejilla con relajación de 10 y 20 píxeles, (C) Interpolación morfológica con muestreo adaptivo con r_{\min} de 10 y 20 píxeles.

Conclusiones

- En esta tesis se propuso un método por el cual se pueden generar mallas de tetraedros para aplicaciones que necesitan discretizar medios continuos, como los métodos de elemento finito. El método permite obtener modelos con topología distinta a través de sus secciones, además es posible automatizar el método que se ha propuesto, de modo que solo se necesite expresar la modalidad de muestreo (regular o adaptivo), el método de interpolación (*splines* o por morfología) y la resolución del modelo que se desea generar. El método permite reducir mediante muestreo adaptivo la complejidad de la malla para aplicaciones que requieren cálculo en tiempo real.
- Dentro del proceso de interpolación de imágenes se propusieron dos métodos. El primero de ellos emplea *splines* para realizar la interpolación, el método entrega buenos resultados siempre que los contornos a interpolar tengan una forma similar. Esto se debe a la forma de obtener las correspondencias para el vector de nodos, si se definiera otra forma de obtener las correspondencias el método podría funcionar aunque los contornos no fueran similares. Esto también se puede complementar con un re-muestreo de los contornos a fin de obtener más información para realizar la interpolación.
- La interpolación morfológica nos permite generar interpolar imágenes aunque la forma y topología de las regiones no sea parecida, pero tiene el inconveniente de que necesita que las intersecciones de estas regiones no sea el conjunto vacío. Este problema se puede resolver realizando un registro de la imagen. Esto consiste en generar una función de transformación que consta de tres matrices, una de escalamiento, otra de traslación y otra más de rotación. Así se obtiene la media morfológica que después se puede transformar utilizando la transformación inversa. Además se puede cambiar el elemento estructurante para poder obtener imágenes más suavizadas.

- Los métodos de muestreo adaptivo reducen la cantidad de puntos e incrementan la resolución del modelo en las regiones de interés, sin embargo este método se puede mejorar si el mapa de distancia se implementara a lo largo del volumen (en 3D) y no solo en los cortes (2D). También se pueden probar más funciones de transferencia a fin de obtener mejores resultados.
- El método de muestreo regular es bastante rápido y al agregar el ajuste a los bordes se mejora la calidad del modelo generado.
- Finalmente se puede decir que el proceso de filtrado de artefactos es lo suficientemente bueno, pero no es 100% confiable, ya que algunos artefactos son muy difíciles de quitar, ya sea porque no existen las suficientes muestras o porque el artefacto es muy difícil de identificar. Este problema se presenta sobre todo en el caso de regiones muy oblicuas o en forma de "V", en regiones más suavizadas el problema casi no se presenta y es posible eliminarlo casi en un 100%.

Bibliografía

- [AMeshGen] Chun-Li Lin, Chih-Han Chang, Chia-Shin Cheng, Chau-Hsiang Wang, Huey-Er Lee, "Automatic finite element mesh generation for maxillary second premolar". Computer Methods and Programs in Biomedicine, Volume 59, Issue 3, pp. 187 – 195.
- [DefMTurp] Miguel A. Padilla Castañeda, Fernando Arámbula Cosío, "Deformable model of the prostate for TURP surgery simulation". Computers and Graphics, 28, pp. 767-777.
- [DImgPro] Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing" , second edition.
- [DRother] Rother Diego, "Simulación de Imágenes Eléctricas en Peces Eléctricos de Descarga Débil". Tesis de Maestría, Departamento de Biología Celular y Molecular Instituto de Biología, Facultad de Ciencias, Abril 2003.
- [GeoTCG] Philip J. Schneider, David H. Eberly, "Geometric Tools for Computer Graphics". Morgan Kaufmann Publishers.
- [Herve] Hervé Delingette, "Toward Realistic Soft-Tissue Modeling in Medical Simulation", Proceedings of the IEEE, Vol. 86, No. 3, March 1998, pp. 512 - 523.
- [Kühnaphel] Kühnaphel U, Cakmak HK, MaaB H. "Endoscopic surgery training using virtual reality and deformable tissue simulation. "Computers and Graphics 2000, 24, pp.671–82.
- [Matlab] www.mathworks.com/access/helpdesk/help/techdoc/ref/delaunayn.shtml
- [MorInter] Marcin Iwanowski, Jean Serra "Morphological Interpolation and Color Images". Iciap, pp. 50, 10th International Conference on Image Analysis and Processing (ICIAP'99), 1999.

- [Morten] Morten Bro-Nielsen, "Finite Element Modeling in Surgery Simulation", Proceedings of the IEEE, Vol. 86, No. 3, March 1998, pp. 490 - 503.
- [MWCHull] <http://mathworld.wolfram.com/ConvexHull.html>
- [NIMDOS] Suriya Natsupakpong, "Numerical Integration Method in Deformable Object Simulation", Case Western Reserve, 2004.
- [Runge] http://en.wikipedia.org/wiki/Runge's_phenomenon
- [Serra] Jean Serra "Image Analysis and Mathematical Morphology" vol. 2. Academic Press, 1988.
- [SimQui] U. Meier, C. Monserrat, M. Alcañiz, MC. Juan, V. Grau, JA. Gil, "SIMULACIÓN QUIRÚRGICA", Medical Image Computing Laboratory (MedICLab), Universidad Politécnica de Valencia, Valencia, Spain.
- [TetMesh] Neil Molino, Robert Bridson, Ronald Fedkiw, "Tetrahedral Mesh Generation for Deformable Bodies". International Meshing Roundtable, 2003.
- [WSpline] [http://en.wikipedia.org/wiki/Spline_\(mathematics\)](http://en.wikipedia.org/wiki/Spline_(mathematics))
- [WRunge] <http://en.wikipedia.org/wiki/Image:Rungesphenomenon.png>