

UNIVERSIDAD SAN PABLO-CEU
ESCUELA POLITÉCNICA SUPERIOR
INGENIERIA SUPERIOR DE TELECOMUNICACIÓN



PROYECTO FINAL DE CARRERA
DISEÑO DE FILTROS ORIENTABLES PARA LA
DETECCIÓN DE ESTRUCTURAS LAMINARES Y
TUBULARES EN VOLÚMENES

Autor: Manuel Sánchez Pau

Director: Carlos Oscar Sánchez Sorzano

Febrero 2009

RESUMEN

La biotecnología es una de las ciencias que ofrece cada día mejoras en la calidad de las actividades humanas. Los avances en sectores como la medicina, la farmacia o la microbiología, están relacionados en gran medida con las nuevas soluciones tecnológicas que se obtienen en este campo. Una de las técnicas que mejoran enormemente los estudios que actualmente se realizan, es el diseño de filtros orientables. Con estos sistemas la detección de estructuras en fotografías o en volúmenes alcanza notables mejoras, no sólo en términos de calidad o en la mejor percepción de las mismas, sino en el tiempo de obtención de los resultados.

El estudio que se va a presentar en este documento es el desarrollo de un conjunto de filtros orientables que gracias a sus capacidades, podrán localizar estructuras tubulares o laminares en volúmenes biológicos. La implementación de los filtros partirá del análisis matemático de ciertas funciones que serán las más adecuadas a las configuraciones de los volúmenes que se están examinando. Posteriormente, se desarrollará un estudio completo acerca de la orientabilidad y las características de dichas funciones y del porqué de la elección de mismas. Los resultados finales del proyecto mostrarán una notable mejora en los estudios desarrollados hasta ahora, en parte, por la inexistencia de variedad en implementaciones de filtros orientables 3D para análisis de estructuras biológicas. Además de esto, se corroborará la validez del estudio mediante el examen de los tiempos de ejecución del sistema con respecto a desarrollos anteriores.

Como conclusión lo que pretende este proyecto es favorecer, en la medida de lo posible, futuras investigaciones biológicas, intentando aportar nueva información a los científicos que trabajan en este campo. De esta manera los filtros implementados serán incluidos en las librerías del Xmipp, programa de desarrollo de investigaciones del CNB, para el uso y progreso de nuevas aplicaciones.

ABSTRACT

Biotechnology offers opportunities for revolutionizing human welfare activities. Progress in medicine areas, as well as microbiology and pharmacy, are closely related to new technology development acquired on this division. One of the techniques that deeply improve the researches these days taking place is the steerable filter. Using this kind of methods is really useful in detecting structures in photographs or volumes, as it reaches significant improvements, not only in quality issues or its better perception, but in terms of obtaining results in a smaller range of time.

The research proposed in this document tries to develop a group of steerable filters that will find tubular or laminar structures in biologic volumes. Filters implementations will begin with a numerical analysis of certain features that will be the most suitable according to the volume configuration that is being studied. Subsequently, a complete study about the way to orientate them, their attributes and the reasons for selecting each of these functions will take place. Therefore, the final results will show an astonishing progress from the studies that have been taking place up till now. On one hand, because of the lack of variety in 3D steerable filters implementations for biological structure analysis. Furthermore, the strength of the research will be corroborated with a complete assessment on timing of the system execution comparing it to previous developments on the subject.

As a conclusion, this project tries to emphasize and increase the quality of future biological research, trying to contribute with innovating data for Scientifics that work in this area. In addition, filters implemented will be included in Xmipp's libraries, program used to improve investigations in CNB, for the use and progress of new applications.



AGRADECIMIENTOS

Este proyecto final de carrera se lo dedico a todas las personas que con su aliento y comprensión han hecho posible la finalización de mis estudios superiores. Agradezco el apoyo que me han dado mis familiares, amigos y profesores, sin los cuales el camino para conseguirlo hubiera sido mucho más difícil. También le agradezco a mi tutor, Carlos Oscar, las directrices y orientaciones que me ha dado, ya que me han servido de estímulo para conseguir mis objetivos. Finalmente me gustaría expresar un profundo agradecimiento hacia mis padres, por el esfuerzo y cariño que me han dado durante esta etapa y por ayudarme a alcanzar uno de los sueños de mi vida.

A todos vosotros

Muchas Gracias

INDICE

1. INTRODUCCIÓN	3
1.1. CONTEXTO Y ANTECEDENTES DEL PROYECTO	3
1.2. OBJETIVOS DEL PROYECTO	6
1.3. EL FILTRO ORIENTABLE	7
1.3.1. <i>Ejemplo: La derivada de un filtro Gaussiano</i>	10
1.3.1.1. <i>La derivada direccional de un filtro gaussiano es orientable</i>	11
1.4. ESTRUCTURA DEL PROYECTO	15
2. DESARROLLO DEL PROYECTO.....	18
2.1. DESARROLLO DE UN FILTRO ORIENTABLE GENÉRICO	18
2.1.1. <i>Demostración del teorema de Freeman</i>	19
2.1.2. <i>Ejemplo de función genérica</i>	20
2.1.3. <i>Adaptación del teorema de Freeman para funciones polinómicas con enventanado</i>	21
2.1.4. <i>Demostración de la aplicación del teorema de Freeman para funciones polinómicas con enventanado de simetría par o impar</i>	22
3. NUESTRO FILTRO ORIENTABLE DESARROLLADO	26
3.1. TIPOS DE FILTROS ORIENTABLES.....	26
3.2. LA ELECCIÓN DEL FILTRO ORIENTABLE	27
3.2.1. <i>El Filtro Orientable de Estructura Tubular</i>	28
3.2.2. <i>El Filtro Orientable de Estructura Laminar</i>	37
3.2.3. <i>Aproximación de la derivada direccional en funciones diferenciables</i>	41
3.3. CARACTERÍSTICAS DEL FILTRO ESCOGIDO: FUNCIONES SEPARABLES	42
4. APLICACIÓN DE LA CONDICIÓN DE ORIENTABILIDAD 3D DE FREEMAN AL FILTRO DISEÑADO.....	47
4.1. EL PROBLEMA.....	47
4.1.1. <i>Demostración del teorema de Freeman para tres dimensiones</i>	49
4.2. LA DEMOSTRACIÓN DEL ERROR	51
5. LA SOLUCIÓN DEL SISTEMA DE FILTRADO.....	53
6. EL MUESTREO ESFÉRICOS DE LOS FILTROS.....	55
7. RESULTADOS	60
7.1. ANÁLISIS DE LOS VOLÚMENES SIMULADOS.....	60
7.1.1. <i>Matlab</i>	65
7.1.1.1. <i>Filtro de estructura tubular</i>	65
7.1.1.2. <i>Filtro de estructura laminar</i>	68
7.1.1.3. <i>La orientación del volumen resultante</i>	70
7.1.2. <i>C++ con Xmipp</i>	73
7.1.2.1. <i>Filtro de estructura tubular</i>	73
7.1.2.2. <i>Filtro de estructura Laminar</i>	75
7.1.3. <i>Visualización con Chimera</i>	77
7.2. ANÁLISIS DE LOS VOLÚMENES REALES	78
7.2.1. <i>Análisis del filtrado tubular</i>	78



PROYECTO FINAL DE CARRERA

7.2.2.	<i>Análisis del filtrado laminar.....</i>	<i>83</i>
7.2.3.	<i>Visualización de los resultados en Chimera.....</i>	<i>89</i>
7.2.4.	<i>Análisis del filtrado múltiple para el volumen laminar real.....</i>	<i>89</i>
8.	ESTUDIO DE TIEMPOS DE EJECUCIÓN	91
8.1.	COMPARATIVA DE TIEMPOS ENTRE EL USO O NO DEL FILTRO ORIENTABLE.....	91
8.1.1.	<i>Implementación de filtro separable</i>	<i>91</i>
8.1.2.	<i>Implementación de las bases para el uso del filtro orientable</i>	<i>92</i>
8.2.	COMPARATIVAS DE TIEMPOS DE EJECUCIÓN CON DIFERENTES VALORES DE MUESTREO ORIENTATIVO	93
8.3.	COMPARATIVA DE TIEMPOS ENTRE LOS DIFERENTES TAMAÑOS DE LOS VOLÚMENES	96
9.	TRABAJOS FUTUROS.....	97
10.	CONCLUSIONES	98
11.	BIBLIOGRAFÍA.....	99
12.	ANEXOS (CÓDIGOS DE MATLAB Y XMIPP)	100
12.1.	CÓDIGO C++ CON LIBRERÍA XMIPP	100
12.2.	CÓDIGO EN MATLAB	106

1. Introducción

1.1. Contexto y Antecedentes del Proyecto

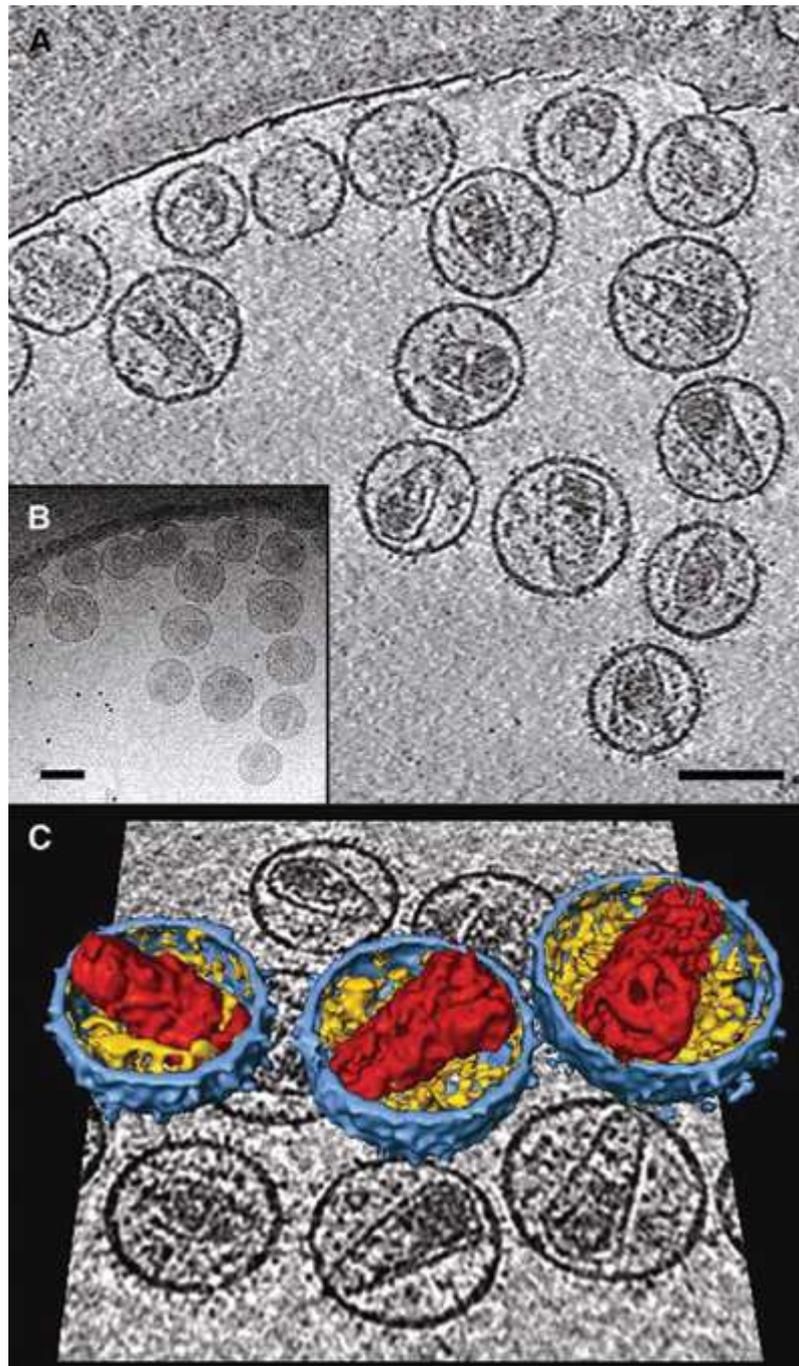
Hoy en día las técnicas de procesado de señal en imágenes están cobrando una gran importancia. Los algoritmos matemáticos y las aplicaciones técnicas han evolucionado tanto, que estos sistemas se están implantando en sectores que no se cabía esperar. Del mismo modo, los proyectos desarrollados están tomando una mayor importancia académica y social lo que han llevado a la creación, por parte de algunas organizaciones, de nuevas subvenciones y becas.

El estudio en este documento se realiza en el campo de la biología, en concreto, en la parte de microscopía electrónica. Esto es así porque los avances que se pueden obtener en este sector son muy interesantes y pueden lograr grandes resultados para la investigación.

Como es ampliamente conocido la microscopía electrónica ha supuesto un salto importante para las investigaciones biológicas desde su invención. Ahora, de la misma forma, el análisis de las tomografías utilizando el sistema de procesado de imágenes supone una mejora en la calidad de los estudios. Un mejor resultado amplía el entendimiento de estructuras, formas y esqueletos de diferentes especímenes biológicos. Esto es primordial para un investigador porque si comprende la configuración de los ejemplares que está analizando, puede extrapolar a otra escala los resultados obtenidos y así, aplicar estos conocimientos a otros estudios como, por ejemplo, los bioquímicos.

Aunque este estudio se centra en un campo en concreto hay que decir que el análisis de señales está abarcando numerosos sectores, como la medicina, la estadística en bolsa, el análisis facial para aplicaciones comerciales, manipulación fotográfica etc.

Un ejemplo de las investigaciones que se están llevando a cabo en el ámbito de la biología, son las que se presentan en el documento “*Comparison of single-particle analysis and electron tomography approaches*” [16]. Las imágenes de los volúmenes que se muestran a continuación presentan reconstrucciones de volúmenes biológicos de dicho proyecto.



Como muestran los resultados de esta ilustración, el entendimiento de las estructuras por parte de los científicos es mayor que en el pasado. Gracias a estos análisis se puede obtener un aumento en el conocimiento acerca de ciertos virus o bacterias, suponiendo así, una posible mejora en las soluciones para ciertas enfermedades.

PROYECTO FINAL DE CARRERA

Estos estudios se basan en la modificación de las imágenes mediante filtros que poseen ciertas propiedades. Dependiendo de las necesidades de las tomografías a analizar, se utilizan unos u otros. La detección de los bordes, la modificación del aspecto o la eliminación de ruido de la imagen son ejemplos de dichas funcionalidades.

Los métodos utilizados para cada uno de los procesos de filtrado son diferentes, aunque en todos ellos, radica la idea principal de obtener nueva información inherente de la imagen, que no puede ser obtenida mediante los sistemas tradicionales. Un ejemplo de ello son los citados métodos de *denoising* o de eliminación de ruido, que son capaces de filtrar las imágenes que presentan poca nitidez para el ser humano. De esta manera, aplicando un filtro, podemos obtener importantes mejoras en la detección de la información de las imágenes.

Además de todo esto, en la actualidad existe un nuevo aspecto que está tomando gran importancia en los sistemas de filtrado. Éste es el análisis de imágenes en tres dimensiones, que es lo que se conoce como el procesado de señal en volúmenes. La información visual que se recibe mediante dichos sistemas es mucho mayor, dado que se aproximan más a la naturaleza de las estructuras o sistemas analizados, lo que supone un valor añadido al entendimiento de los datos.

En este proyecto se van a estudiar imágenes y volúmenes biológicos que presentan estructuras tubulares y laminares, con objeto de detectar su presencia mediante técnicas de análisis automatizadas. Algunas de las operaciones citadas han sido recopiladas de numerosas publicaciones y tesis de estos últimos años, con objeto de analizar en profundidad las mejores soluciones.

En principio, la idea fundamental de este estudio reside en las técnicas de filtrado conocidas como *Steerable filter*, o en castellano “filtros orientables”, que serán desarrolladas a lo largo del documento con gran detalle.

1.2. Objetivos del Proyecto

El objetivo fundamental de este proyecto es desarrollar dos filtros orientables en tres dimensiones, que sean capaces de analizar con eficacia una serie de volúmenes biológicos. El primero deberá detectar estructuras tubulares como filamentos. Mientras que el segundo está orientado a realizar la misma operación pero con configuraciones laminares.

La decisión de implementar estos sistemas como filtros orientables reside en la capacidad que tienen dichas metodologías para la obtención de información de un volumen en tiempos mucho más limitados que otros sistemas. Sin embargo, la elaboración de dichos filtros es notablemente más compleja que mediante las técnicas tradicionales.

Muchos de los ejemplos que se mostrarán a lo largo del documento como verificación de la validez de los filtros 3D, serán demostraciones matemáticas desarrolladas en dos dimensiones. Esto es así para simplificar el entendimiento del proyecto, dado que las funciones que se muestran en la mayoría de los casos son bastante complejas. No existe ningún problema en realizar el estudio de esta forma dado que se ha comprobado que todas esas funciones son extrapolables a tres dimensiones.

Como punto final se pretende hacer un análisis exhaustivo de los resultados obtenidos mediante los sistemas de filtrado a nivel de calidad y como valor añadido se analizarán también las mejoras cuantitativas del tiempo empleado para la ejecución de los mismos.

De este modo, el estudio que se presenta pretende ser un avance en el análisis de volúmenes para la aplicación en labores de investigación biotecnológica o en cualquier sector al que se les pueda aplicar este estudio.

1.3. El Filtro orientable

El conocimiento para abordar esta investigación reside en los anteriormente citados filtros orientables o *Steerable Filters*. Dicha metodología ha sido estudiada desde hace tiempo, sin embargo, a la hora de implementar estos sistemas para darles un uso práctico, se han centrado la mayoría de los esfuerzos en imágenes y no en volúmenes. Una posible hipótesis para este suceso, puede ser la dificultad de manipulación de los volúmenes mediante ordenadores, no sólo por el tiempo a la hora de calcular el filtrado, sino a la hora de realizar los análisis visuales de los resultados. Sin embargo, actualmente los avances en el terreno de la tecnología computacional están siendo enormes, lo que hacen que el filtrado en tres dimensiones mediante técnicas automatizadas pueda ser una realidad bastante factible.

Según las investigaciones preliminares para la búsqueda de información acerca de los filtros orientables, el primer artículo que muestra con gran exactitud el desarrollo de los mismos es el de Willian T. Freeman y Edward H. Adelson “*The Design and Use of Steerable Filters*” que fue publicado en 1991 [1]. Dicho documento es el pilar sobre el que se basan la mayoría de los estudios actuales sobre este campo. Por eso mismo los filtros diseñados en este proyecto partirán del conocimiento de este.

En general, se suele usar el término filtro orientable para describir una clase de filtros que tienen la capacidad de analizar una imagen, 2D o 3D, en cualquier dirección arbitraria de la misma. Las funciones de dichos filtros deben ser capaces de obtener la máxima energía de la imagen y su orientación en el espacio. Esta capacidad reside fundamentalmente en la estructura que presentan las funciones de los filtros. Por ejemplo, si se quiere detectar un filamento en un volumen biológico hace falta un filtro de estructura tubular, si se utilizase otra configuración posiblemente los resultados no serían correctos. Los filtros orientables se obtienen mediante la combinación lineal de un conjunto de bases que, en principio, son versiones rotadas de los mismos. El número de bases a utilizar depende de las características del filtro.

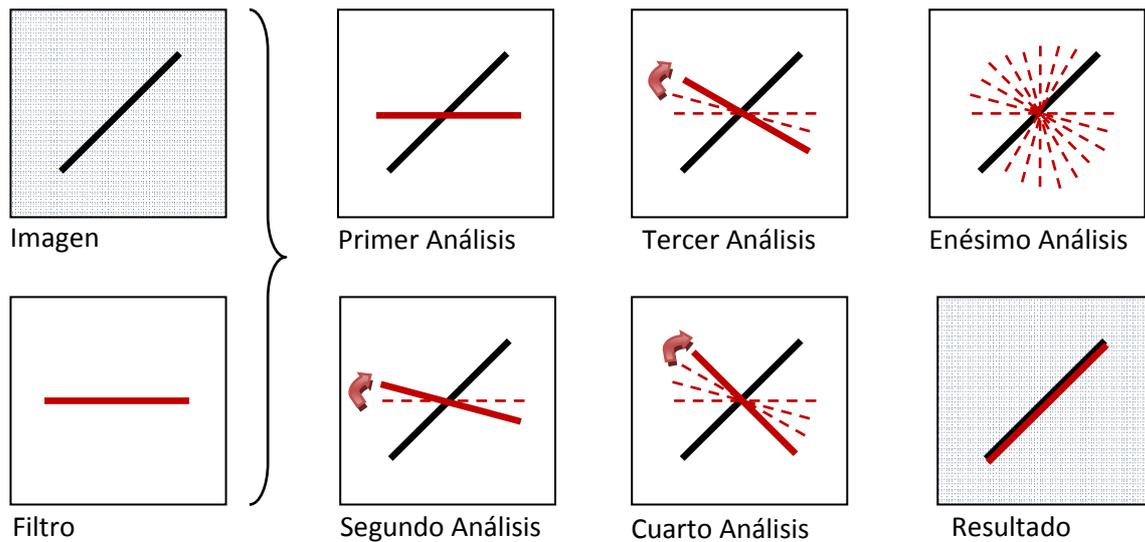
Para la detección en imágenes o en volúmenes, se han investigado cuáles son los mejores métodos para conseguir los resultados óptimos y los criterios importantes a

PROYECTO FINAL DE CARRERA

tener en cuenta. No obstante, para cada tipo de estructura se presentan diferentes configuraciones de filtrado.

Para concretar la definición que se ha hecho de los filtros orientables se va a explicar el porqué de su invención mediante el desarrollo de la idea más básica que se le puede ocurrir a cualquier investigador, que no conozca estos avances, y quiera detectar una estructura en una imagen. En principio lo lógico sería aplicar a la imagen de referencia un filtro adecuado, que rotado en diferentes direcciones, obtenga las orientaciones y la energía de la misma.

A continuación se muestra una ilustración sencilla de dicho método para un único filamento, filtrado por un sistema de estructura tubular.



Si se tiene una imagen como la que se presenta en la figura y se realiza el filtrado, hacen falta numerosas operaciones en diferentes direcciones para conseguir el resultado apropiado. El coste de tiempo de cada una de las operaciones es muy grande, lo que supone gran dificultad a la hora de aplicar estos métodos a aplicaciones reales.

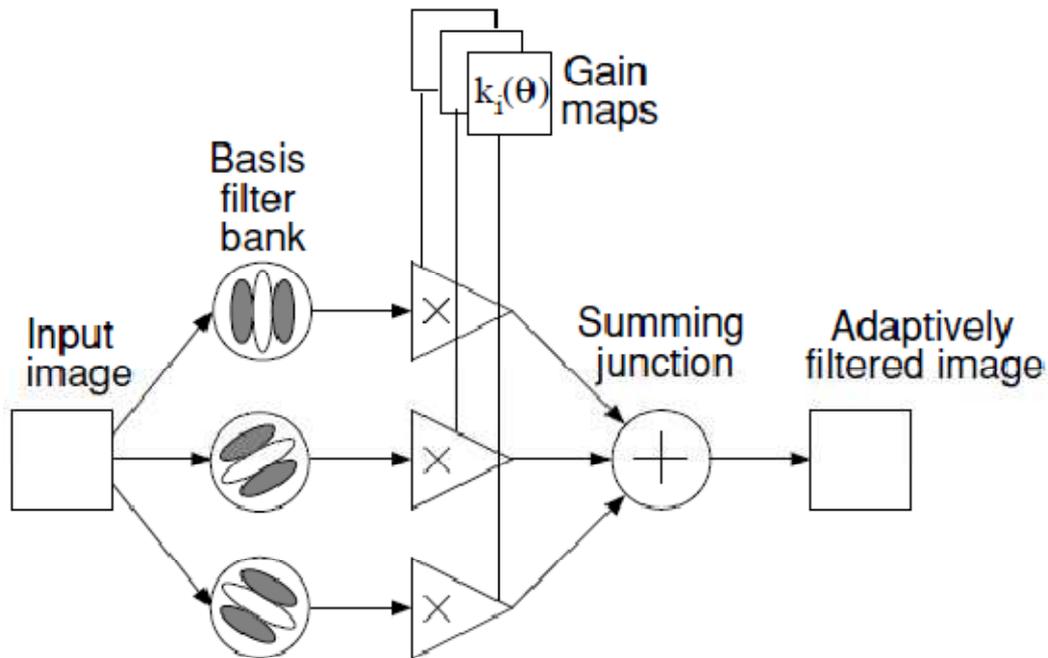
Como puede observarse, para que dicho método sea óptimo es necesario que los ángulos de rotación del filtro sean muy pequeños, es decir, hacer muchas operaciones, porque de otro modo no detectaría la estructura de la imagen.

PROYECTO FINAL DE CARRERA

Como se ha demostrado la traba de este procedimiento es el coste de tiempo de procesamiento debido a la gran cantidad de operaciones que hay que realizar y el tamaño de las imágenes. Por este motivo la pregunta típica que se realiza cualquier experto en este campo es, cuál es el método que optimiza la relación entre la funcionalidad del sistema y su grado de complejidad.

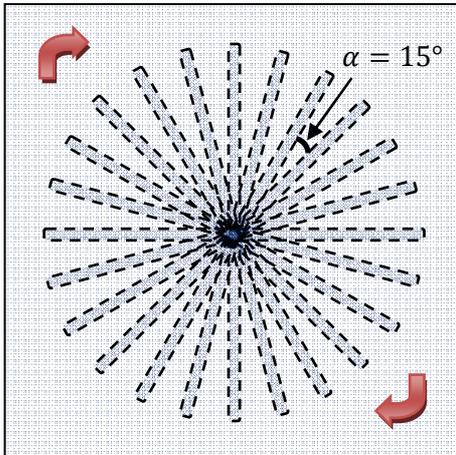
Con esto, aparece el diseño y uso de filtros orientables, que han sido una revolución debido a las mejoras en el tiempo que pueden lograrse.

A continuación se muestra un esquema de filtro orientable para establecer una comparación con el sistema anterior.

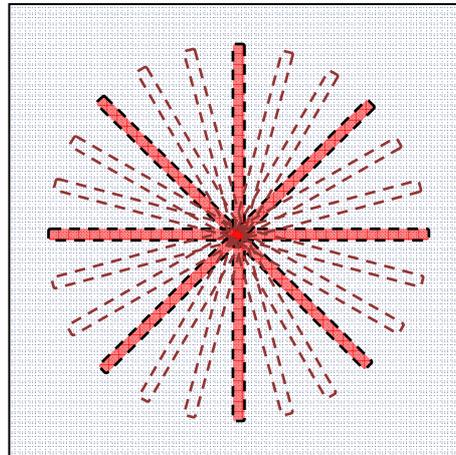


Como se ilustra en este esquema, el filtrado no se realiza indiscriminadamente sino que se obtienen un conjunto de bases, que son versiones rotadas del mismo, y mediante un conjunto de coeficientes que actúan como pesos relativos de dichas bases, se logra una resultado mejor al que se podría obtener con el sistema anterior.

Las ilustraciones que se muestran a continuación muestran un esquema comparativo entre las dos soluciones.



Primera Opción



Steerable Filter

En estas ilustraciones cada una estructuras representa una orientación de filtrado. En la primera opción, el filtro realiza doce operaciones de filtrado mientras que en la segunda, en el caso del *Steerable filter*, simplemente hacen falta realizar tres. Estas serán las bases del filtro y se utilizarán para obtener los resultados de las restantes operaciones mediante la combinación lineal de las mismas.

En los siguientes apartados se irá introduciendo poco a poco más conceptos e ideas del funcionamiento de esta clase de filtros. A continuación se va a desarrollar un sistema sencillo de filtro orientable.

1.3.1. Ejemplo: La derivada de un filtro Gaussiano

Para simplificar el entendimiento del sistema de filtrado y saber a ciencia cierta qué es un filtro orientable en forma matemática, se va a mostrar un ejemplo en dos dimensiones. El sistema más sencillo es una función gaussiana con simetría circular en coordenadas cartesianas (x,y).

$$G(x, y) = e^{-(x^2+y^2)}$$

El escalado y las constantes de normalización de esta función se han puesto a uno por conveniencia.

Antes de empezar el ejemplo es importante advertir que se van a utilizar varias operaciones matemáticas que deben ser estudiadas para asegurar que cumplen todas las condiciones de orientabilidad. Esto es crucial para no comprometer el resultado final del filtro por la incertidumbre de si las operaciones realizadas nos cumplen las premisas iniciales.

De este modo, a continuación se muestra la orientabilidad del operador *derivada direccional* aplicado a un filtro gaussiano.

1.3.1.1. La derivada direccional de un filtro gaussiano es orientable

Como se ha comentado la implementación orientable más simple, es la primera derivada direccional de una función gaussiana de dos dimensiones. Aunque en este apartado sólo se consideran los sistemas de este estilo, el principio de orientabilidad que se muestra, puede ser extendido a cualquier derivada de orden n de una función gaussiana.

Si consideramos la siguiente función $g(x, y)$ y realizamos su derivada direccional en la dirección de x , el resultado obtenido sería la función $g_x(x, y)$, que se muestra en la siguiente ilustración.

$$g(x, y) = e^{-(x^2+y^2)/2}$$

$$g_x(x, y) = \frac{\partial g(x, y)}{\partial x}$$

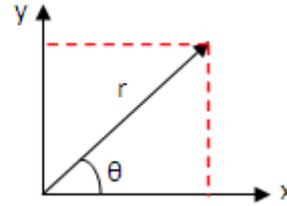
$$g_x(x, y) = -xe^{-(x^2+y^2)/2}$$

El objetivo a partir de ahora es demostrar que la función anterior es orientable. Para que esto sea verdad es necesario que se cumpla que la función pueda ser sintetizada en cualquier dirección como combinación lineal de unas bases, que serán la misma función orientada en diferentes direcciones del espacio. Dado que la exposición de este ejemplo es más sencilla en coordenadas polares, el desarrollo se hará en las mismas.

Paso a coordenadas polares:

$$r = \sqrt{x^2 + y^2} \quad r^2 = x^2 + y^2$$

$$x = r \cos \theta$$



Con lo que si se sustituye en $g_x(x, y)$ el resultado sería:

$$g_x(r, \theta) = -re^{-r^2/2} \cos(\theta)$$

De esta manera la función $g_x(r, \theta)$ se puede separar en dos partes $-re^{-r^2/2}$ y $\cos(\theta)$. Como puede observarse la segunda parte es la que depende de la orientación del ángulo. Dado que estamos interesados en la rotación de la función, vamos a considerar una copia de la misma rotada en $\pi/2$. Sustituyendo el coseno anterior por $\cos(\theta - \pi/2)$ se obtiene lo siguiente:

$$g_x(r, \theta - \pi/2) = -re^{-r^2/2} \cos(\theta - \pi/2)$$

Ahora se utiliza la siguiente relación trigonométrica para desarrollar la fórmula.

$$\cos(A - B) = \cos(A) \cos(B) + \sin(A) \sin(B)$$

De este modo la ecuación resultante es:

$$g_x(r, \theta - \pi/2) = -re^{-r^2/2} (\cos(\theta) \cos(\pi/2) + \sin(\theta) \sin(\pi/2))$$

$$= -re^{-r^2/2} \sin(\theta)$$

Si ahora se realizan las mismas operaciones pero con la derivada en función en la dirección de la y, nos queda:

$$g_y(x, y) = \frac{\partial g(x, y)}{\partial y}$$

$$g_y(x, y) = -ye^{-(x^2+y^2)/2}$$

$$g_y(r, \theta) = -re^{-r^2/2} \sin(\theta)$$

Este resultado es similar al obtenido anteriormente con la rotación de la derivada direccional en función de x . Este hecho es sólo una de las condiciones que tiene que cumplir una función para ser considerada orientable. La otra condición parte de la siguiente idea.

$$g_{\alpha}(r, \theta) = -re^{-r^2/2} \cos(\theta - \alpha)$$

$$g_{\alpha}(r, \theta) = -re^{-r^2/2}(\cos(\theta)\cos(\alpha) + \sin(\theta)\sin(\alpha))$$

$$g_{\alpha}(r, \theta) = \cos(\alpha)(-re^{-r^2/2} \cos(\theta)) + \sin(\alpha)(-re^{-r^2/2} \sin(\theta))$$

$$g_{\alpha}(r, \theta) = \cos(\alpha)g_x + \sin(\alpha)g_y$$

$$g_{\alpha}(r, \theta) = (\cos(\alpha) \sin(\alpha)) \begin{pmatrix} g_x \\ g_y \end{pmatrix}$$

$$g_{\alpha}(r, \theta) = \mathbf{v}(\alpha) \cdot \mathbf{b}$$

En esta última función el vector $\mathbf{v}(\alpha)$ contiene los coeficientes de interpolación de la base \mathbf{b} .

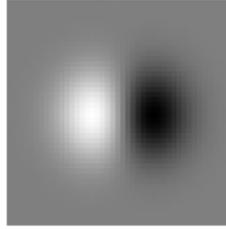
La orientabilidad de la derivada direccional es independiente de la selección que se hagan de las bases. Esto quiere decir que seleccionando arbitrariamente las orientaciones de las bases deben existir unos factores de interpolación que permitan recuperar la derivada en cualquier dirección.

Para una demostración más avanzada acerca de la orientabilidad de cualquier derivada de la función gaussiana de cualquier orden, ver documento [9] de la lista de la bibliografía.

Continuando con el ejemplo de la sección 1.3.1, donde la función a analizar es $G(x, y) = e^{-(x^2+y^2)}$ su derivada direccional en dirección de x es la siguiente:

$$G_1^{00} = \frac{\partial}{\partial x} e^{-(x^2+y^2)} = -2xe^{-(x^2+y^2)}$$

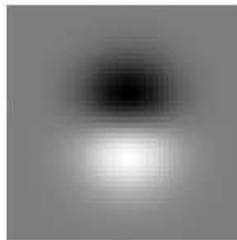
Cuya representación es:



La misma función rotada 90 grados, es:

$$G_1^{90^0} = \frac{\partial}{\partial y} e^{-(x^2+y^2)} = -2ye^{-(x^2+y^2)}$$

Cuya representación es:



Con estas dos últimas funciones podemos diseñar un filtro que sea arbitrario en cualquier dirección mediante la combinación lineal de ambas. Los términos $\cos(\theta)$ y $\sin(\theta)$ son las funciones de interpolación de las bases $G_1^{0^0}$, $G_1^{90^0}$.

$$G_1^\theta = \cos(\theta)G_1^{0^0} + \sin(\theta)G_1^{90^0}$$

Además, como la convolución es una operación lineal, se puede realizar la misma operación pero convolucionando antes las bases con una imagen.

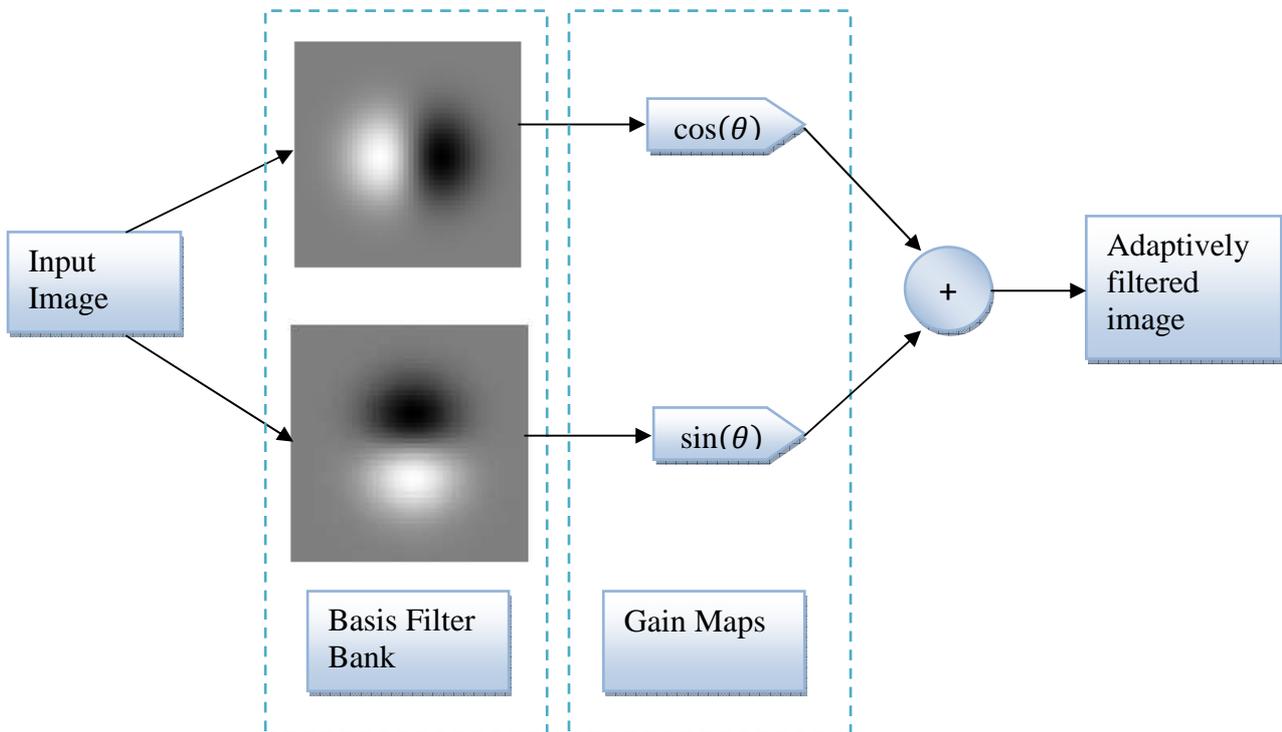
$$R_1^{0^0} = G_1^{0^0} * I$$

$$R_1^{90^0} = G_1^{90^0} * I, \text{ entonces}$$

$$R_1^\theta = \cos(\theta)R_1^{0^0} + \sin(\theta)R_1^{90^0}$$

Lo anterior permite ilustrar de forma matemática lo que es un filtro orientable.

En resumen, un filtro orientable es un filtro que puede ser sintetizado a través de unas bases y unos coeficientes de interpolación y que tiene la propiedad de tener una orientación arbitraria. Para obtener el valor del filtro resultante solo hay que encontrar el valor de las funciones de interpolación mediante su ángulo. A continuación se muestra el esquema típico utilizado en los filtros orientables aplicado a la estructura de este ejemplo:



1.4. Estructura del Proyecto

Para entender cuál ha sido el proceder en este proyecto a continuación se muestra la estructura paso por paso de este documento.

El primer punto después de la introducción será el desarrollo del proyecto (sección 2). En esta parte se realizará en detalle un análisis acerca de cuál es el filtro que puede adaptarse mejor a las necesidades de los volúmenes que tenemos. Esto es así debido a que, matemáticamente, se pueden desarrollar gran cantidad de funciones que pueden ser orientables. En este caso, los parámetros que hay que tener en cuenta para la

PROYECTO FINAL DE CARRERA

elección del filtro serán: el tiempo de ejecución, la capacidad de adaptación del filtro a los volúmenes, la calidad resultante del filtrado, el número de bases utilizadas y la versatilidad a la hora de manipular diferentes tipos de volúmenes. Para ello se desarrollará una función genérica (sección 2.1) que exprese cuales son los posibles sistemas son adaptables a la orientabilidad y a las estructuras de las imágenes a analizar.

En el siguiente apartado se estudiarán varios filtros ya desarrollados, con la idea de conocer sus características para adquirir sus ventajas y eliminar sus limitaciones. A partir de este conocimiento, en la sección 3.2, se diseñarán dos filtros, uno para la detección de estructuras tubulares y el otro para la detección de laminares. Ambos filtros se han implementado en dos lenguajes de programación. El primero de ellos es MATLAB, un lenguaje muy propio para desarrollos matemáticos. El segundo es C++ utilizando la librería XMIPP, que es una forma muy eficaz a la hora de manipular volúmenes y que además tiene numerosas herramientas para realizar funciones de filtrado. De este modo a lo largo de todo el documento se mostrarán ilustraciones de dichas implementaciones para validar las explicaciones realizadas.

Posteriormente se explicarán dos secciones (4 y 5) acerca de la orientabilidad de los filtros según las condiciones de Freeman en tres dimensiones.

En la sección 7 se explicarán finalmente los resultados de los análisis donde se mostrarán dos tipos de sistemas examinados. Para la visualización de los resultados se utilizarán dos programas. Uno de ellos es un *toolbox* de MATLAB, llamado *volume_browser*, en donde se podrán ver los resultados mediante una escala de colores de las energías de los volúmenes. El segundo programa, llamado CHIMERA, es un programa muy potente que mostrará los resultados mediante un sistema de visualización basado en isosuperficies. También se realizarán visualizaciones de los cortes de los volúmenes finales mediante una función de la librería Xmipp llamada *xmipp_show*.

Posteriormente (en la sección 8) se realizará una comparativa de tiempos de ejecución entre las formas tradicionales de filtrado de filamentos y el nuevo sistema de filtrado. Esta etapa es importante para ver la calidad y cantidad de mejora que se ha alcanzado con el filtro.



PROYECTO FINAL DE CARRERA

Como último punto se realizará una conclusión y una posible visión acerca de la aplicación del filtro creado a la vida real. Lo que es importante dado que las investigaciones deben tener un propósito a la hora de desarrollarse.

2. Desarrollo del Proyecto

2.1. Desarrollo de un Filtro Orientable Genérico

En este apartado se va a realizar un estudio acerca de la orientabilidad de los filtros de cualquier tipo. Esto es muy importante dado que tenemos que saber, a ciencia cierta, si la función que vamos a escoger es orientable en todas las condiciones posibles.

En principio, todas las funciones que están limitadas en frecuencia pueden ser filtros orientables. El problema es que hay algunas que son más útiles en la práctica por su escasa necesidad de bases.

Es importante saber cuáles son las condiciones que deben tener las funciones para ser consideradas filtros orientables. Para ello partimos de una función genérica como la siguiente:

$$f^\theta(x, y) = \sum_{j=1}^M k_j(\theta) f^{\theta_j}(x, y)$$

En esta función $f^\theta(x, y)$ distinguimos un sumatorio de dos partes $k_j(\theta)$ que es lo que se describía con anterioridad como los coeficientes de interpolación y $f^{\theta_j}(x, y)$ que es la función rotada en un ángulo θ_j . Ahora hay que descubrir qué funciones $f(x, y)$ pueden satisfacer la función anterior, cuál es el término M (que es el número de bases requeridas) y cuáles son los coeficientes de interpolación.

El desarrollo de la función se realiza en coordenadas polares $r = \sqrt{x^2 + y^2}$ y $\theta = \arg(x, y)$.

$$f^\theta(r, \theta) = \sum_{j=1}^M k_j(\theta) f^{\theta_j}(r, \theta)$$

Siendo f una función cualquiera, puede ser expandida en series de Fourier como:

$$f(r, \theta) = \sum_{n=-N}^N a_n(r) e^{in\theta}$$

De este modo según el teorema de Freeman expresado en el documento [1], la condición de orientabilidad en una función genérica $f^\theta(x, y) = \sum_{j=1}^M k_j(\theta) f^{\theta_j}(x, y)$ expandida en series de Fourier como la expresión anterior, se mantiene, si y sólo si, los coeficientes de interpolación $k_j(\theta)$ son solución de la siguiente matriz:

$$\begin{pmatrix} 1 \\ e^{i\theta} \\ \vdots \\ e^{iN\theta} \end{pmatrix} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ e^{i\theta_1} & e^{i\theta_2} & \dots & e^{i\theta_M} \\ \vdots & \vdots & \vdots & \vdots \\ e^{iN\theta_1} & e^{iN\theta_2} & \dots & e^{iN\theta_M} \end{pmatrix} \begin{pmatrix} k_1(\theta) \\ k_2(\theta) \\ \vdots \\ k_M(\theta) \end{pmatrix}$$

Donde θ es el ángulo de la función y N es su orden. Si para algún n , $a_n(r) = 0$, entonces las correspondiente columna (n^{th}) debería ser eliminada. El interés reside ahora en saber cuál es el número mínimo de bases (M) que se requieren para hacer una determinada función $f(r, \varphi)$ orientable. Antes de mostrar un ejemplo de ello, vamos a demostrar que el anterior teorema es cierto.

2.1.1. Demostración del teorema de Freeman

Sustituyendo el desarrollo en serie de Fourier $f(r, \theta) = \sum_{n=-N}^N a_n(r) e^{in\theta}$ en ambos lados de la condición de orientabilidad, escrita en polares ($f^\theta(r, \theta) = \sum_{j=1}^M k_j(\theta) f^{\theta_j}(r, \theta)$), la ecuación resultante sería:

$$\sum_{n=-N}^N a_n(r) e^{in\theta} = \sum_{j=1}^M k_j(\theta) \sum_{n=-N}^N a_n(r) e^{in\theta_j}$$

Si ahora los sumatorios los proyectamos como un conjunto de ecuaciones, las cuales son escritas en la forma de $e^{-in\varphi}$ para $-N \leq n \leq N$, se obtiene un conjunto simultáneo de ecuaciones lineales de la siguiente manera:

$$a_n(r) e^{in\theta} = \sum_{j=1}^M k_j(\theta) a_n(r) e^{in\theta_j} \quad -N \leq m \leq N$$

Dividiendo a ambos lados de la ecuación por $a_n(r)$ obtenemos:

$$e^{in\theta} = \sum_{j=1}^M k_j(\theta) e^{in\theta_j} \quad -N \leq m \leq N$$

Es decir, existiría un conjunto desde $-N$ hasta N ecuaciones. También se puede simplificar mucho si sólo consideramos las frecuencias positivas $0 \leq n \leq N$, debido a que esto no supone ninguna pérdida en la generalización de la demostración.

De este modo el conjunto resultante sería el siguiente:

$$e^{in\theta} = \sum_{j=1}^M k_j(\theta) e^{in\theta_j} \quad 0 \leq n \leq N$$

Por simplicidad, el resultado es conveniente expresarlo en forma matricial.

$$\begin{pmatrix} 1 \\ e^{i\theta} \\ \vdots \\ e^{iN\theta} \end{pmatrix} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ e^{i\theta_1} & e^{i\theta_2} & \dots & e^{i\theta_M} \\ \vdots & \vdots & \vdots & \vdots \\ e^{iN\theta_1} & e^{iN\theta_2} & \dots & e^{iN\theta_M} \end{pmatrix} \begin{pmatrix} k_1(\theta) \\ k_2(\theta) \\ \vdots \\ k_M(\theta) \end{pmatrix}$$

De esta manera queda demostrado que los coeficientes de interpolación, según el teorema de Freeman, se obtienen con la matriz resultante.

2.1.2. Ejemplo de función genérica

Partiendo de esto, se va a realizar un ejemplo para mostrar cómo se realizaría la orientabilidad de una función buscando sus valores de interpolación y el número de bases necesarios.

Si se tiene, por ejemplo, la siguiente función el desarrollo sería:

$$G_1(r, \varphi) = -2re^{-r^2} \cos(\varphi) = -re^{-r^2} (e^{i\varphi} + e^{-i\varphi})$$

La función $G_1(r, \varphi)$ tiene dos coeficientes $a_n(r)$ no nulos en la descomposición angular de Fourier. Según el teorema anterior, dos bases son suficientes para sintetizar G_1^φ . La interpolación de las funciones según la matriz del teorema quedaría de la siguiente manera:

$$e^{i\theta} = \begin{pmatrix} e^{i\theta_1} & e^{i\theta_2} \end{pmatrix} \begin{pmatrix} k_1(\theta) \\ k_2(\theta) \end{pmatrix}$$

Si cogiésemos ahora los ángulos $\theta_1 = 0^0$ y $\theta_2 = 90^0$ para sintetizar las bases del filtro orientable, nos quedarían las funciones de interpolación $k_1(\theta) = \cos(\theta)$ y $k_2(\theta) = \sin(\theta)$ (desarrollando la formula matricial anterior). Y el resultado del filtro orientable sería:

$$G_1^\theta = \sum_{j=1}^2 k_j(\theta) G_1^{\theta_j} = \cos(\theta) G_1^{0^0} + \sin(\theta) G_1^{90^0}$$

Como hemos demostrado anteriormente para el caso del filtro de derivada de la gaussiana.

2.1.3. Adaptación del teorema de Freeman para funciones polinómicas con enventanado

En esta sección se quiere avanzar un poco más en el teorema de Freeman. El objetivo ahora es aproximar el teorema anterior a funciones polinómicas con enventanado. Esto es importante dado que con dichas funciones se pueden obtener un número limitado de bases, además de conseguir que la función tenga propiedades separables en $[x - y - z]$ (esta propiedad será expuesta en detalle a lo largo del documento y más en concreto en la sección 3.3).

Las funciones polinómicas con enventanado se presentan mediante la siguiente estructura:

$$f(x, y) = W(r) P_N(x, y),$$

Donde $W(r)$ es una función de enventanado y $P_N(x, y)$ es un polinomio de orden N en coordenadas (x, y) , cuyos coeficientes deben depender de r . La combinación lineal de $2N+1$ bases son suficientes para sintetizar la función rotada en cualquier ángulo.

Como se mostró anteriormente, la matriz que da los coeficientes de interpolación de las bases debe aplicarse de igual modo en estas funciones.

$$\begin{pmatrix} 1 \\ e^{i\theta} \\ \vdots \\ e^{iN\theta} \end{pmatrix} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ e^{i\theta_1} & e^{i\theta_2} & \dots & e^{i\theta_M} \\ \vdots & \vdots & \vdots & \vdots \\ e^{iN\theta_1} & e^{iN\theta_2} & \dots & e^{iN\theta_M} \end{pmatrix} \begin{pmatrix} k_1(\varphi) \\ k_2(\varphi) \\ \vdots \\ k_M(\varphi) \end{pmatrix}$$

Por otra parte este teorema especifica de forma genérica que $2N+1$ bases son suficientes para conseguir una función arbitraria para funciones polinómicas. Lo que hay que aclarar es que para ciertas funciones existen un menor número de bases que lo cumplen. Por ejemplo, si $P_N(x, y)$ contiene sólo términos de orden par o impar, entonces solamente son necesarias $N+1$ bases, y la matriz anterior puede ser modificada para contener sólo los términos pares o impares.

2.1.4. Demostración de la aplicación del teorema de Freeman para funciones polinómicas con enventanado de simetría par o impar

En este apartado se va a demostrar por qué solamente son necesarias $N+1$ bases para un polinomio con enventanado de simetría par o impar. Este punto es importante debido a que, como se mostrará en la sección 3.2, la función elegida para nuestro filtro está basada en esta estructura.

Si se consideran los términos de una función polinómica de orden n , expresada en coordenadas cartesianas como $x^k y^{n-k}$, donde $0 \leq k \leq n$, y se reescribe dicho término en coordenadas polares usando $x = r \cos(\varphi)$ y $y = r \sin(\varphi)$, obtenemos la siguiente expresión:

$$x^k y^{n-k} = r^n \cos(\varphi)^k \sin(\varphi)^{n-k}$$

Esta función que se muestra como productos de senos y cosenos elevados a determinadas potencias, puede ser desarrollada como series de Fourier que sólo contengan frecuencias de la forma $n\varphi, (n-2)\varphi, \dots, -(n-2)\varphi, -n\varphi$. De este modo, un polinomio de orden N que contiene sólo términos pares debería tener únicamente frecuencias angulares de $-N \leq m \leq N$. Como hemos dicho antes si el polinomio presenta todas las frecuencias angulares en valor absoluto menores o iguales a N se necesitarían $2N + 1$ bases para desarrollar el filtro. En el caso de simetría par se

PROYECTO FINAL DE CARRERA

necesitan $N + 1$ bases. La demostración de este desarrollo en Fourier es compleja por lo que se va a demostrar dicha afirmación mediante relaciones trigonométricas.

Para analizar esta estructura es necesario conocer ciertas igualdades que se muestran a continuación:

$$\sin(\varphi)^{2n} A = \frac{1}{2^{2N}} \binom{2n}{n} + \frac{1}{2^{2N-1}} \sum_{i=0}^{n-1} \binom{2n}{i} (-1)^i \cos[(2n - 2i)A]$$

$$\cos(\varphi)^{2n-1} A = \frac{1}{2^{N-2}} \sum_{i=0}^{n-1} \binom{2n-1}{i} \cos[((2n-1) - 2i)A]$$

$$\cos(\varphi)^{2n} A = \frac{1}{2^{2N}} \binom{2n}{n} + \frac{1}{2^{2N-1}} \sum_{i=0}^{n-1} \cos[(2n - 2i)A]$$

$$\cos(A)\cos(B) = \frac{1}{2} [\cos(A - B) + \cos(A + B)]$$

$$\cos(A)\sin(B) = \frac{1}{2} [\sin(B - A) + \sin(B + A)]$$

Ahora si analizamos el coseno y el seno de la expresión $x^k y^{n-k} = r^n \cos(\varphi)^k \sin(\varphi)^{n-k}$ de forma independiente, podemos llegar a la expresión principal de la siguiente manera:

$$\sin(\varphi)^{2n-1} A = \frac{(-1)^{N-1}}{2^{2N-2}} \sum_{i=0}^{n-1} \binom{2n-1}{i} (-1)^i \sin[(2n-1-2i)A]$$

De este modo si suponemos que el valor de k es impar, tenemos a su vez que el valor de n es par.

$$\cos^k(\varphi) \sin^{n-k}(\varphi) = \left[\frac{1}{2^{k-1}} \sum_{i=0}^{\frac{k-1}{2}} \binom{k}{i} \cos[(k-2i)\varphi] \right]^*$$

$$* \left[\frac{(-1)^{\frac{n-k-1}{2}}}{2^{n-k-1}} \sum_{j=0}^{\frac{n-k-1}{2}} \binom{n-k}{j} (-1)^j \sin[(n-k-2j)\varphi] \right];$$

Donde $k = 2n - 1 \rightarrow n = \frac{k+1}{2}$, si tenemos $2n' - 1 = n - k \rightarrow n' = \frac{n-k+1}{2}$

De este modo la expresión $\cos^k(\varphi)\sin^{n-k}(\varphi)$ puede desarrollarse como:

$$\begin{aligned} &= \frac{1}{2^{k-1}} \frac{(-1)^{\frac{n-k-1}{2}}}{2^{n-k-1}} \sum_{i=0}^{\frac{k-1}{2}} \sum_{j=0}^{\frac{n-k-1}{2}} \binom{k}{i} \binom{n-k}{j} (-1)^j \cos[(k-2i)\varphi] \sin[(n-k-2j)\varphi]; \\ &= \frac{(-1)^{\frac{n-k-1}{2}}}{2^{n-2}} \sum_{i=0}^{\frac{k-1}{2}} \sum_{j=0}^{\frac{n-k-1}{2}} \binom{k}{i} \binom{n-k}{j} (-1)^j \frac{1}{2} [\sin[(n-k-2j-k+2i)\varphi] \\ &\quad + \sin[(n-k-2j-k-2i)\varphi]]; \\ &= \frac{(-1)^{\frac{n-k-1}{2}}}{2^{n-2}} \sum_{i=0}^{\frac{k-1}{2}} \sum_{j=0}^{\frac{n-k-1}{2}} \binom{k}{i} \binom{n-k}{j} (-1)^j \frac{1}{2} [\sin[(n-2(k+j-i))\varphi] \\ &\quad + \sin[(n-2(j+i))\varphi]]; \end{aligned}$$

De este modo el argumento del primer seno $(n-2(k+j-i))\varphi$ necesita $n, n-2, \dots, n-2 \left(\frac{k-1}{2} + \frac{n-k-1}{2}\right) \rightarrow n, n-2, \dots, n-(n-2) \rightarrow \mathbf{n, n-2, \dots, 2}$

A su vez el argumento del segundo seno $(n-2(j+i))\varphi$ necesita $n, n-2, \dots, n-2 \left(k + \frac{n-k-1}{2}\right) \rightarrow n, n-2, \dots, -(k-1)$; como sabemos que k es impar entonces $-(k-1)$ es par.

$$n-2 \left(k - \frac{k-1}{2}\right) = n-k-1 \text{ que es un factor par.}$$

Luego sólo se necesitan las bases $\sin(n-2i)$ para $i = 0, \dots, \frac{n}{2}$, es decir $N+1$ bases. De este modo se concluye, del mismo modo que mediante la descomposición en series de Fourier que el número de bases necesarias son $N+1$.



PROYECTO FINAL DE CARRERA

NOTA: el término $x^k y^{n-k}$ ha sido desarrollado para el caso de n impar, k impar, pero del mismo modo podría haber sido desarrollado para cualquier otra combinación.

3. Nuestro filtro orientable desarrollado

3.1. Tipos de filtros orientables

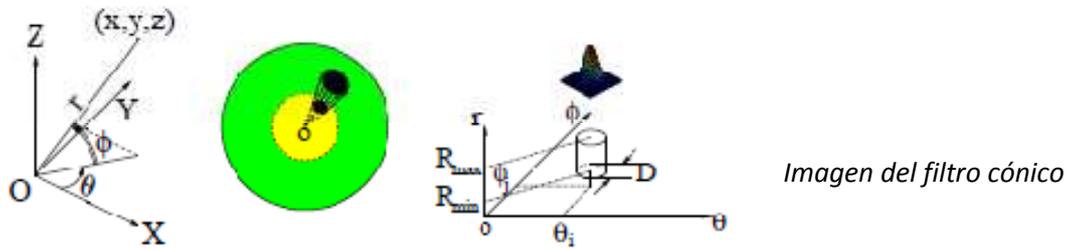
En esta sección se va a hablar de los diseños de filtros que pueden escogerse para realizar el filtrado. En principio, según el artículo de Weichuan Yu, Kostas Daniilidis, y Gerald Sommer: “*A New 3D Orientation Steerable Filter*”[6], existen varios tipos de filtros orientables según las funciones aplicadas y el espacio utilizado. En este caso se van a tratar tres.

El primer estudio fue desarrollado por Freeman y Adelson, los cuales fueron de los primeros en introducir el concepto de orientabilidad. En su caso, los filtros se centraban en bases que eran copias rotadas del mismo y las constantes de interpolación eran funciones trigonométricas basadas en parámetros de la orientación.

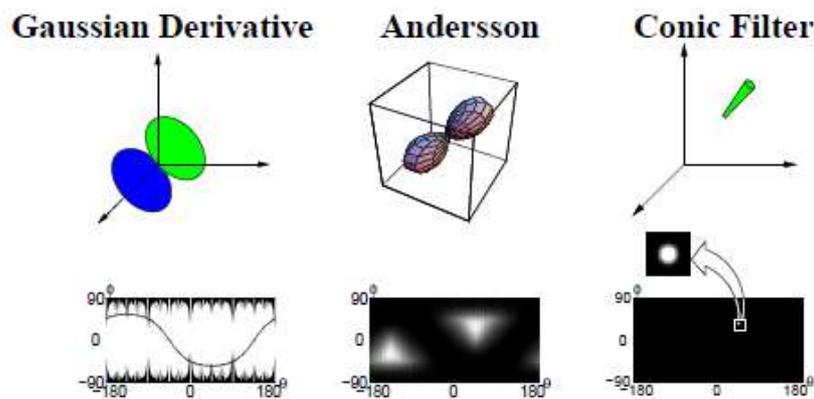
También existe otro tipo de filtro que diseñó Anderson, que al igual que el anterior, utilizaba bases que eran versiones rotadas del filtro. Sin embargo, este último no se desarrolla en un espacio de tiempo, sino en el de frecuencia. Las ventajas de éste y del anterior son la sencillez a la hora de implementar los sistemas, en cambio, los inconvenientes son que las aproximaciones no tienen una resolución fina debido a la gran superficie que hay que filtrar a lo largo de la dirección angular.

El tercer filtro es el llamado filtro de núcleo cónico. En éste, la forma del filtro es un cono que tiene su vértice en el origen del volumen. La ventaja de este filtro en comparación con los anteriores es que usando un filtro con estructura cónica se obtienen mejores resultados en la resolución.

Los tres tipos de filtros están basados en funciones gaussianas. Esto es así debido a que con este tipo de funciones se cumplen las características de orientabilidad y sin tener en cuenta que es una función fácil de tratar, tanto en el espacio temporal como el de frecuencia



El modo de filtrado que se va a escoger en este proyecto es el desarrollado por Freeman y Adelson, también conocido como *Gaussian Derivative*. Esta elección se ha tomado por dos motivos principales, que son que la mayoría de los filtros que se utilizan para las labores que se están desempeñando en este documento se basan en dicha metodología de filtrado y que los sistemas desarrollados son fácilmente adaptables, tanto para el filtrado de estructuras tubulares como para el de laminares. También hay que decir que como han sido estudiados en profundidad su eficacia está más que demostrada.



3.2. La elección del filtro orientable

En este punto del documento se va a explicar todo el proceso de decisión y creación de los dos filtros orientables, tanto para aplicaciones de filamentos (tubulares) como para las membranas celulares (laminares).

Como se ha ido desarrollando anteriormente las funciones gaussianas nos aportan numerosas ventajas por lo que este estudio se centrará en la evolución de éstas en tres dimensiones.

El desarrollo matemático de los filtros escogidos en este proyecto se han obtenido del documento de Mathews Jacob, Thierry Blu, Cedric Vaillant, John H. Maddocks y Michael Unser: “3-D Shape Estimation of DNA Molecules From Stereo Cryo-Electron Micro-Graphs Using a Projection-Steerable Snake” [2]. En dicho escrito, el filtro diseñado es utilizado para la estimación de las estructuras de las moléculas de ADN mediante proyecciones, pero en este caso, el uso que se le va a dar es para otro interés. El beneficio que tiene para nosotros radica en la eficacia que presenta para la detección de estructuras tubulares en filtros orientables. Su estructura lineal envuelta de una máscara de menor energía es muy útil para delimitar las formas de las configuraciones de los volúmenes a analizar. Además, los parámetros que tiene la función son muy versátiles, lo que supone una gran ventaja para la implementación del segundo filtro.

3.2.1. El Filtro Orientable de Estructura Tubular

En este apartado se va a describir cómo va a ser el primer filtro implementado. La notación empleada será ∂_{**} , para representar las derivadas parciales de segundo orden de una función cualquiera $\left(\partial_{**}f(r) = \left(\frac{\partial^2}{\partial^*\partial^*}\right)(f(r))\right)$. Para simplificar esta notación cuando el filtro este desarrollado matemáticamente por completo, las derivadas de la función g_3 pasarán a referirse como g_{3**} , donde cada uno de los asteriscos representa una derivada.

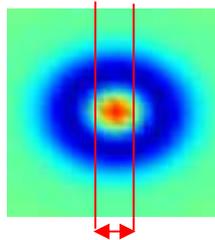
La función matemática es la que se presenta a continuación:

$$h(r) = \sqrt{\frac{3}{20}} \left(\partial_{zz}g_3(r, \sigma) + \partial_{yy}g_3(r, \sigma) - \frac{2}{3}\partial_{xx}g_3(r, \sigma) \right)$$

Como se puede observar, son tres derivadas parciales de segundo orden de la función gaussiana g_3 , que hace referencia a la función $g_D(r, \sigma)$ donde $D = 3$.

$$g_D(r, \sigma) = \frac{1}{(2\pi\sigma)^{D/2}} e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}$$

Esta función depende de dos variables r y σ . El factor σ representa el grosor de la apertura de la función, es decir, la anchura de la parte alargada. Cuanto mayor sea el valor de la misma, mayor es la amplitud del grosor del filtro. Este factor es crucial para el filtrado porque cuanto más se aproxime la anchura de la parte alargada a la del filamento, será mejor el resultado obtenido.



La ilustración anterior hace referencia a un corte del sistema de filtrado para poder observar cuál es el valor de σ . A continuación se va a mostrar cómo se ha llegado a la expresión matemática del filtro.

Una función arbitraria de un volumen en tres dimensiones puede expresarse como:

$$h = a_0 \partial_{xx} g_3 + a_1 \partial_{yy} g_3 + a_2 \partial_{zz} g_3 + a_3 \partial_{xy} g_3 + a_4 \partial_{xz} g_3 + a_5 \partial_{yz} g_3$$

Si lo que se busca es una función de simetría par orientada a lo largo del eje de abscisas, las derivadas parciales ∂_{xy} , ∂_{xz} y ∂_{yz} no contribuyen a la detección que necesitamos, o dicho de otra manera, sus coeficientes a_3 , a_4 y a_5 tendrán valor nulo. Esto es así porque dichos factores representan funciones de simetría impar. De este modo, los únicos factores que nos quedan para realizar la función de filtrado son ∂_{xx} , ∂_{yy} y ∂_{zz} cuyos coeficientes deben ser los parámetros que hay que modificar para lograr la estructura correcta.

La elongación a lo largo de los ejes puede ser adaptada mediante los coeficientes de la segunda derivada respecto al origen. De este modo, para que el “detector” tenga la máxima elongación a lo largo de la dirección del eje de abscisas y parecerse así a la forma de un filamento se coloca $\partial_{xx} h|_{0,0,0}$ a cero, con lo que conseguimos maximizar la función en dicha dirección.

$$\partial_{xx} h = \partial_{xx}(a_0 \partial_{xx} g_3) + \partial_{xx}(a_1 \partial_{yy} g_3) + \partial_{xx}(a_2 \partial_{zz} g_3)$$

La primera derivada del primer x (primer factor) $\partial_{xx}(a_0 \partial_{xx} g_3)$ es:

$$g_{3x}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{x}{\sigma^2}\right) e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}$$

Si derivamos por segunda vez obtendremos $g_{3xx} = \partial_{xx} g_3$ que es:

$$g_{3xx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2}\right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left(1 - \left(\frac{x}{\sigma}\right)^2\right)$$

Del mismo modo hay que derivar dos veces más para obtener el desarrollo total.

$$g_{3xxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2}\right) \left[\left(-\frac{x}{\sigma^2}\right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left(1 - \left(\frac{x}{\sigma}\right)^2\right) + \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left(-\frac{2x}{\sigma^2}\right) \right]$$

Sacando factor común se obtiene:

$$g_{3xxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2}\right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left[\left(-\frac{x}{\sigma^2}\right) \left(1 - \left(\frac{x}{\sigma}\right)^2\right) + \left(-\frac{2x}{\sigma^2}\right) \right]$$

Por lo que resulta que la tercera derivada es:

$$g_{3xxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2}\right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left(\frac{x^3}{\sigma^4} - \frac{3x}{\sigma^2}\right)$$

Y por último se vuelve a realizar la misma operación:

$$g_{3xxxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2}\right) \left[\left(-\frac{x}{\sigma^2}\right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left(\frac{x^3}{\sigma^4} - \frac{3x}{\sigma^2}\right) + \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left(\frac{3x^2}{\sigma^4} - \frac{3}{\sigma^2}\right) \right]$$

Sacando factor común y desarrollando la función se obtiene:

$$g_{3xxxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2}\right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}\right) \left(-\frac{x^4}{\sigma^6} + \frac{6x^2}{\sigma^4} - \frac{3}{\sigma^2}\right)$$

Ahora se sustituye la x , y , y z por cero, como se muestra condición inicial logrando:

$$g_{3xxxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(\frac{3}{\sigma^4} \right)$$

Realizando las mismas operaciones para cada uno de los factores $\partial_{xx}(a_1\partial_{yy}g_3)$ y $\partial_{xx}(a_2\partial_{zz}g_3)$ se obtiene:

$$g_{3yyxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(\frac{1}{\sigma^4} \right)$$

$$g_{3zzxx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(\frac{1}{\sigma^4} \right)$$

De este modo podemos expresar la condición $\partial_{xx}h(\mathbf{r})|_{r=(0,0,0)}$ de la siguiente manera.

$$\partial_{xx}h(\mathbf{r})|_{r=(0,0,0)} = \frac{1}{\sigma^4} (3a_0 + a_1 + a_2) = 0$$

Ahora para que el filtro sea lo más estrecho posible en la dirección de la z , y en la dirección de la y se maximiza las cantidades en dichas direcciones, igual que hacíamos en la resolución anterior.

$$\partial_{yy}h(\mathbf{r})|_{r=(0,0,0)} = \frac{1}{\sigma^4} (a_0 + 3a_1 + a_2) = 0$$

$$\partial_{zz}h(\mathbf{r})|_{r=(0,0,0)} = \frac{1}{\sigma^4} (a_0 + a_1 + 3a_2) = 0$$

Nota: hay que maximizar las funciones con las mismas condiciones que $\partial_{xx}h(\mathbf{r})|_{r=(0,0,0)}$

Debemos imponer una condición extra sobre el modulo de la expresión $h(r)$.

$$\|h(r)\|^2 = 3a_0^2 + 3a_1^2 + 3a_2^2 + 2a_0a_1 + 2a_0a_2 + 2a_1a_2 = 1$$

Como se ha demostrado anteriormente la ecuación $\partial_{xx}h(\mathbf{r})|_{r=(0,0,0)}$ permite lograr una de las condiciones, $-3a_0 = a_1 + a_2$. Si se suman $\partial_{yy}h(\mathbf{r})|_{r=(0,0,0)} +$

PROYECTO FINAL DE CARRERA

$\partial_{zz}h(\mathbf{r})|_{r=(0,0,0)} = a_0 + 4a_1 + 4a_2$ y se sustituye por la primera condición se logra $\partial_{yy}h(\mathbf{r})|_{r=(0,0,0)} + \partial_{zz}h(\mathbf{r})|_{r=(0,0,0)} = -10a_0$.

Del mismo modo si desarrollamos y sustituimos la condición en $\|h(r)\|^2$ el resultado es el siguiente:

$$\|h(r)\|^2 = 3a_0^2 + 3(a_1^2 + a_2^2) + 2a_0(a_1 + a_2) + 2a_1a_2 = 1$$

$$\text{Como } (a_1^2 + a_2^2) = \frac{(a_1+a_2)^2 + (a_1-a_2)^2}{2} \text{ y } 2a_1a_2 = \frac{(a_1+a_2)^2 - (a_1-a_2)^2}{2}$$

$$\|h(r)\|^2 = 15a_0^2 + (a_1 - a_2)^2 = 1$$

Entonces, la solución es $a_0 = \frac{-1}{\sqrt{15}}$ y $a_1 = a_2$, las cuales simplificando quedan $a_1 = \frac{3}{2\sqrt{15}}$ y $a_2 = \frac{3}{2\sqrt{15}}$.

De este modo la ecuación resultante es:

$$h(r) = \frac{3}{2\sqrt{15}}\partial_{zz}g_3(r, \sigma) + \frac{3}{2\sqrt{15}}\partial_{yy}g_3(r, \sigma) - \frac{1}{\sqrt{15}}\partial_{xx}g_3(r, \sigma)$$

Y sacando factor común a las constantes para normalizar se obtiene la función del filtro.

$$h(r) = \sqrt{\frac{3}{20}} \left(\partial_{zz}g_3(r, \sigma) + \partial_{yy}g_3(r, \sigma) - \frac{2}{3}\partial_{xx}g_3(r, \sigma) \right)$$

Como hemos visto, en esta función los factores están multiplicados por una serie de coeficientes. Si se normaliza la fórmula para reescribirla a lo largo de la dirección arbitraria v , y se considera despreciable el primer coeficiente, la expresión resultante es la siguiente:

$$h_v(r) = (\partial_{zz} + \partial_{yy} + \partial_{xx})g_3(r, \sigma) - \frac{5}{3}\partial_{vv}g_3(r, \sigma)$$

La segunda parte del filtro $\partial_{vv}g_3(r, \sigma)$ representa la segunda derivada direccional de la función en una dirección arbitraria v . La derivada direccional de una función cualquiera, $f(\vec{x}) = f(x_1, x_2, \dots, x_n)$ sobre un vector unitario $\vec{v} = (v_1, v_2, \dots, v_n)$ es la función definida por este límite:

$$D_{\vec{v}}f = \lim_{h \rightarrow 0} \frac{f(\vec{x}+h\vec{v}) - f(\vec{x})}{h}$$

Si la función es diferenciable, puede ser escrita en término de su gradiente ∇f , como se demuestra en el apartado 3.2.3.

$$D_{\vec{v}}f = \nabla f \cdot \vec{v}$$

Donde el \cdot denota el producto escalar o producto interno entre vectores.

De este modo podemos realizar el siguiente cálculo para desarrollar $\partial_{vv}g_3(r, \sigma)$. Con la formula $D_{\vec{v}}f = \nabla f \cdot \vec{v}$ aplicada sobre sí misma para obtener la segunda derivada se logra:

$$D^2_{vv}f = D_v(D_vf)$$

Desarrollando la primera derivada se obtiene:

$$D^2_{vv}f = \frac{\partial}{\partial x}(D_vf)v_i + \frac{\partial}{\partial y}(D_vf)v_j + \frac{\partial}{\partial z}(D_vf)v_k$$

Si se sigue desarrollando resulta la expresión:

$$\begin{aligned} D^2_{vv}f &= \frac{\partial}{\partial x} \left[\frac{\partial f}{\partial x} v_i + \frac{\partial f}{\partial y} v_j + \frac{\partial f}{\partial z} v_k \right] v_i + \frac{\partial}{\partial y} \left[\frac{\partial f}{\partial x} v_i + \frac{\partial f}{\partial y} v_j + \frac{\partial f}{\partial z} v_k \right] v_j \\ &+ \frac{\partial}{\partial z} \left[\frac{\partial f}{\partial x} v_i + \frac{\partial f}{\partial y} v_j + \frac{\partial f}{\partial z} v_k \right] v_k \end{aligned}$$

Y si se agrupan los términos da lugar a:

$$\begin{aligned} D^2_{vv}f &= \frac{\partial f}{\partial^2 x} v_i^2 + \frac{\partial f}{\partial yx} v_j v_i + \frac{\partial f}{\partial zx} v_k v_i + \frac{\partial f}{\partial xy} v_i v_j + \frac{\partial f}{\partial^2 y} v_j^2 + \frac{\partial f}{\partial zy} v_k v_j \\ &+ \frac{\partial f}{\partial xz} v_i v_k + \frac{\partial f}{\partial yz} v_j v_k + \frac{\partial f}{\partial^2 z} v_k^2 \end{aligned}$$

Aplicando ahora el teorema de Schwartz, se pueden unir las derivadas parciales de la siguiente forma:

$$D^2_v f = \frac{\partial f}{\partial^2 x} v_i^2 + \frac{\partial f}{\partial^2 y} v_j^2 + \frac{\partial f}{\partial^2 z} v_k^2 + \frac{\partial f}{\partial zy} v_k v_j \\ + 2 \left[\frac{\partial f}{\partial xy} v_i v_j + \frac{\partial f}{\partial xz} v_i v_k + \frac{\partial f}{\partial yz} v_j v_k \right]$$

Sustituyendo la formula final de $D^2_v f$ en la función del filtro y empleando la notación que viene siendo usada, la función principal queda de la siguiente manera:

$$h_v(r) = g_{3zz}(r, \sigma) + g_{3yy}(r, \sigma) + g_{3xx}(r, \sigma) \\ - \frac{5}{3} (g_{3xx}(r, \sigma) v_x^2 + g_{3yy}(r, \sigma) v_y^2 + g_{3zz}(r, \sigma) v_z^2 + 2g_{3xy}(r, \sigma) v_x v_y \\ + 2g_{3xz}(r, \sigma) v_x v_z + 2g_{3yz}(r, \sigma) v_y v_z)$$

Donde v es el vector director de la función.

Como se puede ver, es importante averiguar cuáles son las derivadas parciales de g_3 para xx, yy, zz, xy, xz e yz.

$$g_{3xx}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2} \right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(1 - \left(\frac{x}{\sigma} \right)^2 \right)$$

$$g_{3yy}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2} \right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(1 - \left(\frac{y}{\sigma} \right)^2 \right)$$

$$g_{3zz}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{1}{\sigma^2} \right) \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(1 - \left(\frac{z}{\sigma} \right)^2 \right)$$

$$g_{3xy}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{y}{\sigma^2} \right) e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}$$

$$g_{3xz}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}$$

$$g_{3yz}(r, \sigma) = \frac{1}{(2\pi\sigma)^{3/2}} \left(-\frac{y}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}$$

Si cada uno de los términos $g_{3xx}(r, \sigma)$, $g_{3yy}(r, \sigma)$ y $g_{3zz}(r, \sigma)$, los agrupamos obtenemos:

$$h_v(r) = g_{3zz}(r, \sigma) \left(1 - \frac{5}{3} v_z^2\right) + g_{3yy}(r, \sigma) \left(1 - \frac{5}{3} v_y^2\right) + g_{3xx}(r, \sigma) \left(1 - \frac{5}{3} v_x^2\right) + \frac{10}{3} g_{3xy}(r, \sigma) v_x v_y + \frac{10}{3} g_{3xz}(r, \sigma) v_x v_z + \frac{10}{3} g_{3yz}(r, \sigma) v_y v_z$$

Por último, para mostrar el filtro de la forma $W(r)P_N(x)$, que es la que necesitamos para aplicar el teorema de Freeman en tres dimensiones (que será explicado en detalle en el apartado 4), se sustituirá cada función gaussiana desarrollada anteriormente y se sacará factor común a $\frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right)$;

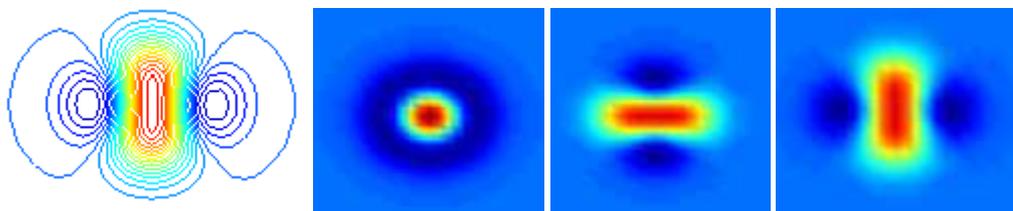
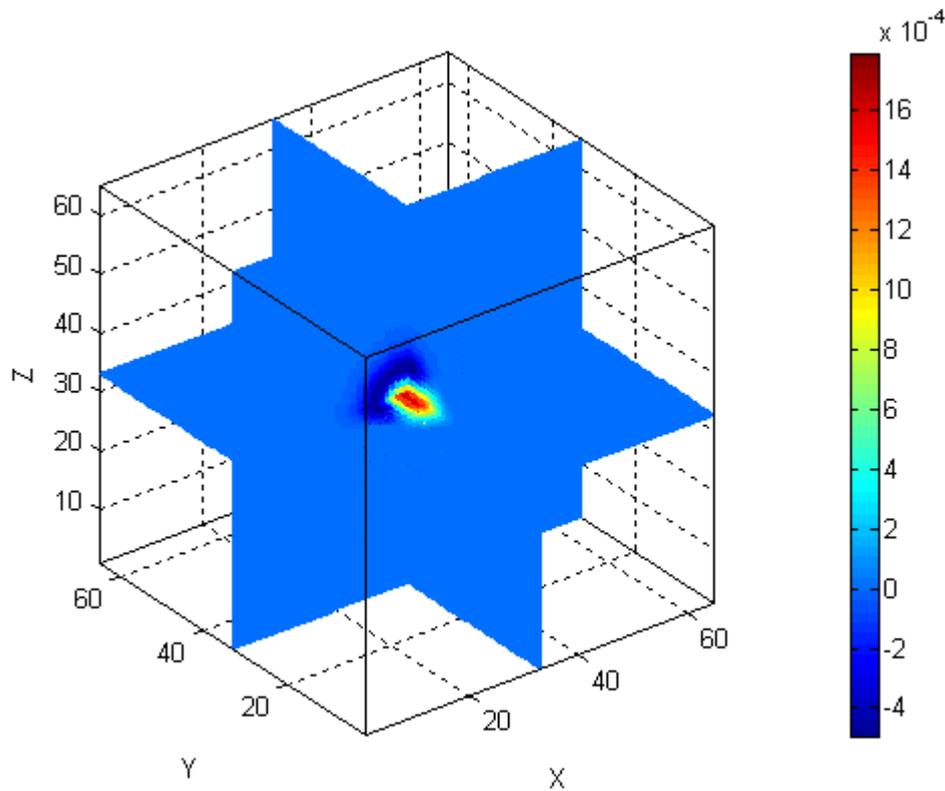
Por lo tanto el resultado final es el que se muestra a continuación:

$$h_v(r) = \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left[\left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{z}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_z^2 \right) + \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{y}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_y^2 \right) + \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{x}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_x^2 \right) + \frac{10}{3} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{y}{\sigma^2} \right) v_x v_y + \frac{10}{3} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) v_x v_z + \frac{10}{3} \left(-\frac{y}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) v_y v_z \right]$$

Esta función se podría separar en dos partes como $f^R(x, y, z) = \mathbf{W}(r)\mathbf{P}_N(x)$ donde $\mathbf{W}(r)$ es cualquier función de inventanado de simetría esférica

$r = \sqrt{x^2 + y^2 + z^2}$, y $\mathbf{P}_N(x)$ es un polinomio de orden n.

A continuación se muestra una construcción de la función del filtro en Matlab para una $\sigma = 3$ y un volumen de 65x65x65.



La imagen que no presenta un fondo azul es la representación de la estructura del filtro mediante la función 3D-contours del *toolbox* volumen_browser. Las imágenes restantes representan los cortes del filtro en $x = 33$, $y = 33$ y $z = 33$ respectivamente.

Desde este punto hay que realizar el proceso de orientación del filtro. Para ello hay que obtener las bases a partir de la función desarrollada y calcular los coeficientes de interpolación.

Para la obtención de las bases únicamente hay que decidir cuáles son las direcciones que van a tomar, es decir, el valor del vector \vec{v} . Es importante saber que se pueden escoger las bases de modo arbitrario siempre y cuando entre ellas no sean linealmente dependientes.

3.2.2. El Filtro Orientable de Estructura Laminar

Para la implementación de este filtro se ha partido de la estructura del filtro tubular, dado que cambiando sus coeficientes, es posible adaptarlo para las estructuras laminares. De este modo e igual que en la sección anterior, la idea es maximizar las expresiones para obtener los coeficientes apropiados para desarrollar la ecuación.

$$h = a_0 \partial_{xx} g_3 + a_1 \partial_{yy} g_3 + a_2 \partial_{zz} g_3 + a_3 \partial_{xy} g_3 + a_4 \partial_{xz} g_3 + a_5 \partial_{yz} g_3$$

Por el contrario, el objetivo ahora es intentar que el filtro tenga una estructura lo más aplanada posible, por lo que en vez de realizar la operación de elongación anterior centrándonos en la dirección de x, se hará en las direcciones y y z, consiguiendo así un sistema plano. También es importante obtener la mínima anchura posible, en la dirección de la x, para lograr la máxima detección en el filtrado.

Dado que las operaciones son similares y para no parecer redundante, se va a mostrar únicamente algunas expresiones.

Si de la misma forma que antes partimos de las derivadas parciales de segundo orden de h con los coeficientes a_3 , a_4 y a_5 con valor nulo, se empieza el desarrollo desde la siguiente formula.

$$\partial_{xx} h = \partial_{xx}(a_0 \partial_{xx} g_3) + \partial_{xx}(a_1 \partial_{yy} g_3) + \partial_{xx}(a_2 \partial_{zz} g_3)$$

$$\partial_{yy} h = \partial_{yy}(a_0 \partial_{xx} g_3) + \partial_{yy}(a_1 \partial_{yy} g_3) + \partial_{yy}(a_2 \partial_{zz} g_3)$$

$$\partial_{zz} h = \partial_{zz}(a_0 \partial_{xx} g_3) + \partial_{zz}(a_1 \partial_{yy} g_3) + \partial_{zz}(a_2 \partial_{zz} g_3)$$

Dado que las derivadas parciales son las mismas en este caso lo único que hay que cambiar es la relación de la maximización de las formulas.

$$\partial_{yy} h(\mathbf{r})|_{\mathbf{r}=(0,0,0)} = \frac{1}{\sigma^4} (a_0 + 3a_1 + a_2) = 0$$

$$\partial_{zz} h(\mathbf{r})|_{\mathbf{r}=(0,0,0)} = \frac{1}{\sigma^4} (a_0 + a_1 + 3a_2) = 0$$

De este modo las formulas anteriores se convierten en un sistema de ecuaciones dependientes del coeficiente a_0 . Si los desarrollamos para obtener la solución de la primera ecuación obtenemos:

$$a_1 = -\frac{(a_0 + a_2)}{3}$$

Si sustituimos a_1 en la segunda expresión ($\partial_{zz}h(\mathbf{r})|_{r=(0,0,0)}$); obtenemos el siguiente resultado:

$$a_0 - \frac{a_0}{3} - \frac{a_2}{3} + 3a_2 = 0;$$

$$a_0 \left(1 - \frac{1}{3}\right) + a_2 \left(3 - \frac{1}{3}\right) = 0;$$

$$-\frac{2}{3}a_0 = \frac{8}{3}a_2;$$

$$a_2 = -\frac{1}{4}a_0;$$

De este modo también obtenemos el valor de a_1

$$a_1 = -\frac{(a_0 + a_2)}{3} = -\frac{\left(a_0 + \left(-\frac{1}{4}a_0\right)\right)}{3} = -\frac{a_0 \left(1 - \frac{1}{4}\right)}{3} = \frac{a_0 \frac{3}{4}}{3} = -\frac{1}{4}a_0;$$

$$\begin{aligned} \|h(r)\|^2 &= 3a_0^2 + 3\left(-\frac{1}{4}a_0\right)^2 + 3\left(-\frac{1}{4}a_0\right)^2 + 2a_0\left(-\frac{1}{4}a_0\right) + 2a_0\left(-\frac{1}{4}a_0\right) \\ &\quad + 2\left(-\frac{1}{4}a_0\right)^2 = 1 \end{aligned}$$

$$\|h(r)\|^2 = 3a_0^2 + \frac{1}{2}a_0^2 - a_0^2 = 1$$

$$a_0 = \sqrt{\frac{2}{5}}$$

$$h_v(r) = \left(-\frac{1}{4} \sqrt{\frac{2}{5}} \partial_{zz} - \frac{1}{4} \sqrt{\frac{2}{5}} \partial_{yy} + \sqrt{\frac{2}{5}} \partial_{xx} \right) g_3(r, \sigma)$$

$$h_v(r) = \sqrt{\frac{2}{5}} \left(-\frac{1}{4} \partial_{zz} - \frac{1}{4} \partial_{yy} + \partial_{xx} \right) g_3(r, \sigma)$$

Si se adapta la expresión obtenida a la derivada direccional de un vector arbitrario \vec{v} y se desprecia el coeficiente $\sqrt{\frac{2}{5}}$, dado que no implica cambio significativo en el resultado del filtrado, la función resultante es la siguiente:

$$h_v(r) = -\frac{1}{4} (\partial_{zz} + \partial_{yy} + \partial_{xx}) g_3(r, \sigma) + \frac{5}{4} \partial_{vv} g_3(r, \sigma)$$

Si ahora se desarrolla esta expresión se obtiene:

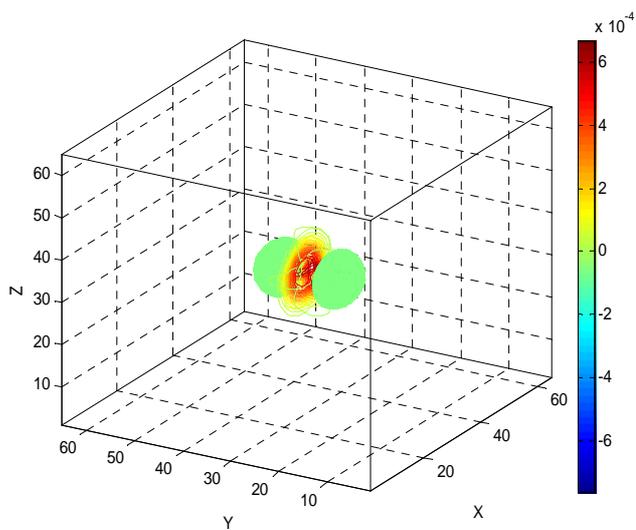
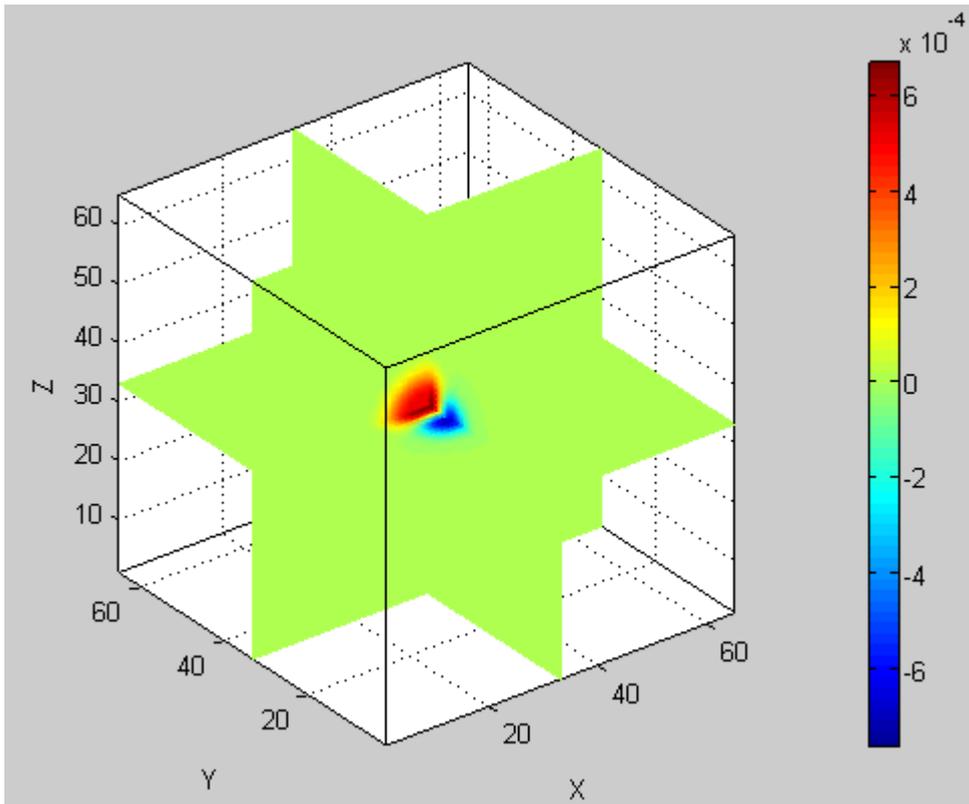
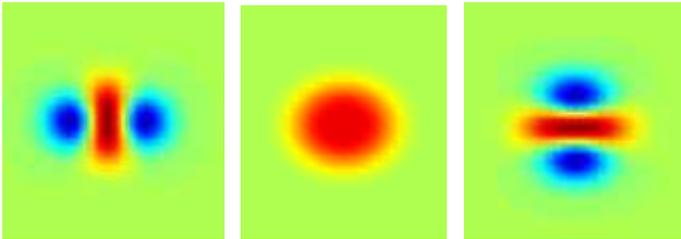
$$\begin{aligned} h_v(r) = & -\frac{1}{4} (g_{3zz}(r, \sigma) + g_{3yy}(r, \sigma) + g_{3xx}(r, \sigma)) \\ & + \frac{5}{4} (g_{3xx}(r, \sigma) v_x^2 + g_{3yy}(r, \sigma) v_y^2 + g_{3zz}(r, \sigma) v_z^2 + 2g_{3xy}(r, \sigma) v_x v_y \\ & + 2g_{3xz}(r, \sigma) v_x v_z + 2g_{3yz}(r, \sigma) v_y v_z) \end{aligned}$$

Agrupando ahora los conjuntos que tengan las mismas derivadas se consigue la siguiente fórmula:

$$\begin{aligned} h_v(r) = & g_{3zz}(r, \sigma) \left(\frac{5}{4} v_z^2 - \frac{1}{4} \right) + g_{3yy}(r, \sigma) \left(\frac{5}{4} v_y^2 - \frac{1}{4} \right) + g_{3xx}(r, \sigma) \left(\frac{5}{4} v_x^2 - \frac{1}{4} \right) \\ & + \frac{10}{4} g_{3xy}(r, \sigma) v_x v_y + \frac{10}{4} g_{3xz}(r, \sigma) v_x v_z + \frac{10}{4} g_{3yz}(r, \sigma) v_y v_z \end{aligned}$$

$$h_v(r) = \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left[\left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{z}{\sigma} \right)^2 \right) \left(\frac{5}{4} v_z^2 - \frac{1}{4} \right) + \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{y}{\sigma} \right)^2 \right) \left(\frac{5}{4} v_y^2 - \frac{1}{4} \right) + \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{x}{\sigma} \right)^2 \right) \left(\frac{5}{4} v_x^2 - \frac{1}{4} \right) + \frac{10}{4} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{y}{\sigma^2} \right) v_x v_y + \frac{10}{4} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) v_x v_z + \frac{10}{4} \left(-\frac{y}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) v_y v_z \right]$$

PROYECTO FINAL DE CARRERA



01-Mar-2009 19:08:10

3.2.3. Aproximación de la derivada direccional en funciones diferenciables

El caso más sencillo de la derivada direccional se da en el espacio tridimensional. Supóngase que se tiene una función diferenciable $z = f(x, y)$. La derivada direccional según un vector $v = (v_x, v_y)$ de la superficie $f(x, y)$ sería:

$$D_v f = \lim_{h \rightarrow 0} \frac{f(x + v_x h, y + v_y h) - f(x, y)}{h}$$

$$D_v f = \lim_{h \rightarrow 0} \frac{f(x + v_x h, y + v_y h) - f(x, y + v_y h) + f(x, y + v_y h) - f(x, y)}{h}$$

$$D_v f = \lim_{h \rightarrow 0} \frac{f(x + v_x h, y + v_y h) - f(x, y + v_y h)}{h} + \lim_{h \rightarrow 0} \frac{f(x, y + v_y h) - f(x, y)}{h}$$

El primero de estos límites puede calcularse mediante el cambio $h' = v_x h$ lo cual lleva, por ser diferenciable la función f , a:

$$\begin{aligned} \lim_{h' \rightarrow 0} \frac{f(x + h', y + v_y h' / v_x) - f(x, y + v_y h' / v_x)}{h' / v_x} \\ = \lim_{h' \rightarrow 0} v_x \frac{\partial f(x, y + v_y h' / v_x)}{\partial x} = v_x \frac{\partial f(x, y)}{\partial x} \end{aligned}$$

NOTA: Es importante que la función sea diferenciable, porque si no lo es, las derivadas parciales no son continuas y esta demostración no es válida.

Procediendo análogamente para el otro límite se obtiene:

$$D_v f = v_x \frac{\partial f(x, y)}{\partial x} + v_y \frac{\partial f(x, y)}{\partial y}$$

Resultado que trivialmente coincide con el producto escalar del gradiente por el vector $v = (v_x, v_y)$;

$$(\nabla f) \cdot v = \left(\frac{\partial f(x, y)}{\partial x}, \frac{\partial f(x, y)}{\partial y} \right) \cdot (v_x, v_y) = v_x \frac{\partial f(x, y)}{\partial x} + v_y \frac{\partial f(x, y)}{\partial y} = D_v f$$

Para los campos vectoriales el concepto de derivada direccional se puede generalizar a funciones de \mathbb{R}^m en \mathbb{R}^n , del tipo:

$$\mathbf{F} : A \subset \mathbb{R}^m \longrightarrow \mathbb{R}^n$$

En este caso la derivada direccional de modo idéntico a como se hacía con funciones de una variable:

$$D_v F = \lim_{h \rightarrow 0} \frac{F(x + hv) - F(x)}{h}$$

Una diferencia con el caso de funciones de reales de una variable es que la existencia de derivadas direccionales según todas las direcciones no implica necesariamente que una función sea diferenciable. Si la función es diferenciable resulta que la aplicación:

$$v \rightarrow D_v \mathbf{F}$$

Es lineal y se cumple además es expresable en términos del jacobiano:

$$D_v \mathbf{F} = (D\mathbf{F})v$$

3.3. Características del filtro escogido: funciones separables

En este punto del proyecto se van a describir las características del filtro que se ha escogido. Es importante saber sus cualidades y sus limitaciones para luego analizar los resultados obtenidos.

Una de las características más efectivas para el análisis que posee el filtro implementado es su carácter separable. Lo que viene a decir esto, es que las operaciones que se realizan sobre el filtro pueden aplicarse independientemente en cada una de las coordenadas sin la alteración de los resultados. Esto es sumamente ventajoso debido a que el manejo de las operaciones es mucho más simple.

El ejemplo más sencillo donde puede observarse esta característica es la operación de convolución. Como se puede intuir, dicha operación es un proceso que

PROYECTO FINAL DE CARRERA

viene implícito en todo tratamiento digital de señales. La convolución para un volumen en tres dimensiones tiene una gran carga operacional, pero si logramos expresar el filtro en coordenadas cartesianas y realizar la operación para cada una de las componentes (x,y,z) podemos disminuir enormemente ese tiempo. El porqué de esto, reside en la simplicidad que tiene la operación de convolución para un vector de cualquier tamaño respecto a un volumen de la misma capacidad. Dicho esto, también es importante mencionar que esta propiedad es sumamente ventajosa en cualquier proyecto biotecnológico, debido a que los volúmenes con los que se opera, suelen ocupar mucho espacio.

En principio todas las funciones no son separables, pero sí que se puede decir, que para las funciones polinómicas, que no cumplen dicha propiedad, existen un conjunto de bases que si que pueden escribirse como funciones separables. De esta manera, se pueden adaptar los filtros para conseguir las ventajas operacionales que suponen los sistemas separables. Con la anterior afirmación, es importante aclarar cuál es la metodología para poder conseguir que cualquier función polinómica pueda expresarse en bases separables. A continuación se muestra la forma de un polinomio expresado mediante bases separables.

$$f^\theta(x, y, z) = \sum_l \sum_j \sum_p k_{ljp}(\theta) x^l y^j z^p$$

Donde cada producto de x , y , z , en el polinomio es una de las bases de la función, junto con sus coeficientes de interpolación $k_{ljp}(\theta)$. Los valores de los exponentes l, j y p representan el orden del polinomio para cada una de las bases.

Con esta propiedad se muestra realmente la importancia de las funciones polinómicas para este tipo de filtrado. También es interesante plantearse en qué medida se presentan dichas ventajas respecto a otro tipo de funciones. En este caso hay que decir que existen funciones que sólo contienen un número mínimo de bases y que además contienen todos los puntos de interés del espacio. Estas son las que mejor eficiencia computacional tienen y son las que se están aplicando en el filtro de este proyecto. Aquí radica uno de los motivos por lo que la función que se ha diseñado es

una polinómica con enventanado, dado que aunque no sea en sí misma una función separable podemos obtener bases que cumplan dicha función.

A continuación se va a mostrar cómo encontrar las funciones orientables con bases separables para cualquier filtro con estructura de polinomio. Se va a considerar sólo el caso de funciones con simetría par o simetría impar que es el caso del filtro que se muestra en este proyecto. De este modo, la función orientable puede ser escrita como:

$$f^\theta(x, y) = G(r)Q_N(x')$$

Donde $G(r)$ es una función gaussiana y por lo tanto puede separarse en bases claramente definidas en x - y de la forma:

$$g(r, \sigma) = \frac{1}{(2\pi\sigma)^{D/2}} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

$$g_D(r, \sigma) = \frac{1}{(2\pi\sigma)^{D/2}} \left(e^{-\frac{x^2}{2\sigma^2}} \right) \left(e^{-\frac{y^2}{2\sigma^2}} \right)$$

Base en x Base en y

Y $Q_N(x')$ es un polinomio de orden n , donde $x' = x\cos(\theta) - y\sin(\theta)$.

Como puede observarse, esta ilustración está realizada para un sistema de dos dimensiones en coordenadas cartesianas. Esto es para simplificar, en medida de lo posible, la comprensión de la matemática del documento.

Si partimos del teorema de Freeman para funciones polinómicas, documentado en el apartado 2.1.3 y demostrado en el apartado 2.1.4, $N+1$ bases son suficientes para desarrollar el filtro de simetría, par o impar. Asumiendo esto último y sabiendo que realmente existen $N+1$ bases para este tipo de filtros, afirmación que no es cierta para todas las funciones, existirán un conjunto de bases separables $R_j(x)S_j(y)$ para las cuales se cumple que:

$$f^\theta(x, y) = G(r) \sum_{j=0}^N k_j(\theta) R_j(x) S_j(y)$$

En este caso el término $G(r)$ representa los coeficientes de la función gaussiana. Además puede verse en la ecuación que el sumatorio va desde $j = 0$ hasta $j = N$ lo que representan las $N+1$ bases necesarias.

Como hemos visto, la función gaussiana puede expresarse como bases separables de x e y , de igual modo, ahora lo que se necesita es demostrar que se puede hacer lo mismo con el polinomio $Q_N(x')$. Esto implica que la función $x' = x \cos(\theta) - y \sin(\theta)$ elevada a un orden N se pueda expresar de forma separable mediante bases.

Como lo que queremos es que el término x' este elevado a cualquier orden, expresaremos dicho término como $(x')^N$. Ahora habrá que igualar las formulas $f^\theta(x, y) = G(r)Q_N(x')$ y $f^\theta(x, y) = G(r) \sum_{j=0}^N k_j(\theta) R_j(x) S_j(y)$ para conseguir nuestro objetivo.

$$G(r)Q_N(x') = G(r) \sum_{j=0}^N k_j(\theta) R_j(x) S_j(y)$$

Si simplificamos la gaussiana a ambos lados obtendríamos:

$$Q_N(x') = \sum_{j=0}^N k_j(\theta) R_j(x) S_j(y)$$

Ahora es necesario utilizar los órdenes mayores de x e y $(x^{(N-j)}y^j)$ para $0 \leq j \leq N$ a ambos lados de la expresión. De esta manera obtendríamos:

$$(x')^N = \sum_{j=0}^N k_j(\theta) x^{(N-j)} y^j$$

Por último hay que encontrar el valor de $k_j(\theta)$ que en este caso es:

$$k_j(\theta) = (-1)^j \binom{N}{j} \cos^{(N-j)}(\theta) \sin^j(\theta)$$

Con lo que la expresión general de la función queda de la forma

$$(x')^N = \sum_{j=0}^N (-1)^j \binom{N}{j} \cos^{(N-j)}(\theta) \sin^j(\theta) [x^{(N-j)} y^j]$$

Para encontrar los valores de $k_j(\theta)$ hay que recurrir a la matriz separable, similar a las de los teoremas de Freeman, que se muestra a continuación

$$\begin{pmatrix} f^{\theta_1}(x, y) \\ f^{\theta_2}(x, y) \\ \vdots \\ f^{\theta_N}(x, y) \end{pmatrix} = \begin{pmatrix} k_1(\theta_1) & k_2(\theta_1) & \vdots & k_N(\theta_1) \\ k_1(\theta_2) & k_2(\theta_2) & \vdots & k_N(\theta_2) \\ \vdots & \vdots & \vdots & \vdots \\ k_1(\theta_N) & k_2(\theta_N) & \dots & k_N(\theta_N) \end{pmatrix} \begin{pmatrix} R_1(x)S_1(y) \\ R_2(x)S_2(y) \\ \vdots \\ R_N(x)S_N(y) \end{pmatrix}$$

De este modo las bases $R_j(x)S_j(y)$ pueden ser escritas como combinación lineal de $f^{\theta_j}(x, y)$ mediante la inversión de la matriz de los coeficientes de interpolación.

4. Aplicación de la condición de orientabilidad 3D de Freeman al filtro diseñado

Un paso en falso que se ha dado en este proyecto, ha sido el análisis de las funciones de interpolación en tres dimensiones mediante la tesis de Freeman [1]. Por desgracia, como se analizará a continuación, no es viable dicho estudio para el filtro que se ha escogido. Este error, que pasó inadvertido en los estudios preliminares, resultó catastrófico en las soluciones finales, lo que condujo a un replanteamiento de la matemática original del proyecto. Por otra parte, el conocimiento adquirido en la búsqueda del problema ha enriquecido los resultados del proyecto, además de conseguir una posible solución satisfactoria en la resolución final.

4.1. El problema

Con la elección de la función del filtro realizada, se pasó al siguiente punto del proyecto, que fue el realizar la orientabilidad del filtro. De este modo, y como se ha comentado anteriormente, se tomó como base el planteamiento matemático del artículo de Freeman, en este caso, la aplicación de su teorema para tres dimensiones. La expresión que se muestra a continuación es la formula inicial, donde $f^R(x, y, z)$ es la función del filtro después de aplicarle una rotación R en tres dimensiones, y R_j identifica la orientación de cada una de las j^{th} bases.

$$f^R(x, y, z) = \sum_{j=1}^M k_j(R) f^{R_j}(x, y, z)$$

Teniendo la función del filtro escogida para la realización de las bases, la única parte de la formula anterior que faltaba, eran los coeficientes de interpolación $k_j(R)$. Estos indicarían cuál es el peso específico que aportaría cada una de las bases a $f^R(x, y, z)$ para un R dado.

Como el filtro es un sistema polinómico, de la forma $f^R(x, y, z) = W(r)P_N(x')$ había que aplicar el teorema del apartado 2.1.3 adaptado a sistemas en tres dimensiones. En este caso el inventariado seguía siendo de simetría par pero de forma esférica.

Para el caso de dos dimensiones el número de bases necesarias dependían de las diferentes sinusoides presentes en la descomposición angular de Fourier, o como se ha demostrado del desarrollo del punto 2.1.4, del carácter simétrico de la función. De manera análoga, en tres dimensiones, se puede realizar la misma descomposición aunque la solución produciría armónicos de carácter esférico. El desarrollo de estos armónicos es complejo y no se van a desarrollar pero vienen descritos en el documento de R. Courant and D. Hilbert “Methods of Mathematical Physics” [13]. Como se demostrará a lo largo de este apartado el número de bases necesarias para este tipo de funciones es $(N + 1)^2$ en vez de sólo $N + 1$ que eran las necesarias en dos dimensiones.

Si partimos de esto, el teorema se explicaría de la siguiente manera:

Dada una función de tres dimensiones con simetría axial de la forma $f(x, y, z) = W(r)P_N(x)$, donde $P_N(x)$ es un polinomio de orden N de simetría par o impar. Y del mismo modo, siendo α, β y γ los cosenos directores de los ejes de simetría de $f^R(x, y, z)$ y α_j, β_j y γ_j los de $f^{Rj}(x, y, z)$. Entonces la ecuación

$$f^R(x, y, z) = \sum_{j=1}^M k_j(\alpha, \beta, \gamma) f^{Rj}(x, y, z)$$

Se mantiene, si y sólo si:

$$(a) M \geq \frac{(N+1)(N+2)}{2}$$

(b) $k_j(\alpha, \beta, \gamma)$; El cual tiene que cumplir la siguiente relación:

$$\begin{pmatrix} \alpha^N \\ \alpha^{(N-1)}\beta \\ \alpha^{N-1}\gamma \\ \alpha^{N-2}\beta^2 \\ \vdots \\ \gamma^N \end{pmatrix} = \begin{pmatrix} \alpha_1^N & \alpha_2^N & \dots & \alpha_M^N \\ \alpha_1^{(N-1)}\beta_1 & \alpha_2^{(N-1)}\beta_2 & \dots & \alpha_M^{(N-1)}\beta_M \\ \alpha_1^{(N-1)}\gamma_1 & \alpha_2^{(N-1)}\gamma_2 & \dots & \alpha_M^{(N-1)}\gamma_M \\ \vdots & \vdots & \dots & \vdots \\ \gamma_1^N & \gamma_2^N & \dots & \gamma_M^N \end{pmatrix} \begin{pmatrix} k_1(\alpha, \beta, \gamma) \\ k_2(\alpha, \beta, \gamma) \\ k_3(\alpha, \beta, \gamma) \\ \vdots \\ k_M(\alpha, \beta, \gamma) \end{pmatrix}$$

Si la función de enventanado es una función gaussiana $W(r)$ entonces $f^R(x, y, z)$ de la forma que se expresa en $f^R(x, y, z) = \sum_{j=1}^M k_j(R) f^{Rj}(x, y, z)$ puede ser escrita sustituyendo $x' = \alpha x + \beta y + \gamma z$ en $f^R(x, y, z) = W(r) P_N(x')$

4.1.1. Demostración del teorema de Freeman para tres dimensiones

Partiendo de la siguiente ecuación:

$$f^R(x, y, z) = \sum_{j=1}^M k_j(\alpha, \beta, \gamma) f^{Rj}(x, y, z)$$

Si se igualan sólo los términos de mayor orden de un polinomio cualquiera cuya variable sea x' y se divide en ambos lados la parte de enventanado, $W(r)$, obtenemos:

$$(\alpha x + \beta y + \gamma z)^N = \sum_{j=1}^M k_j(\alpha_j x + \beta_j y + \gamma_j z)^N$$

La expresión del polinomio puede ser cualquiera porque se sustituyen a ambos lados de la ecuación.

De este modo, expandiendo la potencia enésima en ambos lados, igualando las potencias de $[x, y, z]$ y expresando el resultado en forma matricial se obtiene la siguiente relación:

$$\begin{pmatrix} \alpha^N \\ \alpha^{(N-1)}\beta \\ \alpha^{N-1}\gamma \\ \alpha^{N-2}\beta^2 \\ \vdots \\ \gamma^N \end{pmatrix} = \begin{pmatrix} \alpha_1^N & \alpha_2^N & \dots & \alpha_M^N \\ \alpha_1^{(N-1)}\beta_1 & \alpha_2^{(N-1)}\beta_2 & \dots & \alpha_M^{(N-1)}\beta_M \\ \alpha_1^{(N-1)}\gamma_1 & \alpha_2^{(N-1)}\gamma_2 & \dots & \alpha_M^{(N-1)}\gamma_M \\ \vdots & \vdots & \dots & \vdots \\ \gamma_1^N & \gamma_2^N & \dots & \gamma_M^N \end{pmatrix} \begin{pmatrix} k_1(\alpha, \beta, \gamma) \\ k_2(\alpha, \beta, \gamma) \\ k_3(\alpha, \beta, \gamma) \\ \vdots \\ k_M(\alpha, \beta, \gamma) \end{pmatrix}$$

Esta matriz resultante para cualquier término de orden menor del polinomio de f^R y f^{Rj} puede generar condiciones linealmente dependientes. Esto se demuestra del siguiente modo. Si se consideran los coeficientes $x^p y^q z^r$ en la expresión $f^R(x, y, z) = \sum_{j=1}^M k_j(\alpha, \beta, \gamma) f^{Rj}(x, y, z)$, para potencias de valores $p + q + r < N$ se obtiene;

$$\alpha^p \beta^q \gamma^r = \sum_{j=1}^M k_j \alpha_j^p \beta_j^q \gamma_j^r$$

Asumiendo que $P_N(x')$ tiene una simetría par o impar, las potencias de x' se pueden diferenciar sólo mediante enteros pares. Si se considera el resultado de los coeficientes de los términos en $f^R(x, y, z) = \sum_{j=1}^M k_j(\alpha, \beta, \gamma) f^{R_j}(x, y, z)$ del orden de $p + q + r + 2$ podría existir, por lo menos, las siguientes tres ecuaciones:

$$\alpha^{p+2} \beta^q \gamma^r = \sum_{j=1}^M k_j \alpha_j^{p+2} \beta_j^q \gamma_j^r$$

$$\alpha^p \beta^{q+2} \gamma^r = \sum_{j=1}^M k_j \alpha_j^p \beta_j^{q+2} \gamma_j^r$$

$$\alpha^p \beta^q \gamma^{r+2} = \sum_{j=1}^M k_j \alpha_j^p \beta_j^q \gamma_j^{r+2}$$

Ahora, utilizando el hecho de que la suma de los cuadrados de los cosenos directores es uno, se puede sustituir $\alpha^2 = 1 - \beta^2 - \gamma^2$ y $\alpha_j^2 = 1 - \beta_j^2 - \gamma_j^2$ en cada una de las ecuaciones anteriores obteniendo como resultado la ecuación $f^R(x, y, z) = \sum_{j=1}^M k_j(\alpha, \beta, \gamma) f^{R_j}(x, y, z)$ de la que partíamos. Así, todas las condiciones resultantes que no se contemplan en la matriz de los términos del polinomio de orden n , son linealmente dependientes en la condición resultante del polinomio de orden $n + 2$. Por lo tanto, si las condiciones de la matriz se satisfacen y el polinomio $P_N(x')$ contiene términos de orden par o de orden impar, entonces la expresión $f^R(x, y, z) = \sum_{j=1}^M k_j(\alpha, \beta, \gamma) f^{R_j}(x, y, z)$ se cumple. Porque hay $\frac{(N+1)(N+2)}{2}$ condiciones en la matriz, debe haber un $M \geq \frac{(N+1)(N+2)}{2}$.

Si hubiéramos procedido a la inversa se demostraría de igual modo que son necesarias un número de bases M mayor que $\frac{(N+1)(N+2)}{2}$ para que se cumplan los requisitos de la matriz de interpolación y así que se cumpla la condición de orientabilidad en la expresión inicial.

4.2. La demostración del error

Si partimos de la expresión de nuestro filtro:

$$h_v(r) = \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{r^2}{2\sigma^2}} \right) \left[\left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{z}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_z^2 \right) + \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{y}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_y^2 \right) + \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{x}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_x^2 \right) + \frac{10}{3} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{y}{\sigma^2} \right) v_x v_y + \frac{10}{3} \left(-\frac{x}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) v_x v_z + \frac{10}{3} \left(-\frac{y}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) v_y v_z \right]$$

Se sabe que para que el teorema de Freeman se cumpla, el polinomio debe depender de x' . De este modo la expresión final de polinomio, dado que es de segundo orden, debería poder expresarse como:

$$a + b(x + y + z) + c(x + y + z)^2$$

$$a + b(x + y + z) + c[(x + y)^2 + z^2 + 2(x + y)z]$$

$$a + b(x + y + z) + c[x^2 + y^2 + z^2 + 2xy + 2xz + 2yz]$$

Donde a es el término independiente, b son los coeficientes de primer orden de x , y y z , y c son los coeficientes de segundo orden de x , y y z . De este modo hay que ver si se puede expresar el filtro según dicha estructura. Si se determina la dirección del vector $v(1,1,1)$ el polinomio resultante sería el siguiente:

$$P(x') = \left[\left(-\frac{1}{\sigma^2} + \frac{5}{3\sigma^2} + \left(\frac{1}{\sigma^4} - \frac{5}{3\sigma^4} \right) z^2 \right) + \left(-\frac{1}{\sigma^2} + \frac{5}{3\sigma^2} + \left(\frac{1}{\sigma^4} - \frac{5}{3\sigma^4} \right) y^2 \right) + \left(-\frac{1}{\sigma^2} + \frac{5}{3\sigma^2} + \left(\frac{1}{\sigma^4} - \frac{5}{3\sigma^4} \right) x^2 \right) - \frac{5}{3\sigma^4} (2xy + 2xz + 2yz) \right]$$

NOTA: En rojo se han puesto los términos de a y en azul los términos de c (No existen términos para b por lo que su valor es cero)

Entonces para obtener la expresión de c de nuestro filtro tiene que cumplir que $c = \frac{1^2}{\sigma^4} - \frac{5}{3\sigma^4}$ para los términos z^2, y^2 y x^2 y también que $c = -\frac{5}{3\sigma^4}$ para los términos $2xy, 2xz$ y $2yz$. Entonces tendría que poder existir la siguiente igualdad:

$$\frac{1^2}{\sigma^4} - \frac{5}{3\sigma^4} = -\frac{5}{3\sigma^4}$$

$$\boxed{-\frac{2}{3\sigma^4} \neq -\frac{5}{3\sigma^4}}$$

Por lo tanto no puede expresarse el polinomio que nosotros queremos de la manera x' lo que supone que no se puede aplicar el teorema de freeman.

Como el filtro de estructura laminar se basa en la misma estructura de filtrado es imposible conseguir un resultado diferente, lo que implica que ambos filtros no pueden ser objeto de estudio mediante el sistema de tres dimensiones de Freeman.

Aunque la construcción del filtro se obtuviera mediante los principios de los teoremas de Freeman con estos lo único que se pretendía obtener era la certeza de que la función implementada era perfectamente orientable. Dicho esto, ahora se tiene el impedimento de que el filtro no es aplicable en uno de sus teoremas, pero como veremos en la siguiente sección se encontró una posible solución simplificando la matemática de las funciones.

5. La solución del sistema de filtrado

Como se ha explicado en el apartado anterior, el filtro que había sido seleccionado, no cumple los requisitos del teorema de Freeman del polinomio en tres dimensiones. Por ese motivo se ha desarrollado una solución para los dos filtros (el tubular y el laminar).

A lo largo del proyecto se ha demostrado que el fundamento básico para obtener un filtro orientable es conseguir unas bases que cumplan una serie de características y que mediante una combinación lineal de las mismas multiplicadas por unos ciertos coeficientes pueda lograrse un filtro arbitrario en cualquier dirección. De este modo, si analizamos la expresión de nuestro filtro:

$$h_v(r) = g_{3zz}(r, \sigma) \left(1 - \frac{5}{3} v_z^2\right) + g_{3yy}(r, \sigma) \left(1 - \frac{5}{3} v_y^2\right) + g_{3xx}(r, \sigma) \left(1 - \frac{5}{3} v_x^2\right) + \frac{10}{3} g_{3xy}(r, \sigma) v_x v_y + \frac{10}{3} g_{3xz}(r, \sigma) v_x v_z + \frac{10}{3} g_{3yz}(r, \sigma) v_y v_z$$

Podemos ver que se puede adaptar para cumplir dichas características.

$$h_v(r) = \boxed{g_{3zz}(r, \sigma)} \boxed{\left(1 - \frac{5}{3} v_z^2\right)} + \boxed{g_{3yy}(r, \sigma)} \boxed{\left(1 - \frac{5}{3} v_y^2\right)} + \boxed{g_{3xx}(r, \sigma)} \boxed{\left(1 - \frac{5}{3} v_x^2\right)} + \boxed{g_{3xy}(r, \sigma)} \boxed{\left(\frac{10}{3} v_x v_y\right)} + \boxed{g_{3xz}(r, \sigma)} \boxed{\left(\frac{10}{3} v_x v_z\right)} + \boxed{g_{3yz}(r, \sigma)} \boxed{\left(\frac{10}{3} v_y v_z\right)}$$

Si cogemos como bases los términos encuadrados en rojo, es decir, las derivadas gaussianas de segundo orden, y como coeficientes de interpolación los términos encuadrados en azul, podemos ver que la estructura fundamental de los sistemas orientables se mantiene. A partir de esto, hay que verificar que se puede realizar el filtrado con esta nueva configuración. Para ello es necesario analizar cada uno de los componentes que presenta la función.

Las bases de este nuevo esquema serán simplemente funciones gaussianas, y como se ha demostrado en el apartado 1.3.1.1, la derivada direccional de este tipo de

PROYECTO FINAL DE CARRERA

funciones es orientable. Esta última afirmación verifica que no existe ningún impedimento en el uso de estos términos como bases. Sin embargo, como se explicaba en secciones anteriores, las bases son las que cumplen la función de orientar los filtros, no como sucede en esta nueva configuración. Realmente, si se analiza con detenimiento cuál es el objetivo de los filtros orientables, se puede afirmar que no es importante qué componente es el que realiza la orientabilidad del filtro, sino que realmente el conjunto sea orientable. De esta manera, para nuevo filtro, las componentes del vector de dirección forman parte de los coeficientes, por lo que los pesos específicos para cada una de las bases determinarán los valores de la dirección del filtro.

De esta forma la estructura de nuestros filtros para la programación en los diferentes lenguajes será la siguiente.

$$h_v(r) =$$

$$\begin{aligned} & \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{z}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_z^2 \right) + \\ & \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{y}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_y^2 \right) + \\ & \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(-\frac{1}{\sigma^2} \right) \left(1 - \left(\frac{x}{\sigma} \right)^2 \right) \left(1 - \frac{5}{3} v_x^2 \right) + \\ & \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(-\frac{x}{\sigma^2} \right) \left(-\frac{y}{\sigma^2} \right) \left(\frac{10}{3} v_x v_y \right) + \\ & \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(-\frac{x}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) \left(\frac{10}{3} v_x v_z \right) + \\ & \frac{1}{(2\pi\sigma)^{3/2}} \left(e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}} \right) \left(-\frac{y}{\sigma^2} \right) \left(-\frac{z}{\sigma^2} \right) \left(\frac{10}{3} v_y v_z \right) \end{aligned}$$

Nota: Para el filtro de estructura laminar lo único que cambian son los términos mostrados en azul.

Una de las ventajas que se mantienen en esta estructura es que el número de bases es igual que en el anterior método, lo que supone que los tiempos de ejecución serán similares.

6. El muestreo esféricos de los filtros

Como se ha venido explicando a lo largo del documento, los componentes principales para obtener un sistema de filtrado orientable son las bases y los coeficientes de interpolación. Por otra parte, para que el sistema pueda ser implementado y que obtenga las estructuras de los volúmenes biológicos, es importante que el filtro sea orientado en todas las direcciones posibles. De esta manera habrá personas que se pregunten qué diferencia existe entre la idea inicial de un investigador, explicada en la introducción, respecto al sistema que se tiene ahora. La explicación de esto radica en el tiempo a la hora de obtener los resultados.

Si para realizar el filtrado en una dirección concreta hace falta un tiempo t_1 y realizamos dicha operación N veces el tiempo resultante de todo el filtrado será Nt_1 . En cambio, si realizamos el sistema con filtros orientables, hará falta filtrar el volumen con esa metodología un número M de veces, que son las bases del sistemas, y luego habría que obtener los coeficientes de interpolación para cada una de las direcciones. Si consideramos que t_2 es el tiempo de búsqueda de dichos coeficientes, que es infinitamente menor que t_1 , el tiempo final para un sistema orientable sería: $Mt_1 + Nt_2$. Como las direcciones tienen que ser las mayores posibles para detectar bien la estructura del volumen, en cualquiera de los dos métodos el número de iteraciones de búsqueda, N , tiene que ser muy alto. Por otro lado si hemos diseñado el filtro orientable para que tenga solamente seis bases, el valor de M es muy bajo respecto a N . De ese modo el resultado de los tiempos para cada uno de ellos es:

Sistema orientable: $Mt_1 + Nt_2. \rightarrow Nt_2$.

Como N es muy grande y M es muy pequeña el factor Mt_1 es despreciable, lo que nos lleva a que el resultado es Nt_2 .

Sistema no orientable: Nt_1 .

Como puede verse el sistema orientable es mucho mejor que el otro. Para cuantificar en qué medida lo es, se desarrollarán cálculos específicos en los apartados del capítulo 9.

PROYECTO FINAL DE CARRERA

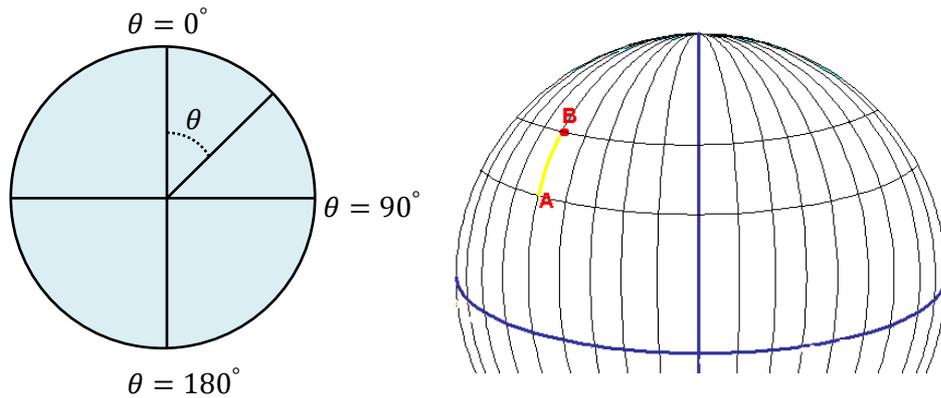
El cálculo de N parte de saber cuáles son las necesidades del volumen. Si el volumen es sencillo, no harán falta muchas muestras para determinar su posición y dirección. Por el contrario, en volúmenes reales esto no pasa, por lo que es bueno que el muestreo sea abundante.

Para realizar el muestreo correctamente no solo es importante el número de muestras, si no la dirección que representan las mismas y por tanto la posición entre unas u otras. Como se puede intuir el método más apropiado para el muestreo es que la distancia entre las muestras sea la más regular posible, es decir, que las muestras estén equidistantes entre sí.

Como lo que nos importa son las direcciones, no se puede realizar el estudio en coordenadas cartesianas, dado que el valor para una dirección $x = 1, y = 1, z = 1$ tendría un resultado es similar a $x = -1, y = -1, z = -1$ o $x = 2, y = 2, z = 2$, ya que presentan el mismo vector director. De este modo las coordenadas utilizadas deberían ser las polares. Sin embargo, esto deriva en otro problema de programación, ya que no es fácil determinar un sistema que haga un muestreo en diferentes direcciones del espacio y que a la vez tome la distancia de dichas muestras de forma equitativa.

Si cogiésemos por ejemplo un número de muestras N y hubiera que repartirlas a lo largo de una esfera no sería posible otorgar un número constante de muestras por cada ángulo posible desde el origen, porque la distancia entre las muestras no sería la misma para cada ángulo. Lo que viene a decir la afirmación anterior, es que si repartimos N muestras de forma constante, para valores de θ (mostrado en la ilustración posterior) desde $\theta = 0$ hasta $\theta = 180$, los valores que se encuentren más próximos a los valores de $\theta = 0$ y $\theta = 180$ tendrán más muestras que los que se encuentran próximos a $\theta = 90$.

PROYECTO FINAL DE CARRERA



De este modo si consideramos θ en los puntos A y B, la distancia entre las muestras de B será menor que las de A, debido a que hay que repartir el mismo número de muestras en menor perímetro.

Para la solución de este problema se realizó el siguiente cálculo:

Si sabemos que el valor máximo que puede tomar θ es 180° y dividimos dicho valor entre un entero, podemos obtener los valores θ_j para los que se asignaran un conjunto de muestras variables dependiendo del ángulo dado. De este modo, $deltaAng$ será a partir de ahora dicho entero y N_{tilt} será el valor del ángulo de muestreo $\left[N_{tilt} = \text{round}\left(\frac{180}{deltaAng}\right) \right]$. Ahora, para cada valor de θ_j se calcularán cuantas muestras son necesarias para el muestreo mediante la formula $deltaRot = \frac{deltaAng}{abs(sind(\theta_j))}$. De este modo ya sabemos el número de ángulos θ_j que hay y el número de muestras que tienen cada uno de ellos. Para repartir dichas muestras a lo largo de las circunferencias que se forman en la esfera por los ángulos θ_j , se va a proceder del mismo modo que en el caso de θ . Se dividirán los 360° de las circunferencias generadas por el número de muestras de cada θ_j . Este dato se conocerá como N_{rot} $\left[N_{rot} = \text{round}\left(\frac{360}{deltaRot}\right) \right]$. Por lo tanto ya sabemos la distancia que hay que tomar entre muestras para un mismo ángulo θ_j . De esta manera conocemos la orientación de cada una de las muestras de la esfera.

Los valores $\theta_j = 0^\circ$ y $\theta_j = 180^\circ$ tomarán únicamente una muestra dado que sólo representan una dirección.

PROYECTO FINAL DE CARRERA

Como el diseño del filtro está implementado en coordenadas cartesianas el valor el vector director de las muestras debe ser expresado en dichas coordenadas y no en polares por lo que hay que hacer un cambio de coordenadas de la forma:

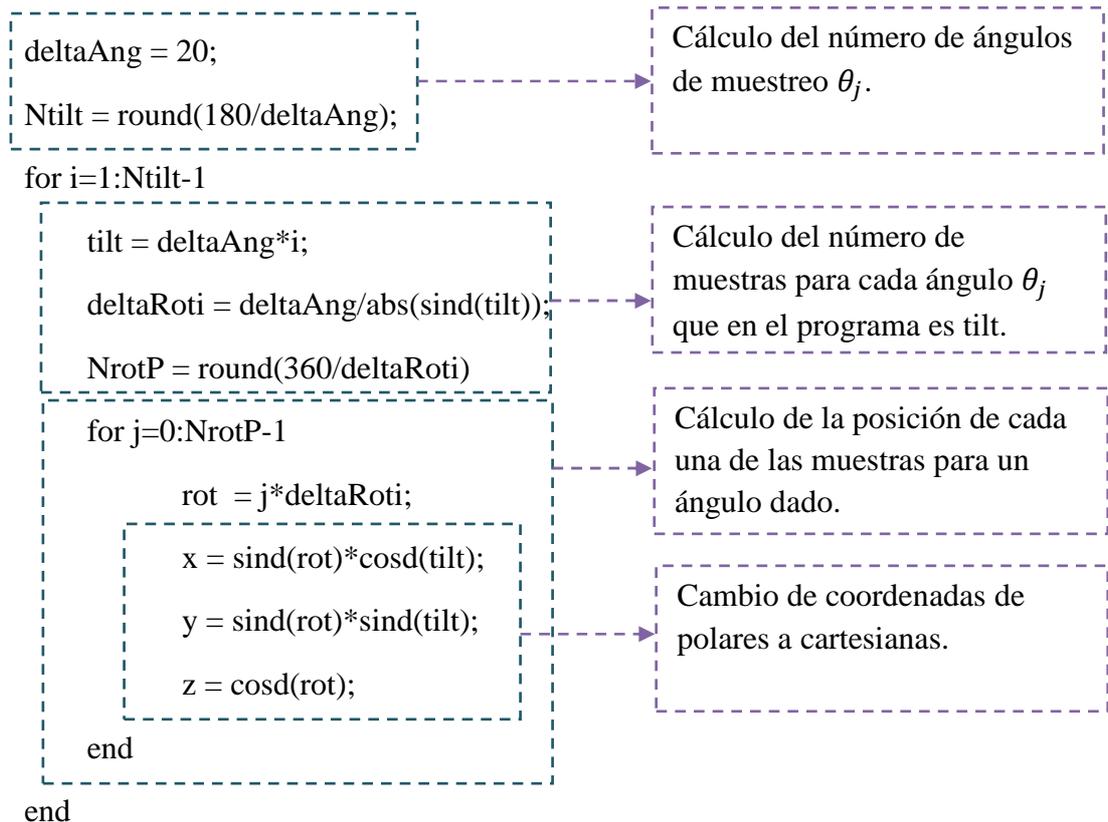
Siendo rot la posición de las muestras dentro de la circunferencia formada por el ángulo θ_j . El cambio de coordenadas es el siguiente:

$$x = \text{sind}(rot) \text{cosd}(\theta_j)$$

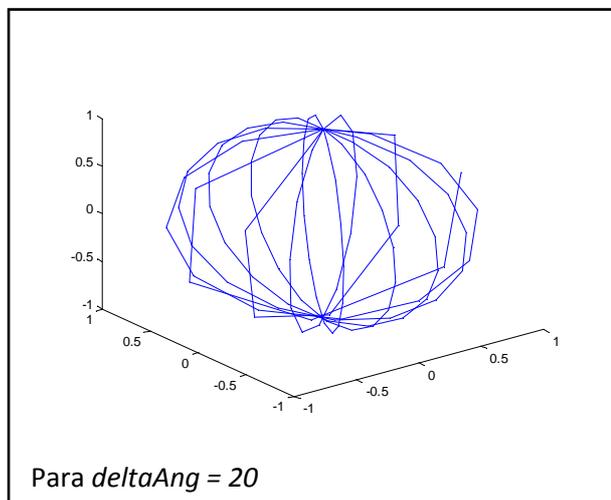
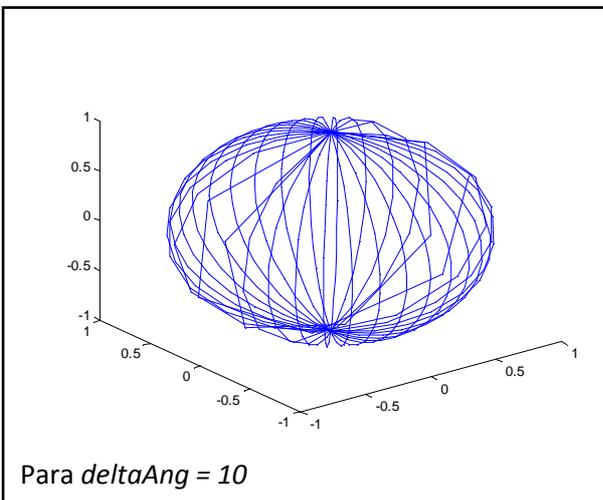
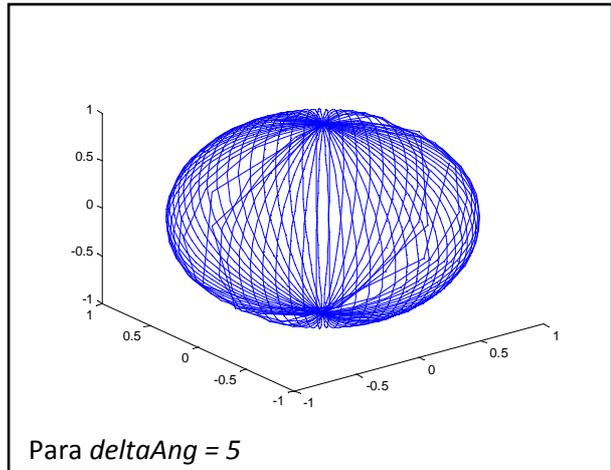
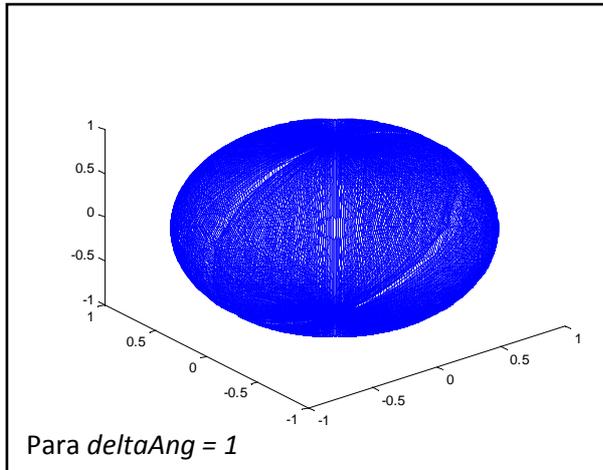
$$y = \text{sind}(rot) \text{sind}(\theta_j)$$

$$z = \text{cosd}(rot)$$

A continuación se mostrará dicha programación para verificar el entendimiento de la solución de muestreo.



Ahora se mostrarán volúmenes representativos de la función de muestreo para ver la calidad resultante dependiendo del número de muestras. Cada punto azul representa una muestra y por lo tanto una dirección para realizar el filtrado.



Como se ha observado el parámetro de calidad es deltaAng , pero este valor no representa el número de muestras total que se toman para el análisis sino el número entero que divide al ángulo θ .

NOTA: A partir de este punto del proyecto se considerará que la implementación de los filtros tubulares y laminares ya se ha realizado en Matlab y en C++. No se ha desarrollado ninguna explicación de la implementación del código dado que ya han sido explicadas las configuraciones matemáticas y se presentan los códigos fuente de los programas en los anexos. Esto no significa que la programación de estas librerías haya sido trivial.

7. Resultados

La parte más importante de un proyecto son los resultados que se obtienen una vez desarrollado el estudio completo. No obstante, en esta sección se van a presentar dos tipos de resultados. El primero de ellos, es análisis del filtrado de unos volúmenes llamados *Phantom*, que son volúmenes simbólicos con estructuras de filamentos y membranas. El segundo es un estudio completo con volúmenes reales de gran tamaño. El análisis de estos últimos son los que se pueden considerar como el resultado final del proyecto, dado que son volúmenes similares a los que se utilizan en las investigaciones de biotecnología.

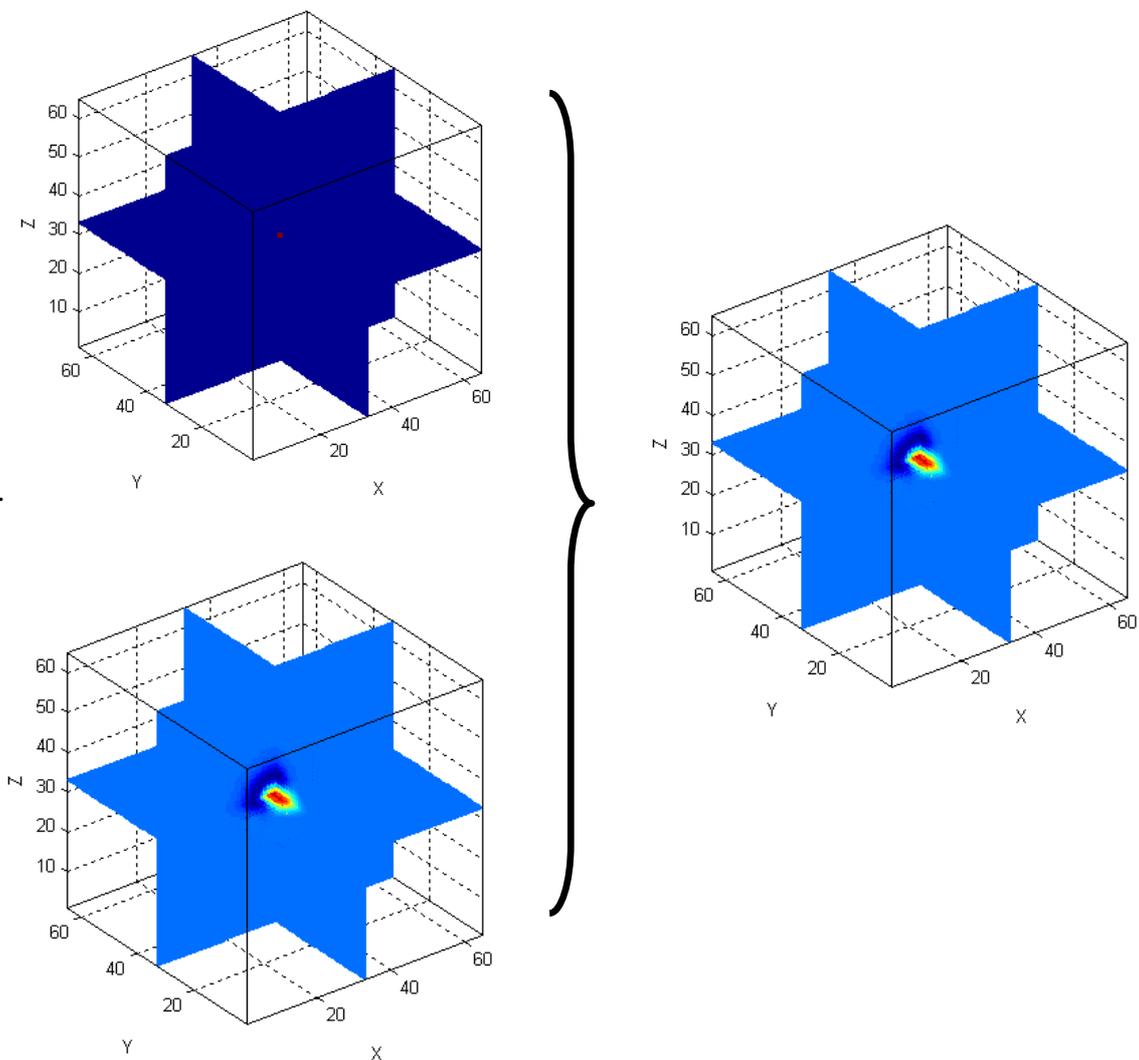
Por otra parte es importante saber que el uso de volúmenes simulados es crucial para el desarrollo de la investigación debido a la versatilidad que presentan. Esto es así porque están formados por estructuras simples y fácilmente diferenciables en el contorno. Además, estos volúmenes, tienen tamaños mucho más pequeños que los reales, lo que supone una mayor sencillez a la hora de manejarlos y de visualizar los posibles errores. Un buen resultado en este tipo de volúmenes es fundamental para asegurar que los volúmenes reales tengan unos resultados correctos. De hecho, gracias a éstos se pudo detectar que nuestros filtros no se podían implementar mediante la tesis de Freeman en tres dimensiones. Además han servido para corregir algunos errores de implementación que han pasado inadvertidos en los volúmenes reales por ser demasiado complejos a la hora de visualizar los resultados.

7.1. Análisis de los Volúmenes Simulados

Como se ha venido comentando a lo largo del documento la esencia de los filtros es la detección de ciertas estructuras en volúmenes. Para poder analizar los resultados en la profundidad, es importante saber de antemano si las funciones que se han implementado en la práctica cumplen los propósitos que se exponían en el análisis matemático principal. Para ello es importante partir la estructura inicial del filtro, sin utilizar sistemas de orientabilidad, ni buscar coeficientes de interpolación, es decir, corroborar que las funciones pueden detectar estructuras por sí mismas en una dirección concreta. Para llegar a esto se analizaron varios puntos críticos del filtrado.

El primero punto fue asegurar que las convoluciones de los filtros con volúmenes a analizar se realizaban de forma correcta, independientemente de si la función era separable o no. En este caso, la forma más sencilla para demostrarlo es hacer que el volumen a analizar sea una función delta, dado que el resultado de la convolución entre un filtro y un volumen que posee una única delta centrada en el origen, es la misma función del filtro.

A continuación se muestran los resultados del proceso para uno de los filtros

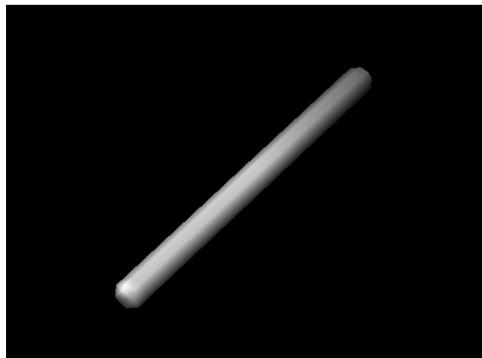


Comprobada la validez de la convolución se debe verificar si cada uno de los filtros cumple su labor de detección en unas direcciones concretas. Esto supone coger cada una de las funciones $h_v(x, y)$ de los apartados del diseño de los filtros (3.2.1 y

PROYECTO FINAL DE CARRERA

3.2.2) y sin realizar ninguna acción de obtención de bases o concreción de los coeficientes de interpolación, obtener el filtrado de un volumen con una estructura simbólica mediante los sistemas diseñados. Este punto es importante porque sí se sabe la forma de los volúmenes resultantes en ciertas direcciones, se puede detectar si existen errores en la implementación o en el diseño de los filtros. Como es lógico pensar el resultado de este último punto no es determinante, dado que el número de muestras que se analizan no es igual al muestreo del sistema completo. Por otra parte, esta depuración sí que puede dar una pequeña aproximación de cómo puede quedar el resultado final del filtrado. Para que los resultados sean coherentes las orientaciones de las funciones tienen que ser iguales a las de las estructuras de los volúmenes a analizar porque todavía no son filtros orientables en cualquier dirección. Para que se cumpla dicha premisa el vector \vec{v} , que expresa la dirección del filtro, tiene que tener el mismo valor que la orientación que la estructura del volumen.

Ahora se va a mostrar cómo podría ser la forma en tres dimensiones de un volumen filamentoso simbólico para el filtro de detección de estructuras tubulares.

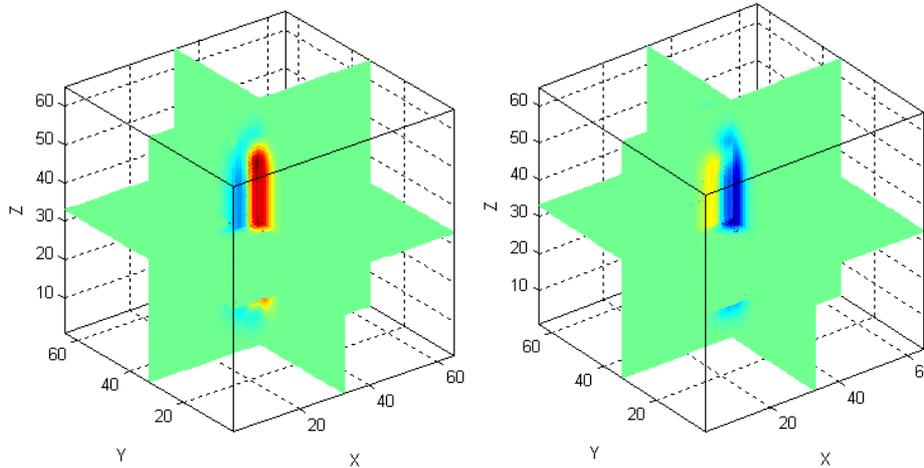


Chimera: Volumen Phantom del filamento

Como se muestra en la ilustración, al ser una estructura alargada y con un radio de grosor bastante pequeño el volumen presenta las mismas características que un filamento.

Ahora para verificar la función del filtro se va a mostrar el resultado del proceso de filtrado para dos volúmenes mediante el sistema tubular. Uno de volúmenes tendrá

un filamento simbólico orientado a lo largo de la dirección del filtro y el otro no. La dirección del la estructura tubular es la de un filamento vertical.

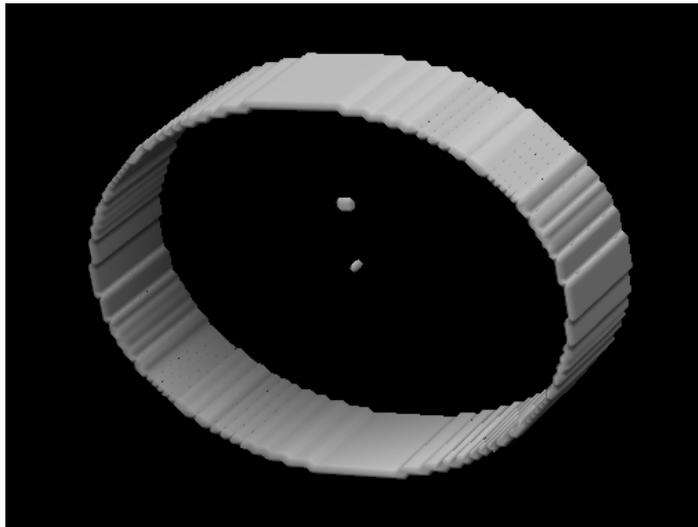


Los resultados para el volumen de la izquierda, donde el filtro ha sido orientado en la misma dirección que la imagen, son válidos dado que ha detectado que la máxima energía se encontraba en esa dirección. Por otro lado el volumen de la izquierda cuyo filtrado se ha hecho en la dirección otra dirección no ha detectado correctamente la estructura. De este modo se verifica que cuando la función se alinea en la misma dirección que la estructura del volumen, el filtro detecta la máxima energía.

Los tamaños de los volúmenes son de 65x65x65 lo que representa para este tipo de filtrados un volumen pequeño.

Como el lógico pensar, aunque la dirección del vector \vec{v} es relevante en este tipo de estructuras, su sentido no lo es, ya que, debido a la simetría del filamento la orientación en $v(1,1,1)$ será la misma que para $v(-1, -1, -1)$. Esta consideración será tomada en cuenta a la hora obtener la orientación en estructuras más complejas mediante los metodos de muestreo.

Para la simulación del filtro laminar se ha utilizado un volumen simbólico que presenta una estructura cilíndrica. Dicha configuración es semejante a la que se puede encontrar en formas biológicas como membranas o paredes celulares. El tamaño del volumen es de 128x128x32. A continuación se muestra una representación del mismo mediante el programa Chimera.



Chimera: Volumen Phantom de la pared celular

Como puede verse, a diferencia del anterior, este sistema es aplanado, aunque presenta de manera similar la relevancia del vector \vec{v} en lo referente a su dirección y sentido.

En la ilustración existen dos cuerpos más pequeños en el interior de la membrana (cilindro externo) que se utilizan para verificar si pueden detectarse estructuras de tamaños diferentes en el mismo volumen. Estas características serán analizadas en las secciones posteriores con mayor precisión.

La verificación de la función laminar no es viable utilizando el mismo proceso que para el anterior filtro. Esto es resultado de la incapacidad de detección de toda la estructura que presenta el volumen mediante una sola orientación de filtrado. De este modo la validez se reflejará en secciones posteriores cuando se utilicen los métodos de orientabilidad. Por otra parte, como sabemos que la función del filtro de detección laminar es una evolución de la del filtro tubular es prácticamente imposible que los resultados sean erróneos únicamente para una de las dos estructuras.

Habiendo analizado los resultados de las funciones y observando que, en principio, son coherentes, se va a proceder a realizar el análisis de los resultados para los sistemas orientables, es decir, para la función completa.

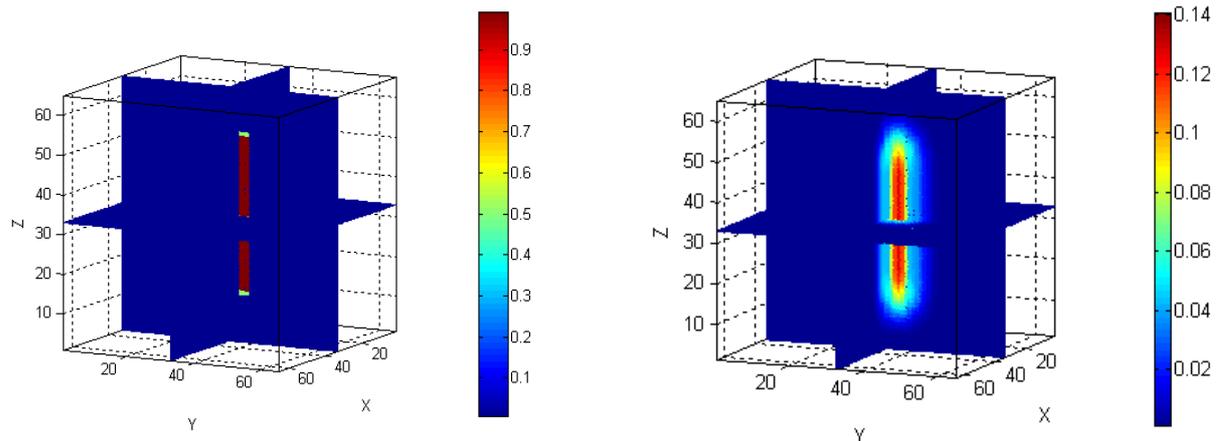
Como se comentó en la estructura del documento (sección 1.4) y en otras secciones, los filtros han sido diseñados para dos lenguajes de programación. En los dos apartados posteriores se mostrarán cada uno de ellos en dichos lenguajes

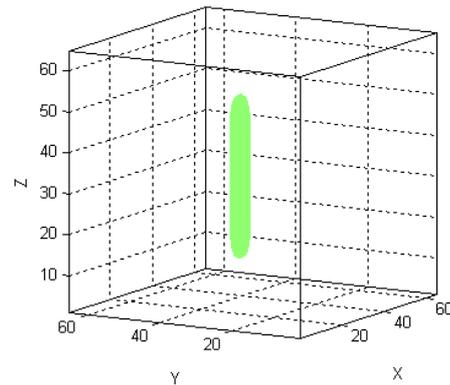
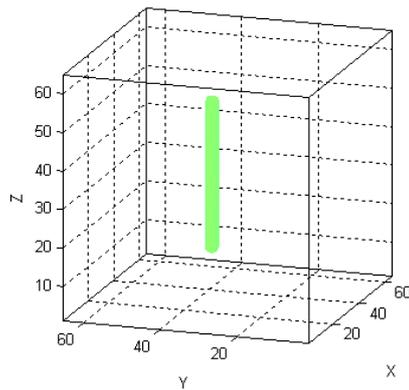
7.1.1. Matlab

7.1.1.1. Filtro de estructura tubular

En esta sección se analizarán los resultados de los filtros completos, es decir, usando las técnicas de orientabilidad. Como se mostro en la sección 6.1, la estructura para el filtrado se basa en el muestreo del filtro mediante diferentes valores del vector \vec{v} orientados a lo largo de una esfera en donde la distancia entre las muestras es la misma. El parámetro para determinar el número de muestras era conocido como *deltaAng*.

Ahora para analizar los resultados se empezará por una orientación sencilla, un volumen con un único filamento vertical de tamaño 65x65x65, para que la visualización no sea muy engorrosa. Los parámetros para el estudio serán $\sigma = 3$ y $deltaAng = 20$. De este modo los resultados obtenidos se muestran a la derecha del volumen de referencia.

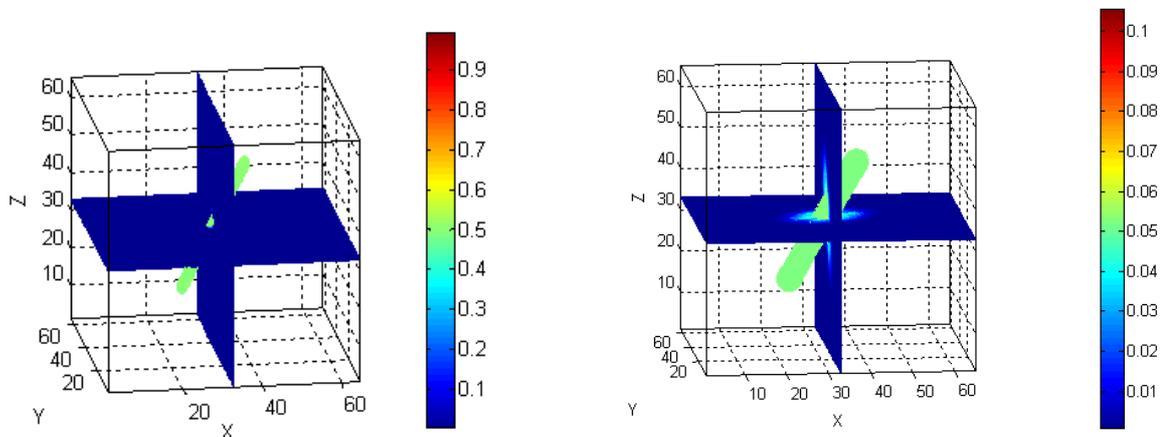




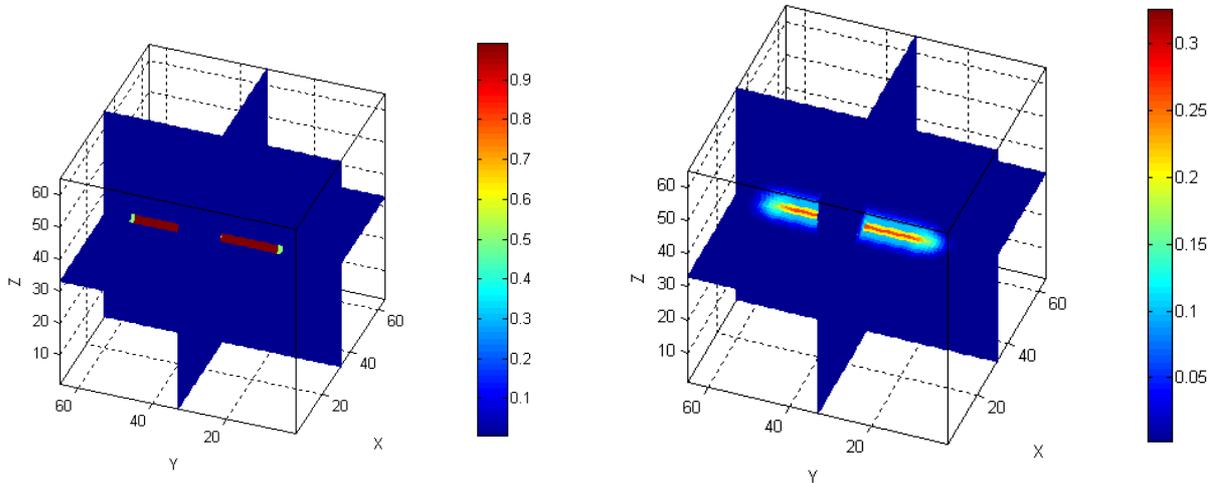
Observando los volúmenes se puede apreciar que las escalas de los originales son diferentes a las de los resultantes. Esto puede suceder por múltiples factores, pero en realidad no son importantes dado que el principal objetivo es detectar la estructura del filamento.

La librería utilizada para la visualización es el *toolbox* de Matlab conocido como *volumen_browser*. Las representaciones mostradas en primer lugar son cortes de los volumen para valores de $(x = 33, y = 33, z = 33)$. Posteriormente se muestran dos volúmenes con una representación llamada *Iso-display* en la que se reconstruyen los valores de energía máximos del filamento. En algunos casos ambas formas se pueden combinar para una mejor visualización de los resultados. A continuación se mostrarán volúmenes con estructuras en diferentes orientaciones y con diferentes parámetros.

1.- Para un volumen simbólico que posee una estructura tubular orientada en la dirección $(x = 1, y = 1, z = 1)$. con un parámetro $\sigma = 3$ y un número de muestras de $\text{deltaAng} = 20$



2.- Para un volumen simbólico que posee una estructura tubular orientada en la dirección del eje y, con un parámetro $\sigma = 1$ y un número de muestras de $\text{deltaAng} = 20$.

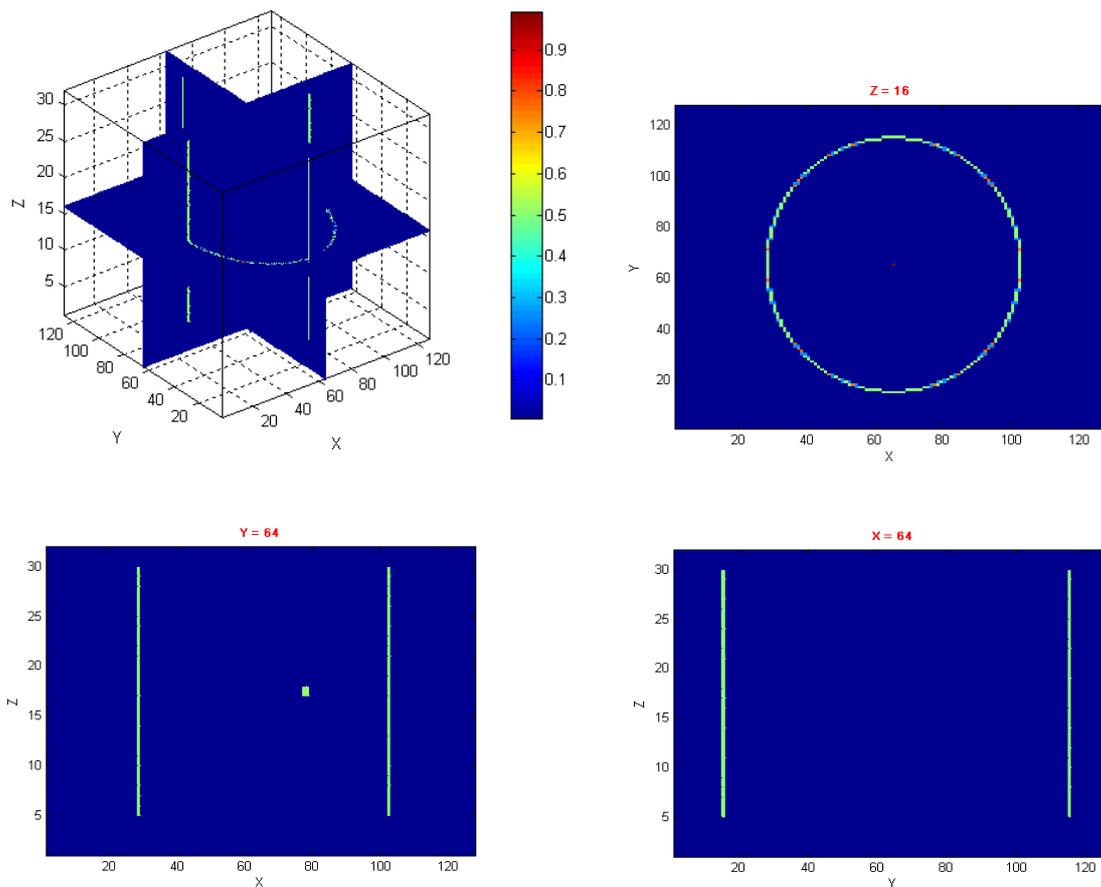


Como pueden observarse en todos los resultados el filtro detecta correctamente las estructuras analizadas. Sin embargo, es importante observar que la variable σ ha tenido gran importancia en los mismos. Si nos fijamos en el primer ejemplo, el valor de σ era tres, lo que presentaba un grosor en la estructura del filtro superior a la del filamento. Esto supuso que en la representación *Iso-displays* fuera más gruesa. Esto no implica un error del filtro sino que el parámetro escogido para el filtrado no ha sido el más apropiado. En las secciones posteriores se hablará de un método para optimizar este suceso seleccionando varias sigmas a la hora de filtrar.

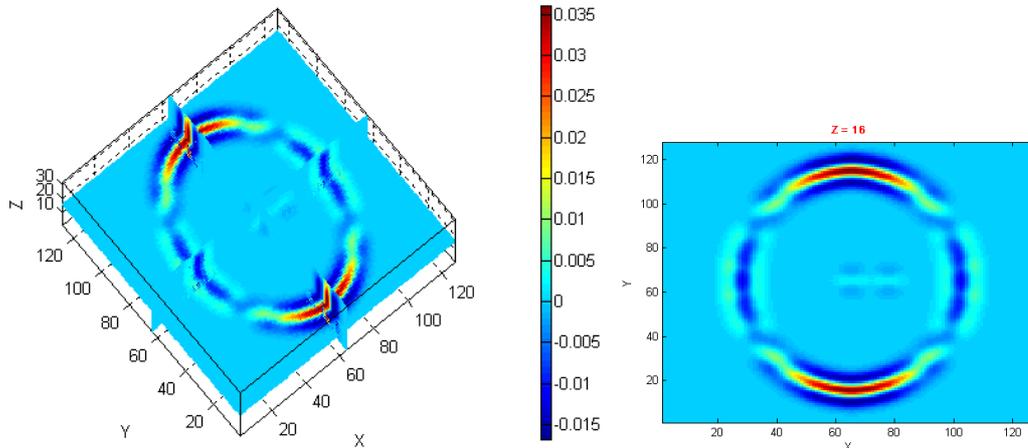
La variación del parámetro de muestreo en este tipo de volúmenes simbólicos no supone una gran importancia dado que poseen estructuras muy simples y no se aprecia la diferencia entre valores próximos. En la sección de 8 se analizarán en más detalle cuales son los valores adecuados para realizar el filtrado dependiendo de la relación tiempo/calidad.

7.1.1.2. Filtro de estructura laminar

Al igual que en el anterior filtro el proceso de obtención de los resultados es similar. La ilustración que se muestra a continuación es la representación del volumen simbólico de la estructura laminar. Los cortes del volumen están realizados para valores de $x = 64$, $y = 64$, y $z = 16$.

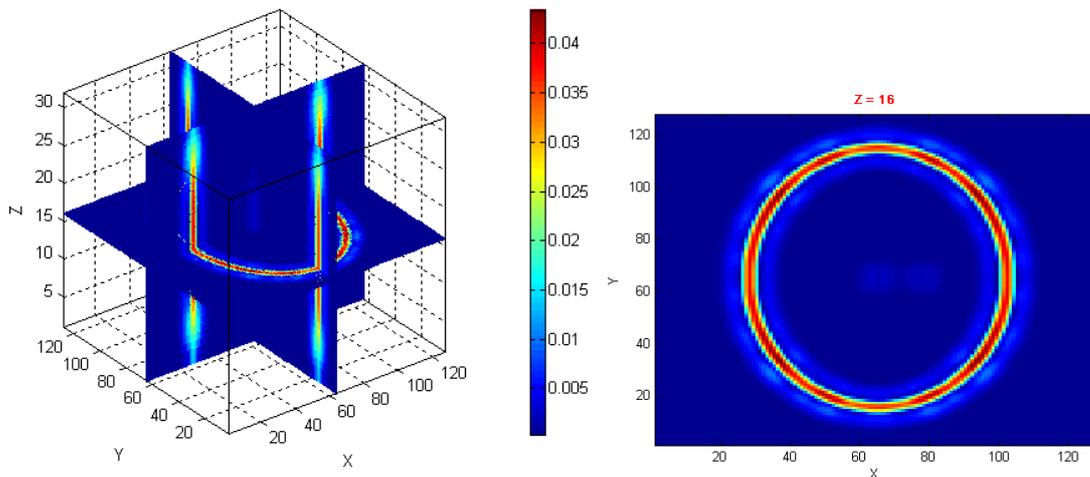


La orientación del filtrado, como pasaba en el caso anterior, es muy importante para la obtención de la estructura completa. Si el sistema se realiza en una sola dirección los resultados no son completos. El valor del parámetro σ representa el grosor de la parte de detección del filtro, lo que supone que para valores elevados el filtro será capaz de obtener estructuras gruesas, mientras que para valores pequeños lo hará detectará las finas. A continuación se mostrará cual es el efecto de el filtrado de este sistema en una sola dirección para verificar que es necesario más de una orientación para reconocer la estructura.



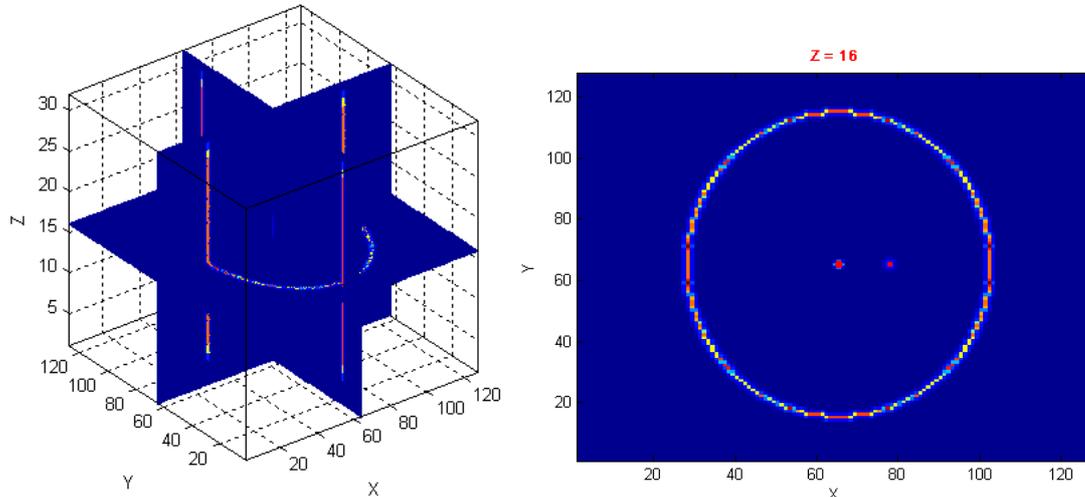
Como puede observarse sólo parte de la estructura de la membrana ha sido detectada. Esto es así porque el filtro se ha colocado en la orientación $v(0,0,1)$ lo que produce que se detecten únicamente las estructuras paralelas al eje z y que a la vez sean perpendiculares a los ejes x e y.

El resultado final del filtrado es el que se muestra a continuación.



Como se puede ver en la ilustración la obtención de la membrana ha sido correcta. Esto supone que el análisis matemático realizado en las secciones anteriores y la obtención de los coeficientes para el filtro laminar han sido apropiados. Si se analiza en detalle el corte del volumen se puede visualizar que no existen las dos partículas que quedaban anteriormente reflejadas en el volumen original. Esto no significa que los resultados sean erróneos sino que el parámetro σ no era el apropiado para todas las estructuras que estaban presentes en el volumen. Dado que el valor de $\sigma = 3$ era

demasiado grande para dichas partículas, a continuación se expondrá el resultado para un valor de $\sigma = 0.5$.

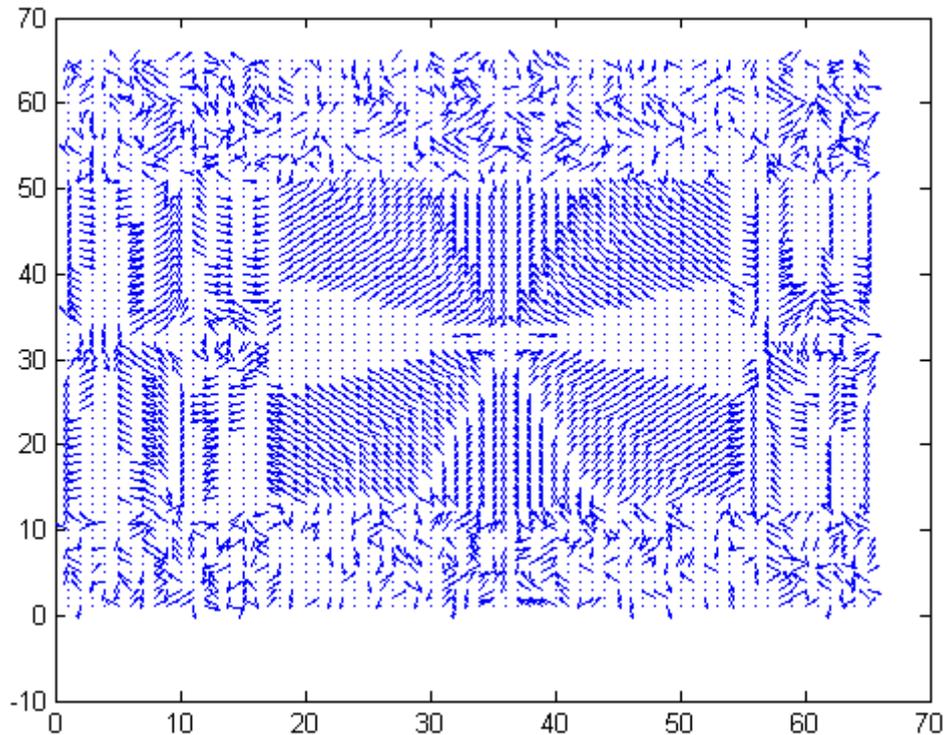


Como se ilustra en el anterior corte, las partículas sí que han quedado reflejadas en este resultado mientras que los valores del contorno del cilindro exterior han quedado sesgados. La pregunta ahora es cómo pueden obtenerse las mejores condiciones de filtrado para volúmenes que aún presentando las mismas estructuras no presentan los mismos tamaños o grosores. De este modo, para la obtención de resultados de mejor calidad, el estudio de los volúmenes reales se va a realizar con diferentes sigmas y posteriormente se realizará un estudio combinado de los resultados.

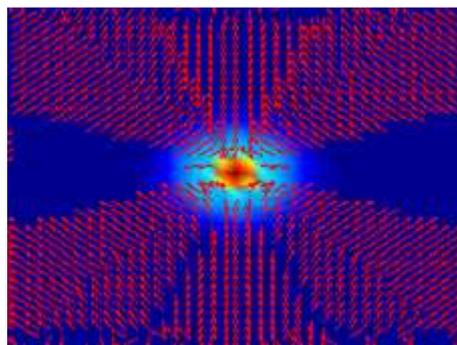
7.1.1.3. La orientación del volumen resultante

La obtención de la ubicación de la energía máxima del volumen en los procesos de filtrado es el resultado final que determina la detección de la estructura analizada. Por otra parte, también es importante saber cuáles han sido las direcciones que ha tomado el filtro para conseguir dicha energía. Para determinar esta información se guardan para cada punto del volumen las coordenadas del vector \vec{v} que ha obtenido la máxima energía para ese punto. De este modo, al final de cada proceso de filtrado, se genera un volumen paralelo, de igual tamaño al volumen entrante, que posee los vectores \vec{v} del filtrado final.

A continuación se mostrarán los resultados para un proceso de filtrado completo para cada uno de los filtros.



La ilustración anterior muestra un conjunto de pequeñas flechas que indican las direcciones de los vectores \vec{v} para cada uno de los puntos de un volumen con estructura tubular orientada a lo largo del eje z. La imagen es la sección del volumen para $z = 33$, por lo que las zonas punteadas que presenta la imagen representan vectores que han tomado las direcciones del eje z. Para simplificar un poco su visualización se va a superponer la imagen anterior con la misma sección del resultado final del filtro.



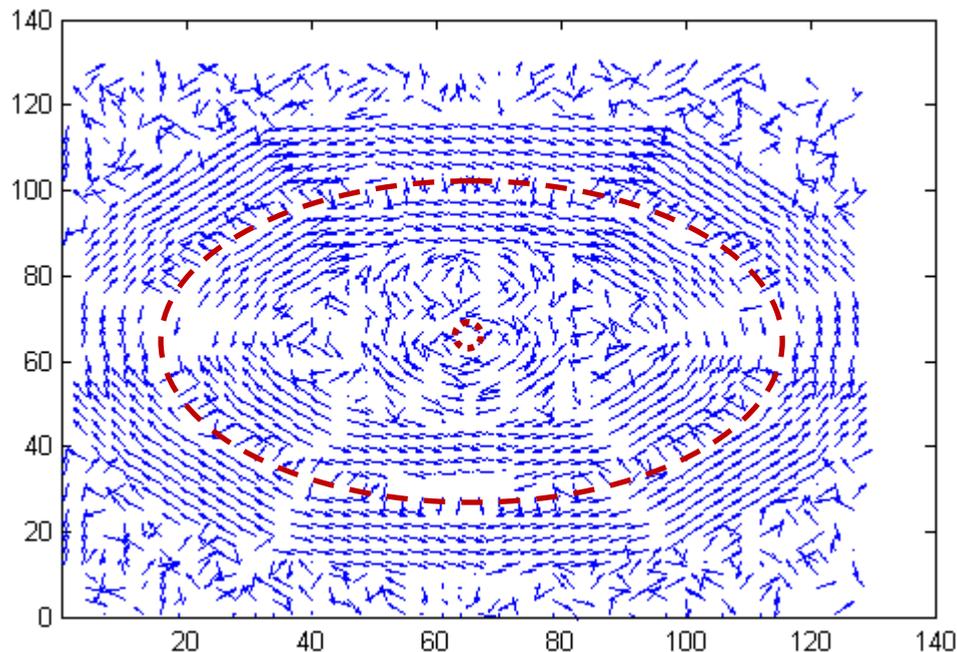
Como puede observarse la zona de máxima energía está rodeada de vectores que presentan direcciones aproximadamente iguales a la estructura del volumen analizado.

PROYECTO FINAL DE CARRERA

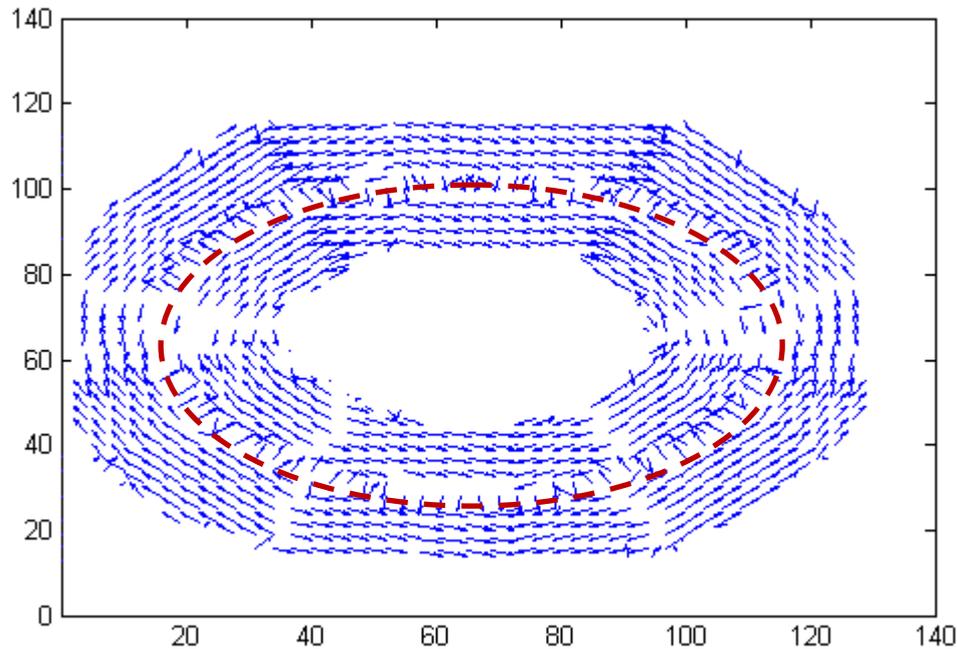
Dado que el sistema filtrado es un volumen simbólico las partes externas han sido retiradas. Esto es así porque los valores de estas zonas son nulos, por lo que no aportaban ninguna información.

A continuación se mostrará el mismo análisis para el filtro de estructura laminar. En este caso la sección del volumen se ha hecho en la misma dirección pero con un valor de $z = 16$.

En este caso la representación de las flechas es mucho más intuitiva debido a que prácticamente crean una configuración similar a la forma de la estructura analizada.



Si eliminamos, del mismo modo que en el ejemplo anterior, las flechas que no aportan ninguna información en la detección de la estructura, la imagen queda de la mucho más clara. En este caso las flechas son prácticamente perpendiculares al eje z . En el centro de la imagen también se puede observar que existen un pequeño grupo de flechas entorno a la componente laminar más pequeña del volumen. Para limpiar un poco la información, a continuación se va a mostrar la imagen resultante sin las flechas, para que nos muestren la orientación de la estructura principal.

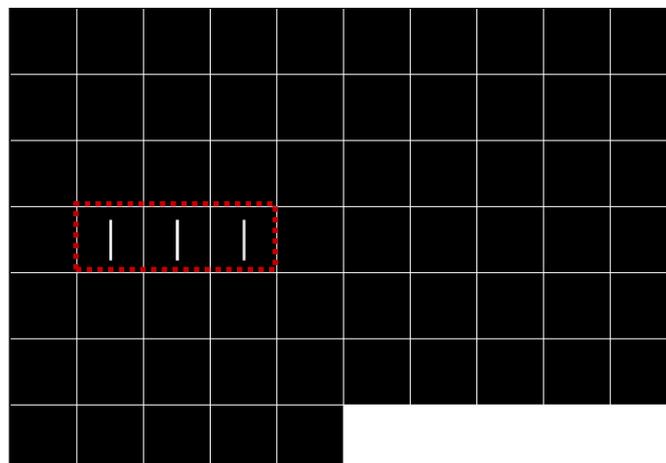


Con la se ha mostrado en los datos de esta sección este tipo de análisis aporta nueva información que puede ser relevante a la hora de analizar estructuras con configuraciones complejas o de difícil visualización.

7.1.2. C++ con Xmipp

7.1.2.1. Filtro de estructura tubular

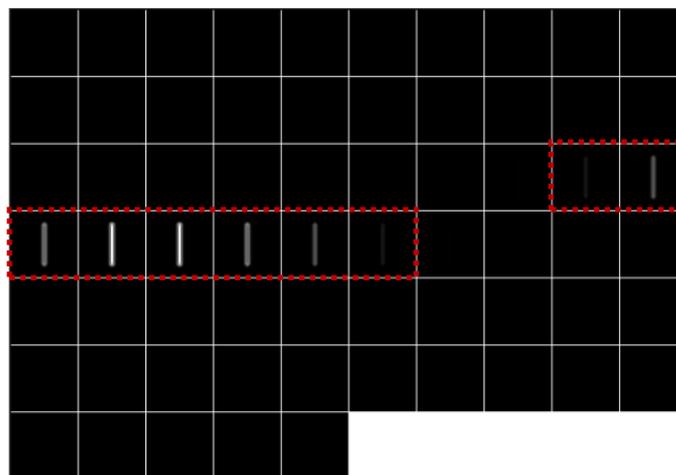
A partir de ahora se va a desarrollar el análisis de los volúmenes simbólicos utilizando la programación en C++ con las librerías de Xmipp. Este segundo estudio no sólo se realiza para verificar si la implementación del sistema ha sido la correcta, sino para adquirir una nueva visión de los resultados.



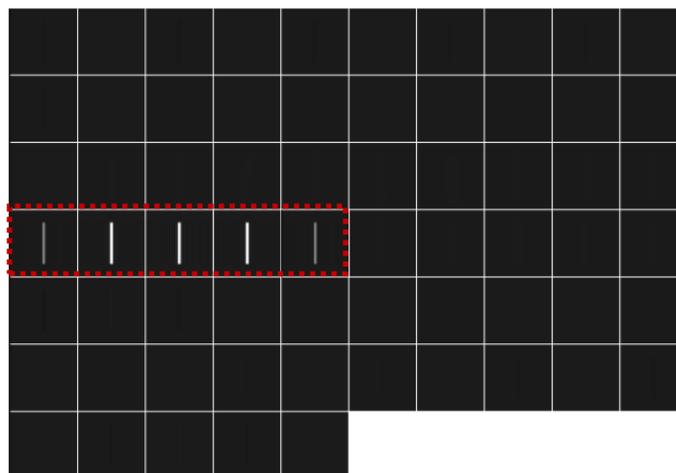
PROYECTO FINAL DE CARRERA

La imagen anterior es la visualización mediante la librería Xmipp de un volumen simbólico. La representación de este sistema se basa en cortes del volumen para las orientaciones de los ejes. En este caso las secciones han sido en la dirección del eje x . Cada uno de los cuadros que se muestran son los valores para $x = 1, 2, 3, 4, \dots$. De este modo para los valores de $x = 32$, $x = 33$ y $x = 34$ existe una estructura tubular en la dirección del eje z .

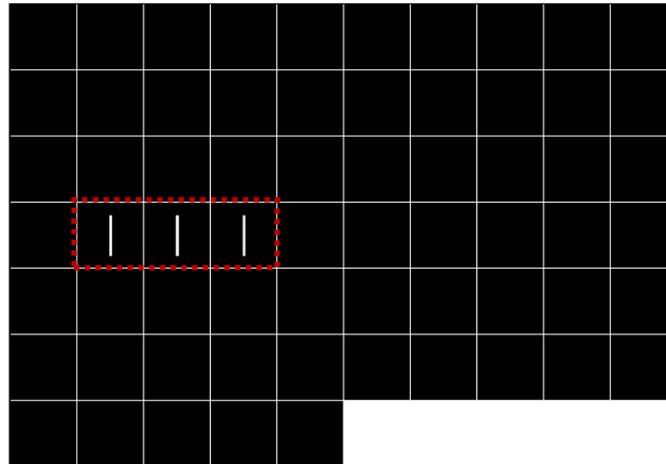
El resultado para una $\sigma = 2$ con una calidad de $\text{deltaAng} = 20$ es el siguiente.



Como muestran las secciones, la detección ha sido correcta. Por otra parte, el valor de σ es mayor que el de la estructura analizada. Si realizamos otra vez el filtrado para un valor de $\sigma = 1$ obtendríamos el resultado siguiente.



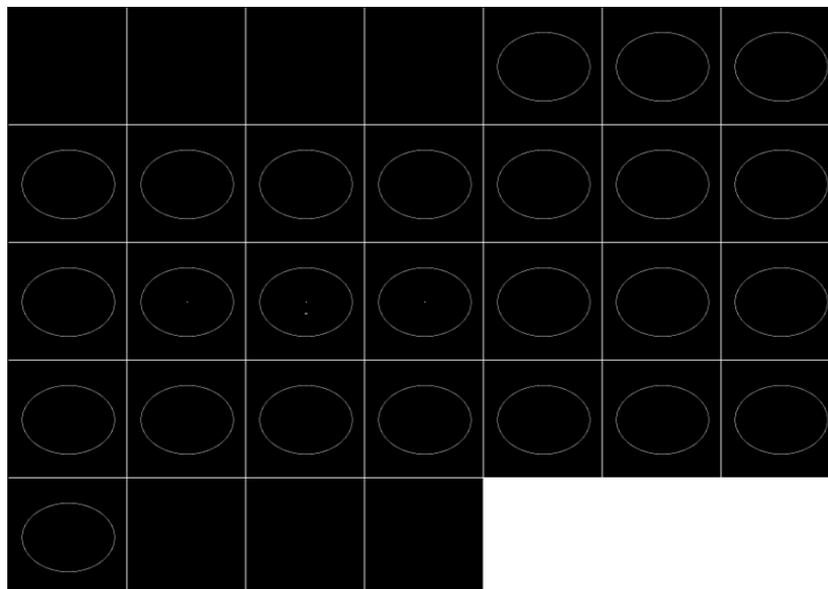
Como puede verse en este último caso los resultados han sido mejores. De este modo, si se realiza el filtrado para un valor de sigma menor, el resultado es el siguiente.



Sin lugar a dudas el volumen resultante presenta la misma estructura tubular que se presentaba en el volumen inicial. Por lo que se puede concluir que la implementación en C++ es también correcta.

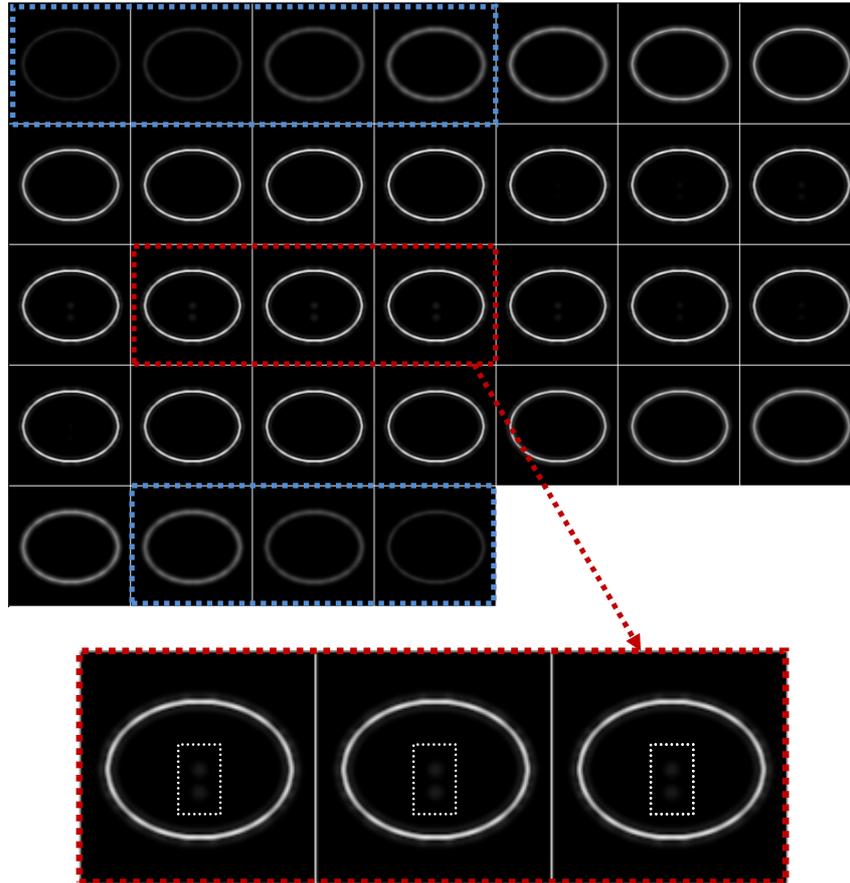
7.1.2.2. Filtro de estructura Laminar

La imagen que se presenta a continuación es la representación del volumen simbólico utilizado para la verificación de los procesos de filtrado laminar.

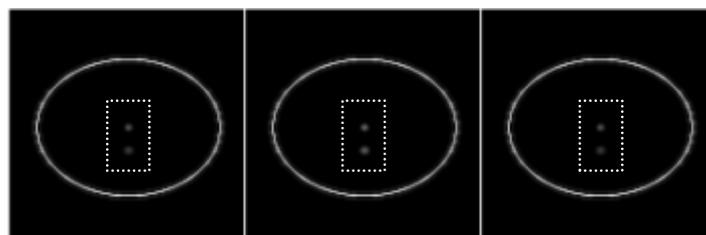


Como se comentaba en secciones anteriores este volumen presenta dos partículas situadas en el interior del cilindro que simula la pared celular. En este caso hay que fijarse bien en los cortes de $x = 15, 16$ y 17 para poder percatarse de ellas.

Si filtramos dicho volumen para un valor de $\sigma = 2$ el resultado es el que se muestra a continuación.

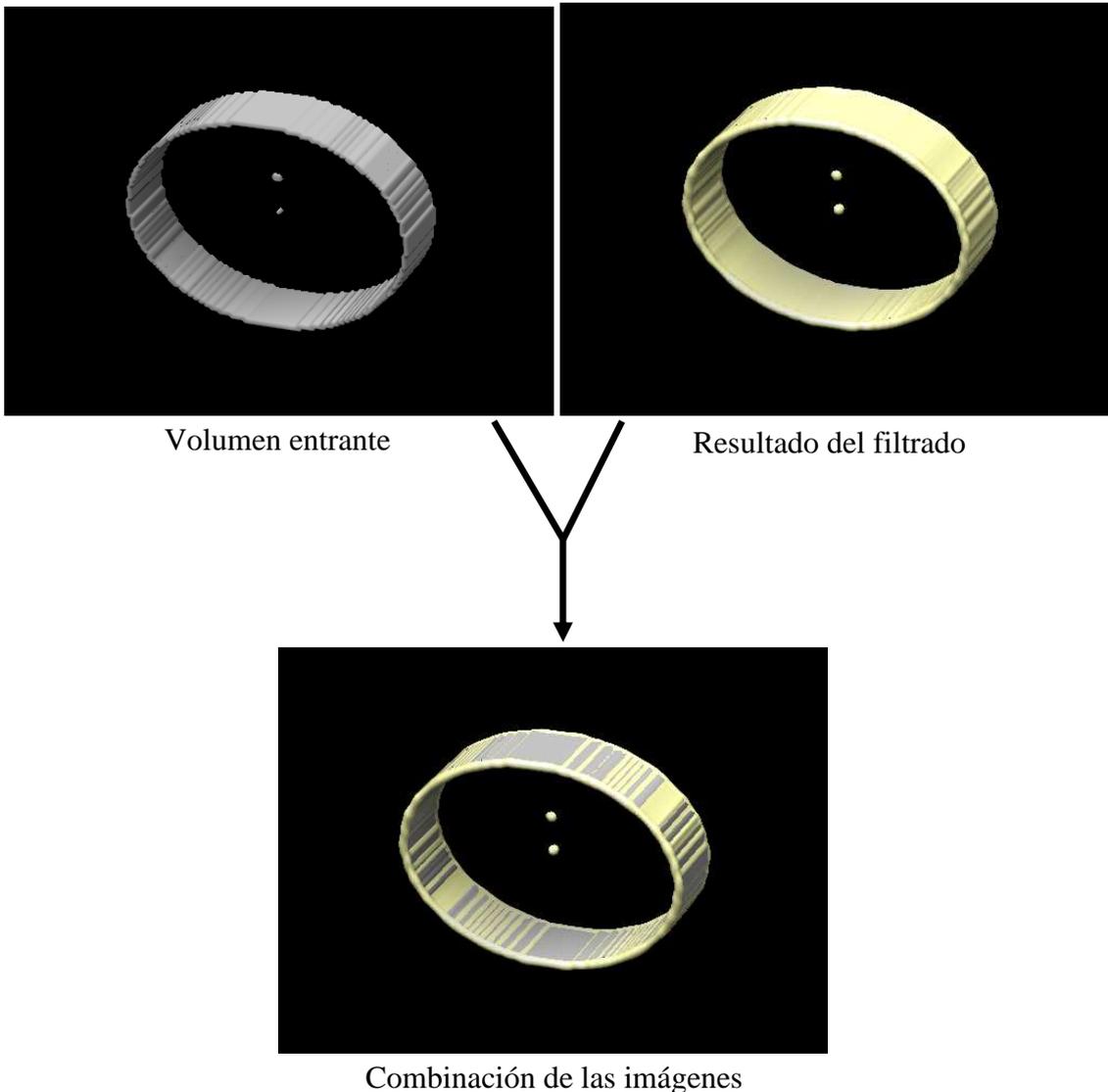


Para los valores centrales se ha hecho una ampliación para demostrar que no se detectan las partículas interiores para valores tan grandes de sigma. Por otra parte se muestran, en recuadros azules, ciertas representaciones de la estructura cilíndrica que no aparecen en la imagen inicial. Estas cortes no representan un error en la detección dado que la estructura analizada ha sido filtrada por una única sigma que posiblemente es mayor al valor apropiado. A continuación se mostrarán los resultados del volumen resultante del proceso de filtrado para una sigma de valor 1 para verificar que se detectan las estructuras interiores.

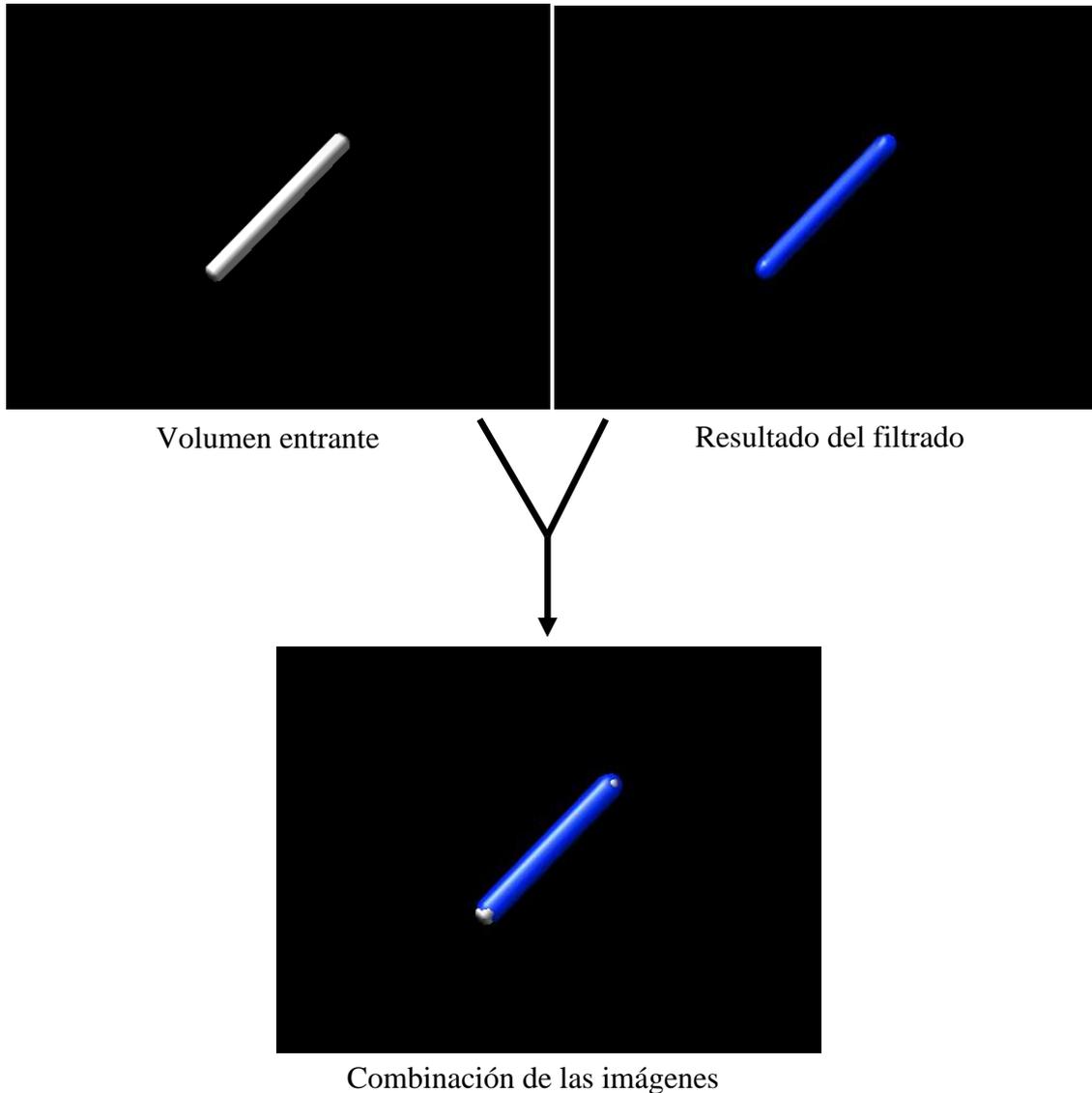


7.1.3. Visualización con Chimera

A continuación se muestra la reconstrucción de los resultados en Chimera para los sistemas laminares. Todas las imágenes presentan en el cilindro exterior una forma escalonada en vez de regular. Este suceso es debido a que el programa Chimera no reconstruye de forma precisa cambios bruscos para estructuras finas. El volumen final es resultado del filtrado para una $\sigma = 1$ y una $deltaAng = 20$. Como puede verse la combinación de las imágenes muestra la gran similitud existente entre el volumen entrante y el resultado del filtrado. Por otra parte es importante observar que las partículas que se encontraban en el interior del cilindro han sido reconocidas durante el proceso de filtrado.



A continuación se mostrará de manera análoga los resultados para el filtrado de estructuras tubulares.



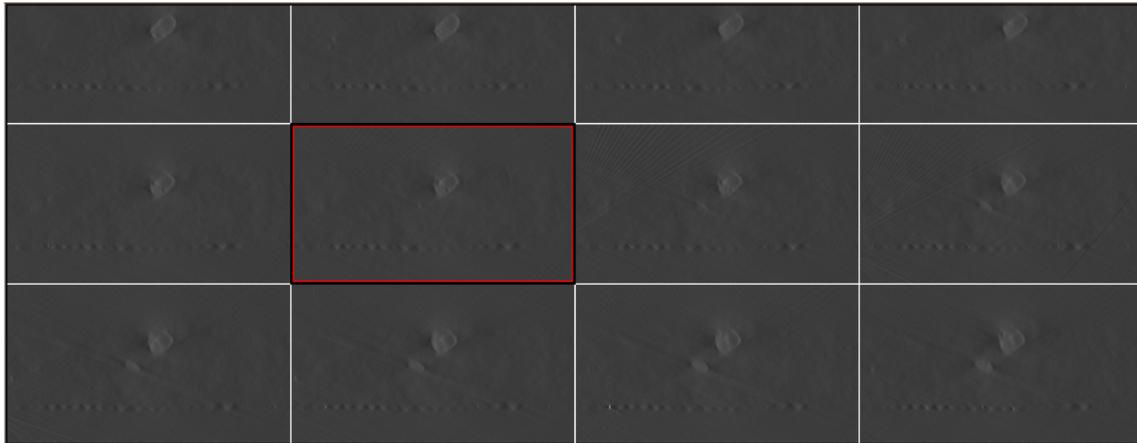
7.2. Análisis de los volúmenes reales

7.2.1. Análisis del filtrado tubular

En esta sección se mostrarán los resultados definitivos de este proyecto. Dada su importancia los volúmenes se presentarán de numerosas formas y se analizarán las características de cada una de ellas.

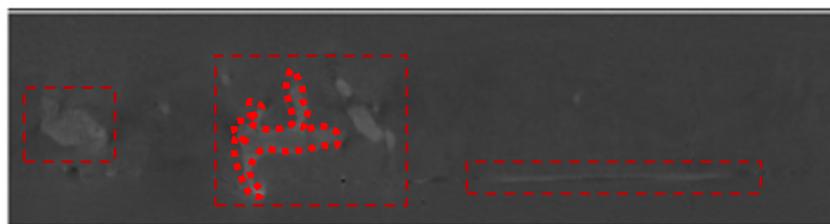
PROYECTO FINAL DE CARRERA

Para el filtro de estructura tubular se ha realizado el estudio para un conjunto de filamentos que se encuentran en un volumen biológico de tamaño 462x120x212. Como se mostrará a continuación, su visualización no es trivial mediante la librería Xmipp, por eso se recurrirá con frecuencia a la ayuda de Chimera para reconocer en tres dimensiones los resultados obtenidos.



Representación de cortes del volumen filamentoso según el eje z

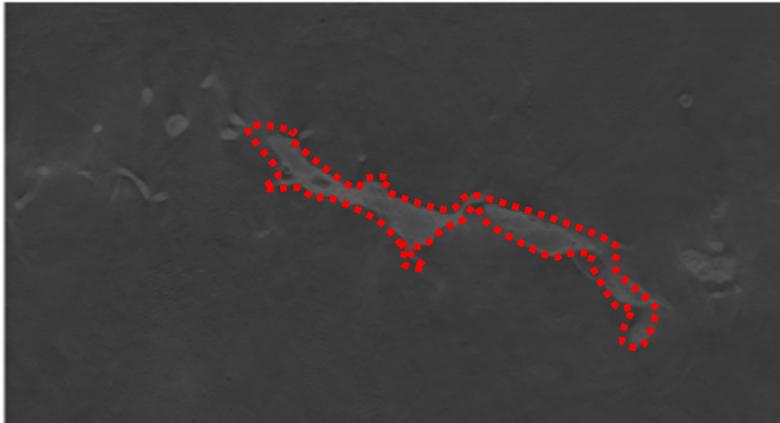
Debido al gran tamaño que presenta el volumen únicamente pueden ser mostrados algunos cortes del volumen.



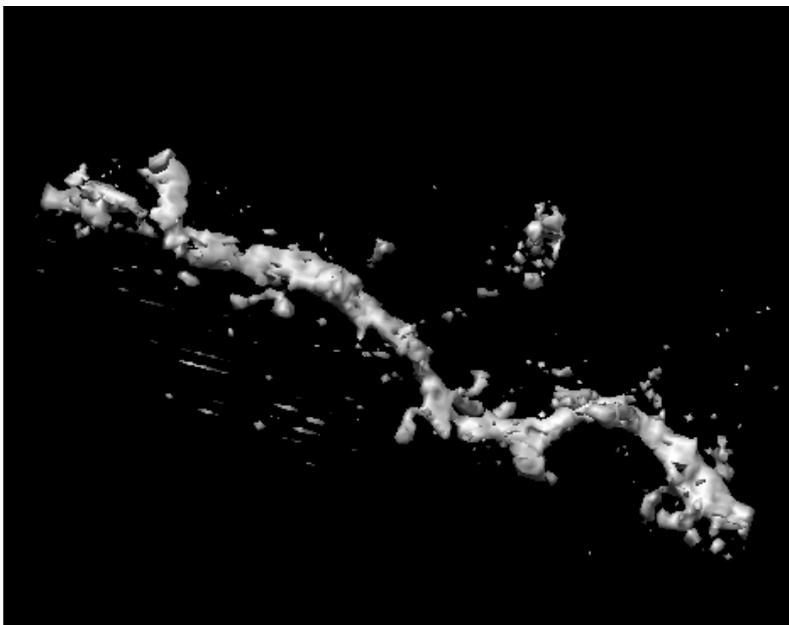
Representación de cortes del volumen filamentoso según el eje x

Aunque no se puede apreciar de manera precisa se pueden observar que los filamentos de las imágenes no tienen una estructura uniforme y que poseen diversas orientaciones en el espacio.

NOTA: Adjunto a este documento se proporciona un CD donde se proporcionan una batería de pruebas de los volúmenes filamentosos y laminares.



Representación de cortes del volumen filamentosos según el eje y



Representación en Chimera del volumen filamentosos

La visión en tres dimensiones de la estructura del filamento en Chimera es mucho más intuitiva que los cortes. La reconstrucción de está basada en un sistema de isosuperficies.

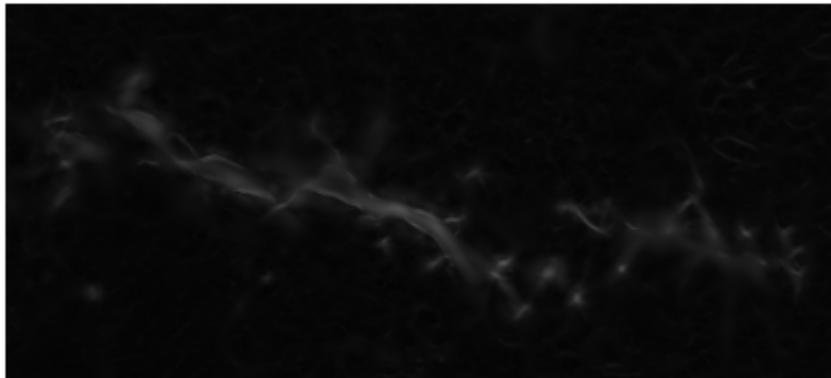
Como se ilustra en las figuras anteriores el carácter no regular de los filamentos es un impedimento para el filtrado debido a que no es posible una elección única para el valor del parámetro σ .

El problema de utilizar una única sigma es que para un volumen con varios filamentos de diferentes grosores el resultado que se obtiene no es el óptimo para todos, si no para aquellos que se aproximen más a la condición del filtro. Por esto, es interesante realizar el filtrado con varias sigmas para obtener cada uno de ellos con una

PROYECTO FINAL DE CARRERA

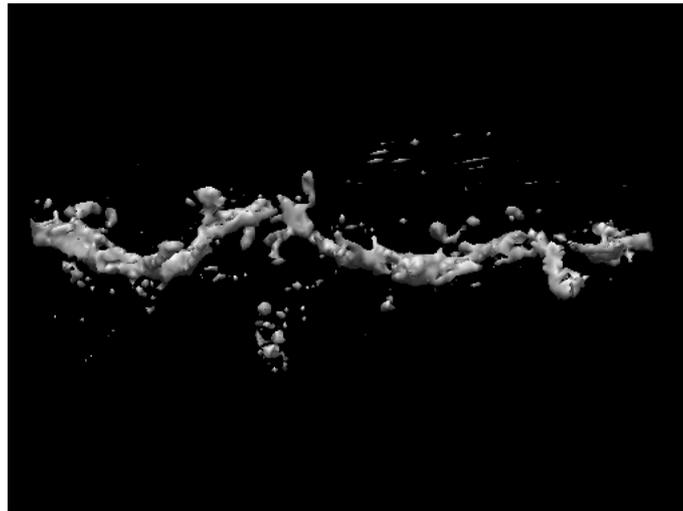
buena calidad. No obstante el hecho de que se realicen varias operaciones de filtrado implica que existe la necesidad de agrupar los resultados finales de algún modo. Para que no exista ninguna traba a la hora del escalado del volumen, es necesario normalizar las energías de los volúmenes finales antes de obtener las energías máximas del conjunto. De esta forma si los resultados obtenidos de cada una de las sigmas se expresan en tanto por ciento, la comparación de dichos porcentajes entre los diferentes volúmenes lograría la energía máxima para cada uno de los filamentos con distintas parámetros sigmas.

A continuación se mostrará el filtrado tubular del volumen filamentososo para un corte en la dirección del eje y . Los parámetros utilizados son $\sigma = 1$ y $\text{deltaAng} = 5$.

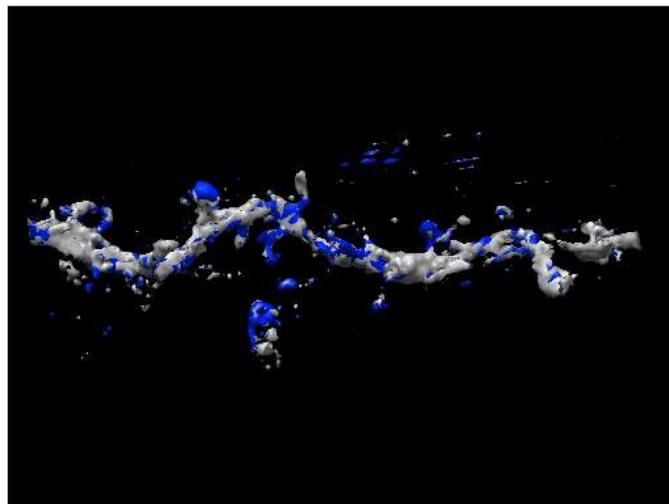


Como pueden observarse las estructuras resultantes son muy finas, esto es debido a que la sigma era muy pequeña.

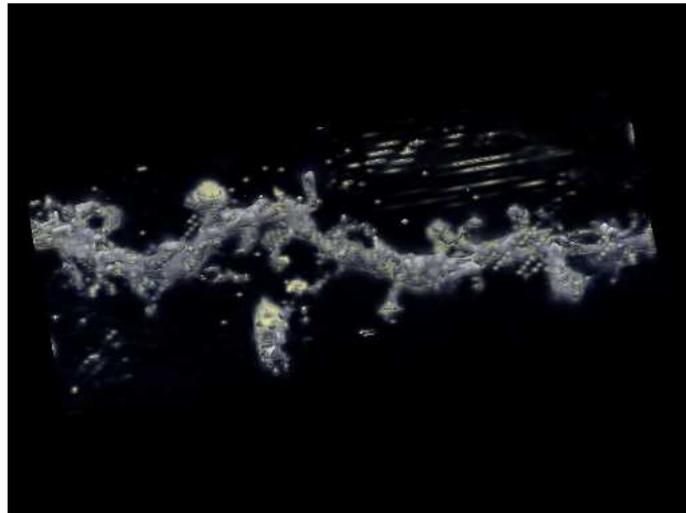
La imagen que se muestra a continuación es el volumen original para ser filtrado. Posteriormente se mostrarán dos imágenes que presentan la combinación de los volúmenes original y resultante del filtrado. Lo que se pretende con esto es poder visualizar la diferencia entre ambos y ver si el resultado del filtrado ha sido el correcto.



Como se puede observar en la siguiente fotografía existen dos colores para la representación de las estructuras. El color azul representa la estructura del volumen resultante después del filtrado y el gris representa el volumen original. En este esquema los dos volúmenes están superpuestos lo que significa que ambos tienen una estructura similar. Las componentes azules de la imagen no representan un error en el sistema de filtrado dado que la detección se ha producido en la mayor parte del volumen. Probablemente la causa de este error se deba a las irregularidades del volumen a analizar.



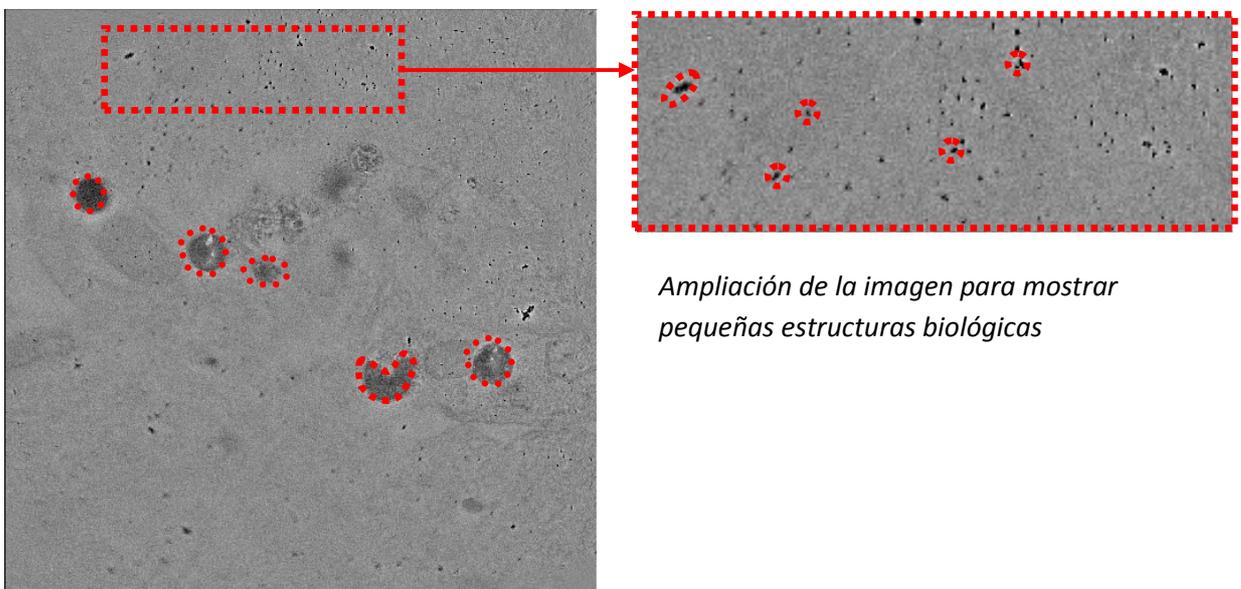
Como añadido, a continuación se muestra la misma configuración que en la anterior imagen pero representando el resultado del filtrado por encima del volumen original para verificar que la estructura resultante es similar a la analizada.



7.2.2. Análisis del filtrado laminar

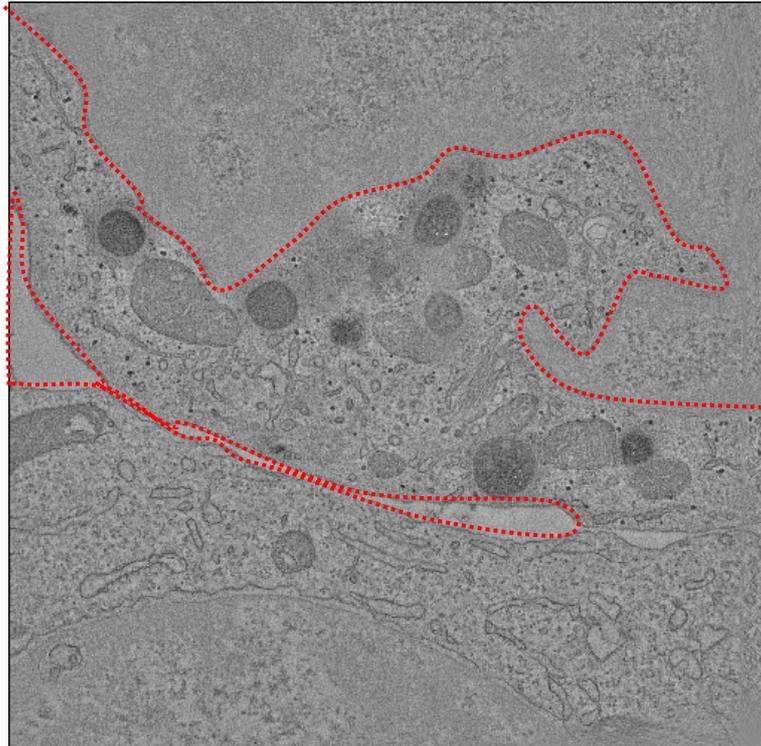
Uno de los resultados con más acierto en este proyecto han sido las adaptaciones de los volúmenes reales a los sistemas de filtrado laminar. Como se mostrará a continuación, las imágenes para este estudio muestran una gran gama de estructuras propias de configuraciones biológicas. Dichas formas se presentan comúnmente en la mayoría de las investigaciones de este campo. Además, el volumen examinado presenta un tamaño de 512x512x19 por lo que su visualización aporta mucha información para el análisis. Por todo esto, los resultados de este volumen se van a extender para poder escudriñar con todo lujo de detalles la información que nos aporta.

El volumen inicial presenta la siguiente forma para el plano $z = 1$;



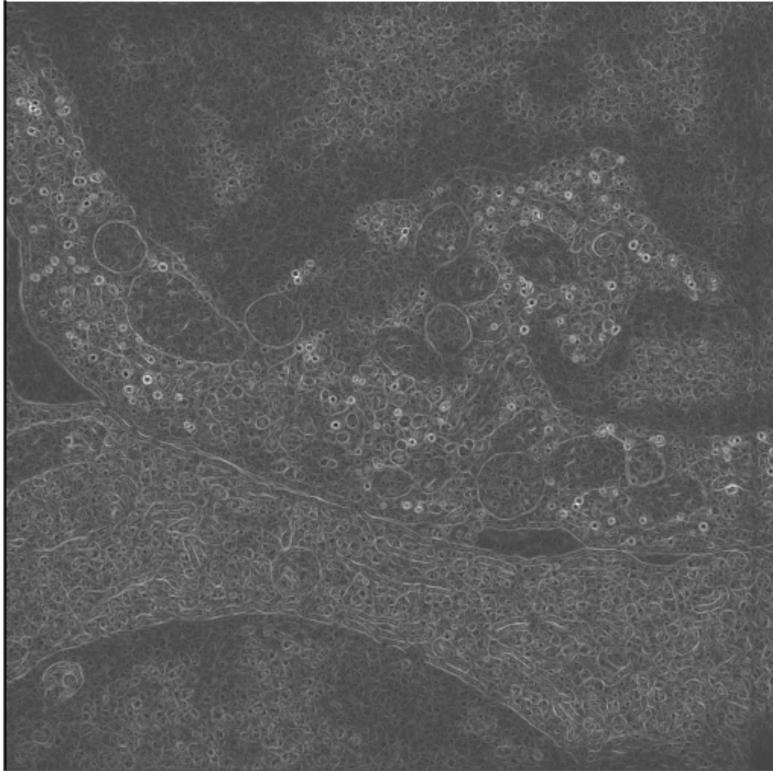
Ampliación de la imagen para mostrar pequeñas estructuras biológicas

Como se observa en la ilustración las membranas y parece celulares de la imagen presentan diferentes tamaños y formas. Algunas de ellas son visibles a primera vista mientras que otras no. Si se realiza el corte para otro valor de z (por ejemplo $z = 8$) el resultado muestra que además de los sistemas apreciados en la imagen anterior, existen muchas membranas rodeando el volumen.



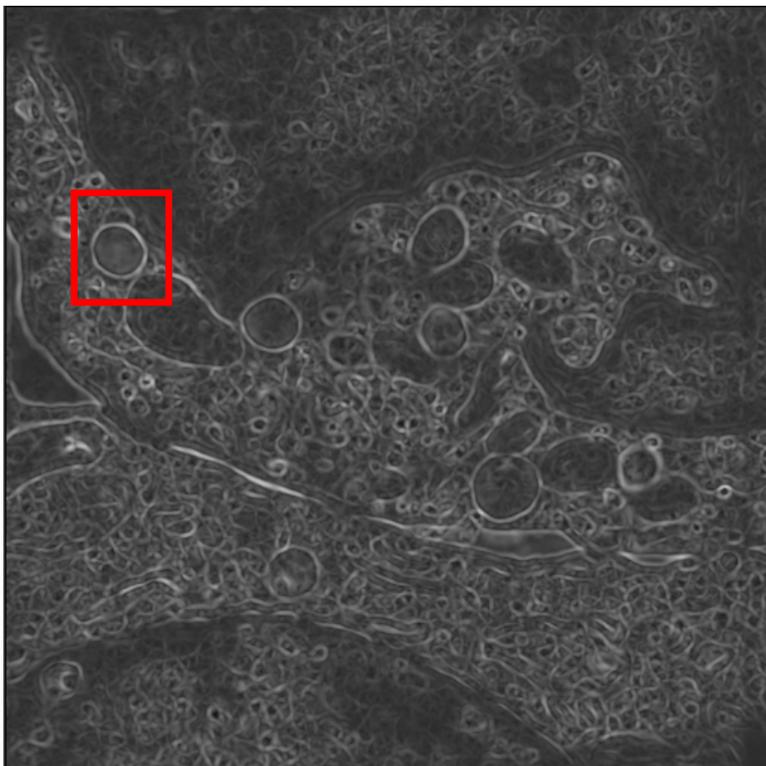
Las zonas delimitadas en rojo presentan algunas de las membranas que existen en el volumen.

Para apreciar cuales de las componentes σ que son necesarias para calcular el filtrado óptimo de volumen, se mostrarán los resultados para valores de $\sigma = 1, 2, \dots, 5$ el valor de $\text{deltaAng} = 1$ para mostrar la máxima calidad en estos resultados.



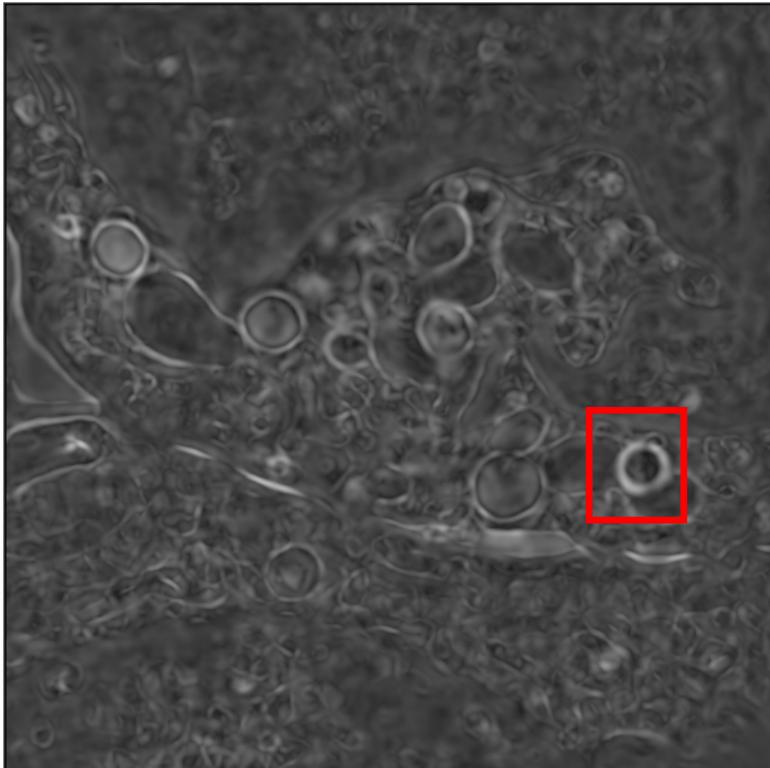
La imagen que se muestra a la izquierda es el filtrado para el valor de $\sigma = 1$.

Como puede observarse el sistema detecta con mayor detalle los filamentos y estructuras más finas, es decir, las pequeñas configuraciones que aparecen resaltadas en blanco. Como es lógico pensar en este tipo de representación las zonas de máxima energía son presentadas por las partes más próximas al blanco.



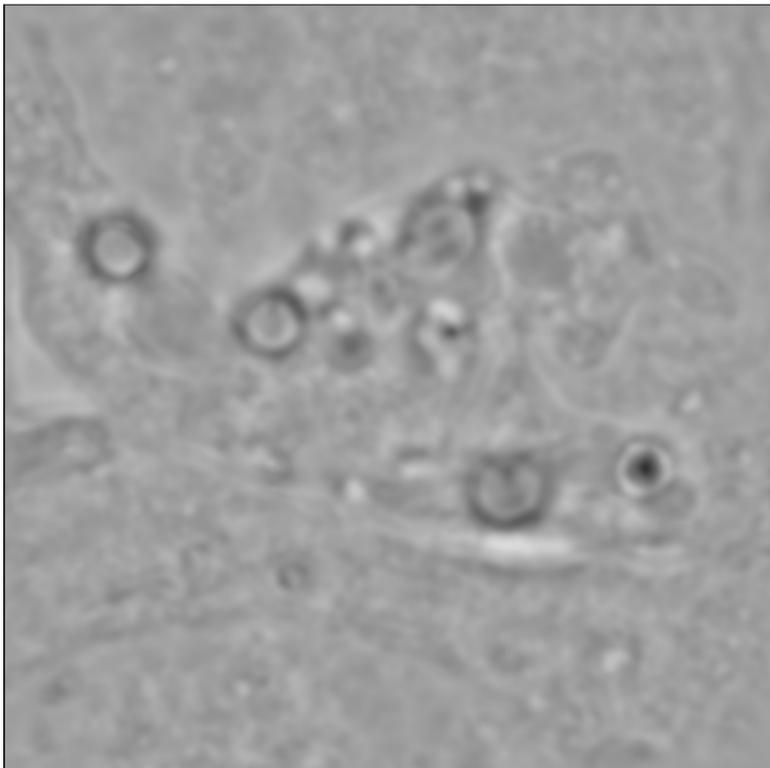
La imagen que se muestra a la izquierda es el filtrado para el valor de $\sigma = 2$.

En este caso el filtrado ya no detecta las estructuras más pequeñas pero en cambio obtiene una mayor energía en componentes mayores, como es el caso de la sección que se muestra en rojo. También es importante fijarse en que ha existido un incremento en la detección de energía en las membranas que rodean a las estructuras centrales.



La imagen que se muestra a la izquierda es el filtrado para el valor de $\sigma = 3$.

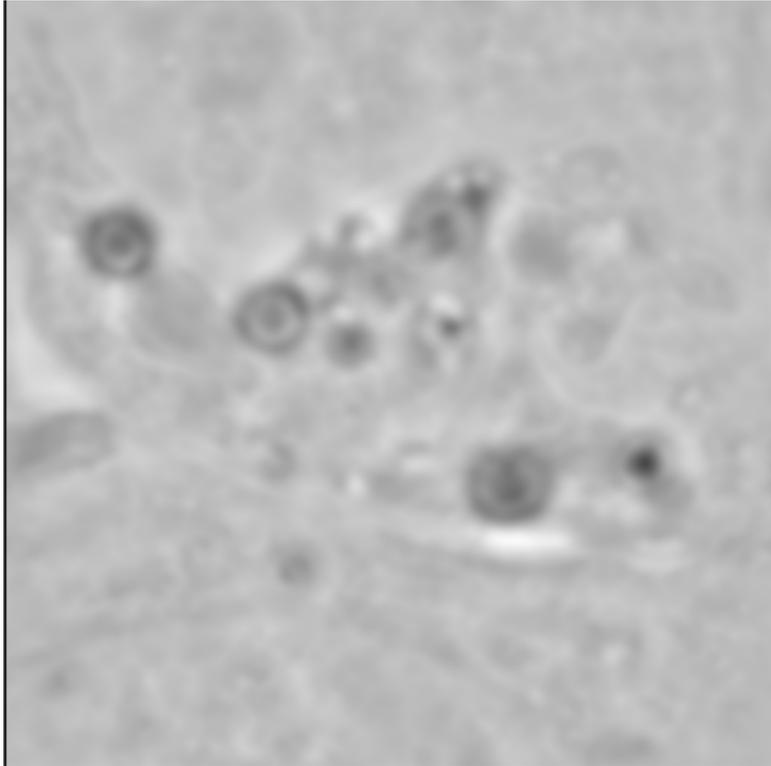
En este caso el filtrado ya no detecta ninguna estructura pequeña. También ciertas partes de la membrana han desaparecido. Por otra parte la energía para ciertas estructuras sigue aumentando, por ejemplo, la que se muestra encuadrada en rojo.



La imagen que se muestra a la izquierda es el filtrado para el valor de $\sigma = 4$.

En este proceso de filtrado no se detectan muchas de las estructuras iniciales por lo que mucha información no se ha adquirido en mediante este valor de σ .

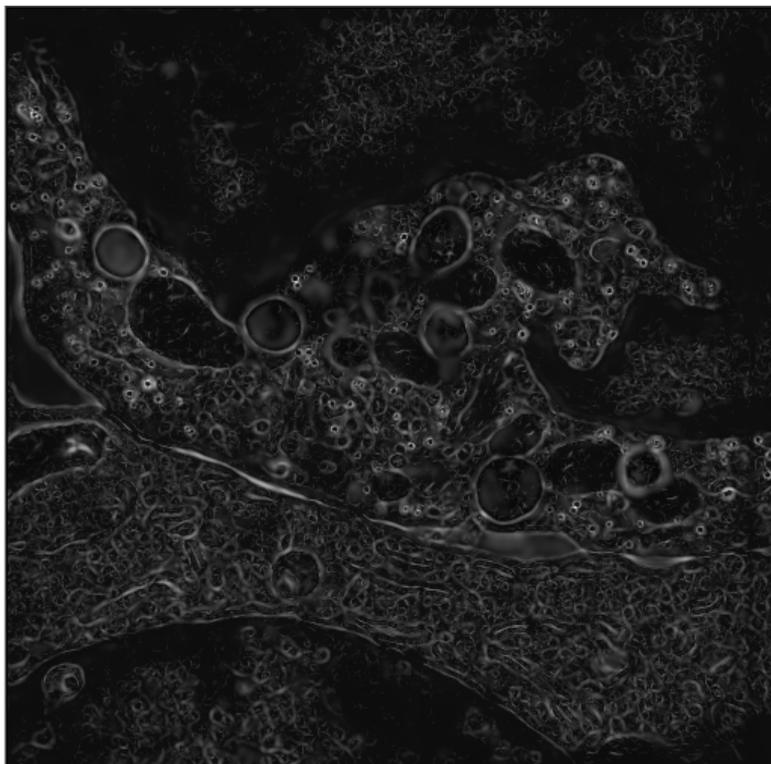
Sí que es cierto que existen zonas muy aplanadas que aún adquieren puntos de energía interesantes como en las regiones de la parte superior.



La imagen que se muestra a la izquierda es el filtrado para el valor de $\sigma = 5$.

En este proceso de filtrado la información que se obtiene es prácticamente nula con respecto a los anteriores filtrados por lo que se puede decir que para este volumen el grosor máximo de detección de filtrado es $\sigma = 5$.

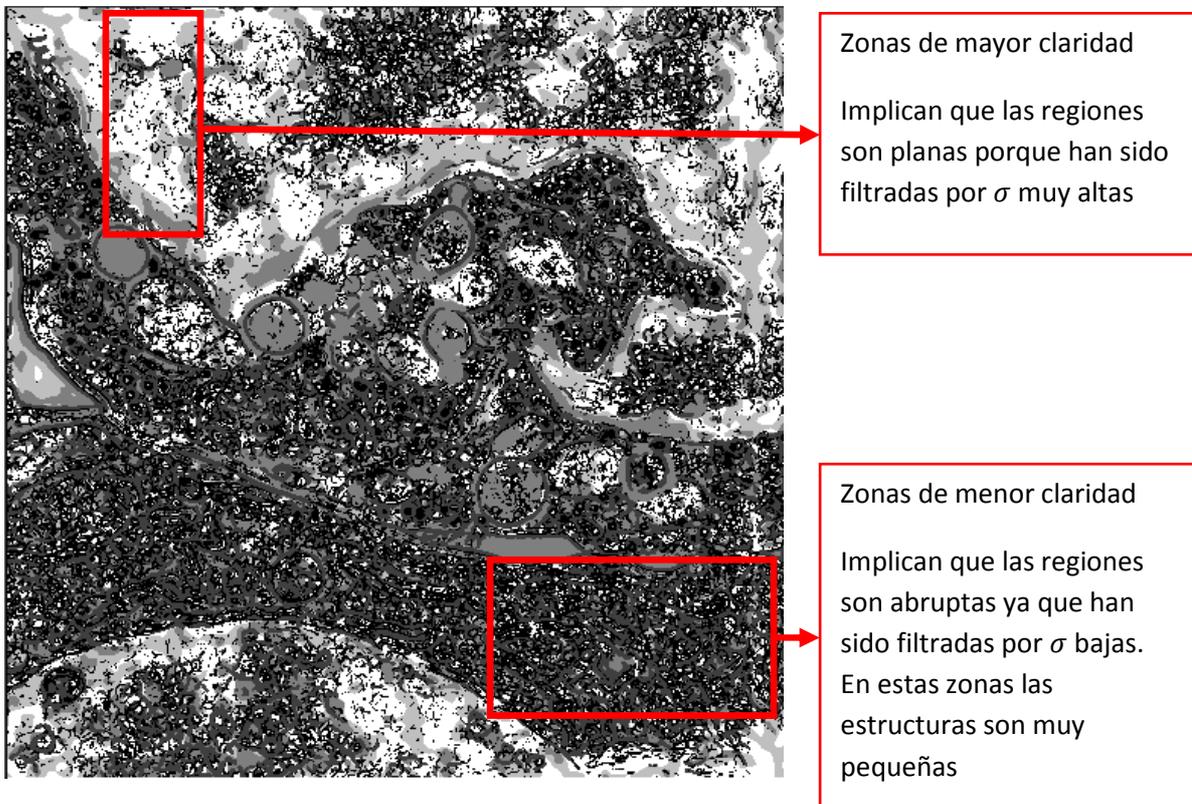
A continuación se van a mostrar los resultados del conjunto de las energías máximas obtenidas en los procesos anteriores (esta técnica se explico en detalle en el apartado 7.1.2.).



Esta imagen muestra todas las estructuras que se han ido definiendo desde $\sigma = 1$ hasta $\sigma = 5$.

De este modo los filtros implementados pueden obtener el análisis de cualquier volumen que presente configuraciones de diferentes tamaños siempre y cuando sean capaces de definir los umbrales de grosor para el valor de σ .

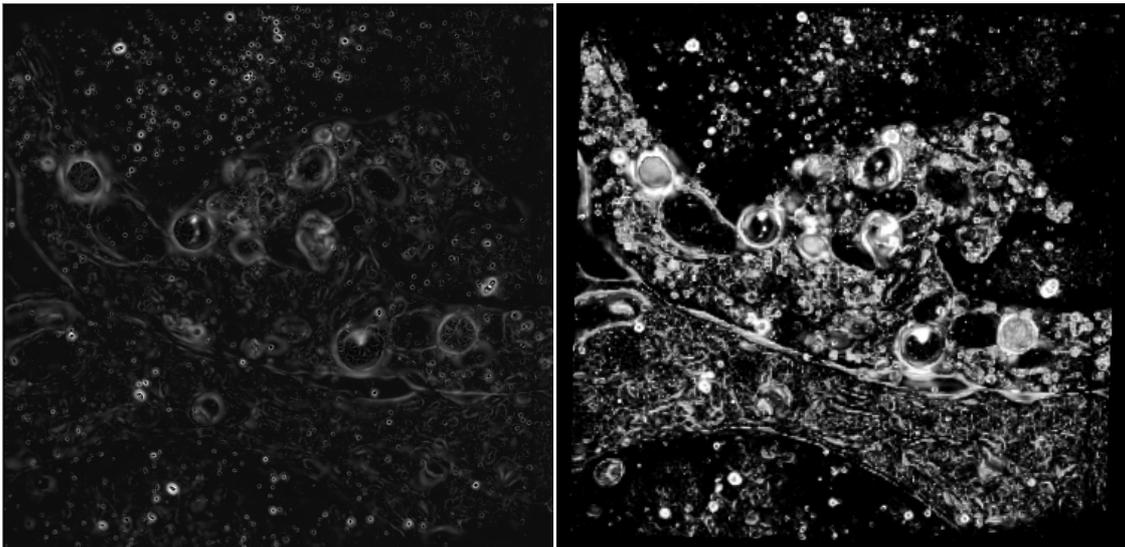
Realizado el proceso de combinación de las energías dependiendo de sigma, la duda que puede surgir en este punto es cuáles son los valores de sigma para cada uno de los puntos del volumen obtenido. Para ello se ha reconstruido una especie de histograma en tres dimensiones que muestra estos valores para cada coordenada del volumen. La sección que se muestra a continuación es un ejemplo de ello. Las regiones de la imagen que se poseen mayor claridad hacen referencia a valores de sigma altos, mientras que las oscuras lo hacen para los bajos.



Este sistema de representación no solo es útil para saber cuál es la mayor proporción de energía adquirida para un valor de sigma, sino para ver la evolución de las estructuras a lo largo de sus secciones. De este modo se puede observar cómo van creciendo las membranas en grosor y a la velocidad a lo que lo hacen a lo largo del volumen. Esta propiedad no es fácilmente visible mediante las ilustraciones que se muestran en este documento dado que para ver la eficacia del sistema hay que ir analizando la evolución de las estructuras en muchos cortes del volumen.

7.2.3. Visualización de los resultados en Chimera

La diferencia en la visualización de los resultados con el programa Chimera y con el Xmipp no son tan grandes como en anteriores casos. Esto es debido a los limitados cortes que pueden existir en la dirección del eje z para un volumen de 512x512x19. Sin embargo la visualización con Chimera permite ver a la vez un conjunto de estructuras que se presentan en diferentes planos.



Sección $z=1$ del volumen resultante representado en Xmipp

Representación con Chimera del volumen resultante mediante esquemas sólidos

Si comparamos las dos imágenes anteriores, la reconstrucción en Chimera presenta tanto la presencia de las pequeñas estructuras en la parte superior de la imagen (plano $z = 1$), como las membranas que se encuadran en la parte inferior (planos $z \geq 5$). De esta manera en la imagen de la derecha se puede ver toda la detección de las estructuras laminares del volumen analizado.

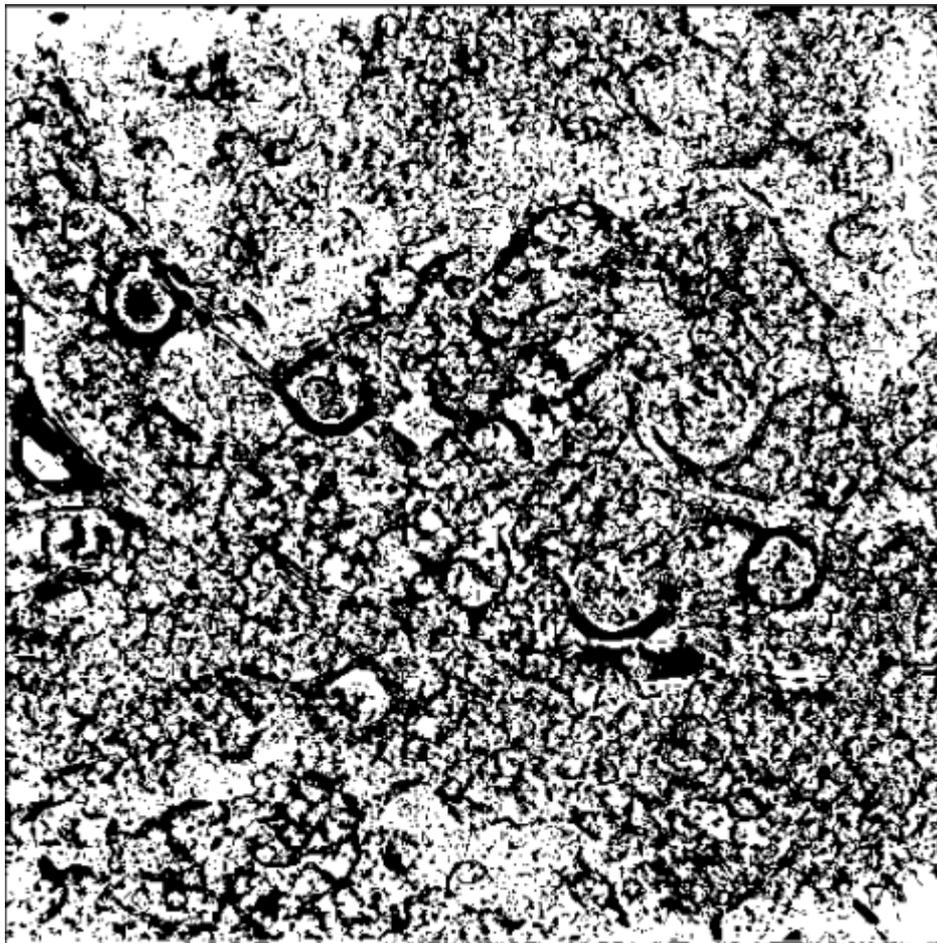
7.2.4. Análisis del filtrado múltiple para el volumen laminar real

Una de las acciones que todavía no se ha examinado a lo largo de este documento es utilizar los dos filtros implementados para analizar el mismo volumen. Esta prueba es importante dado que un volumen biológico no presenta únicamente

PROYECTO FINAL DE CARRERA

estructuras de un tipo, sino una gran gama de ellas. De este modo, si filtramos el volumen utilizado en la sección anterior con el filtro de estructura tubular, en vez del de laminar, obtendremos un análisis diferente del volumen. En este caso, si comparamos los volúmenes resultantes de cada uno de los procesos de filtrado y examinamos punto por punto cual de los dos ha obtenido mejores resultados, podemos crear un esquema que ilustre donde ha sido mejor la detección de energía para cada uno de los filtros.

A continuación se muestra dicho esquema para una sección del volumen.



Las zonas en color negro son las partes que han sido detectadas mejor por los procesos de filtrado tubular. Por el contrario las zonas de color blanco representan la detección del filtro laminar. Como se puede ver, es difícil determinar si un filtro es mejor que otro a la hora de la detección.

8. Estudio de tiempos de ejecución

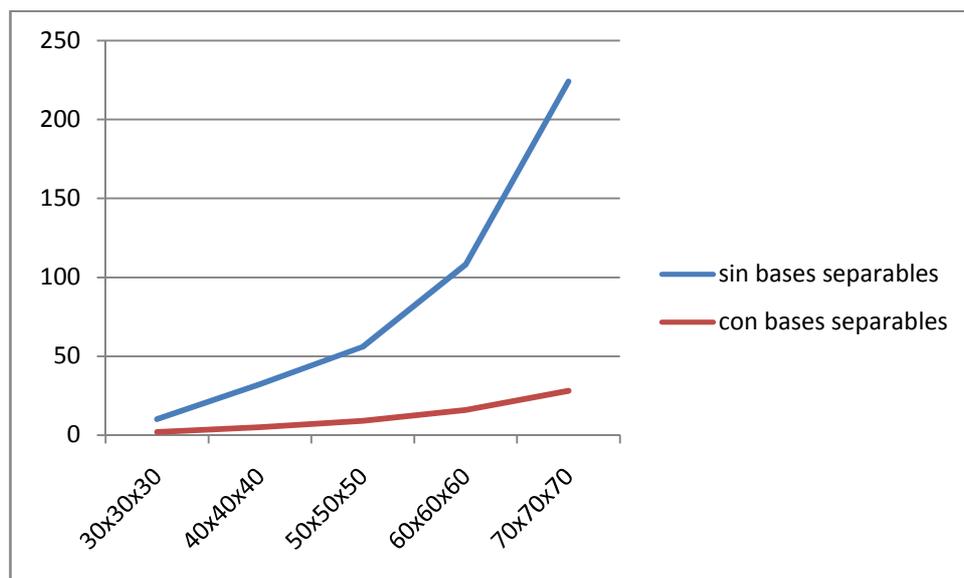
Una de las principales ventajas que han sido expuestas a lo largo del documento es la mejora en los tiempos de ejecución a la hora de ejecutar programas. En este apartado se va a cuantificar el valor de esa mejora.

8.1. Comparativa de tiempos entre el uso o no del filtro orientable

8.1.1. Implementación de filtro separable

El empleo de funciones separables agiliza el proceso de filtrado debido a que ahorra gran cantidad de operaciones de convolución en tres dimensiones. El siguiente estudio muestra una pequeña comparativa de los tiempos de ejecución para el programa diseñado en Matlab. El cálculo de los tiempos se realiza mediante la función *Profiler*. Para que no fuera muy arduo la obtención de los resultados se ha realizado el análisis del filtrado únicamente para las direcciones de las bases. Los números que se muestran en la tabla son valores de tiempo en segundos.

	30x30x30	40x40x40	50x50x50	60x60x60	70x70x70
sin bases separables	10	32	56	108	224
con bases separables	2	5	9	16	28



Como se observa en la gráfica, el coste de ejecución de las bases que no están implementadas mediante funciones orientables crece mucho más rápido a medida que aumenta el tamaño del volumen. Aunque esta característica no sea la que ahorra más tiempo de ejecución, sí que es un valor más a tener en cuenta en el resultado total del filtrado.

8.1.2. Implementación de las bases para el uso del filtro orientable

Como se mostró en la sección 6, los tiempos de mejora de los filtros orientables

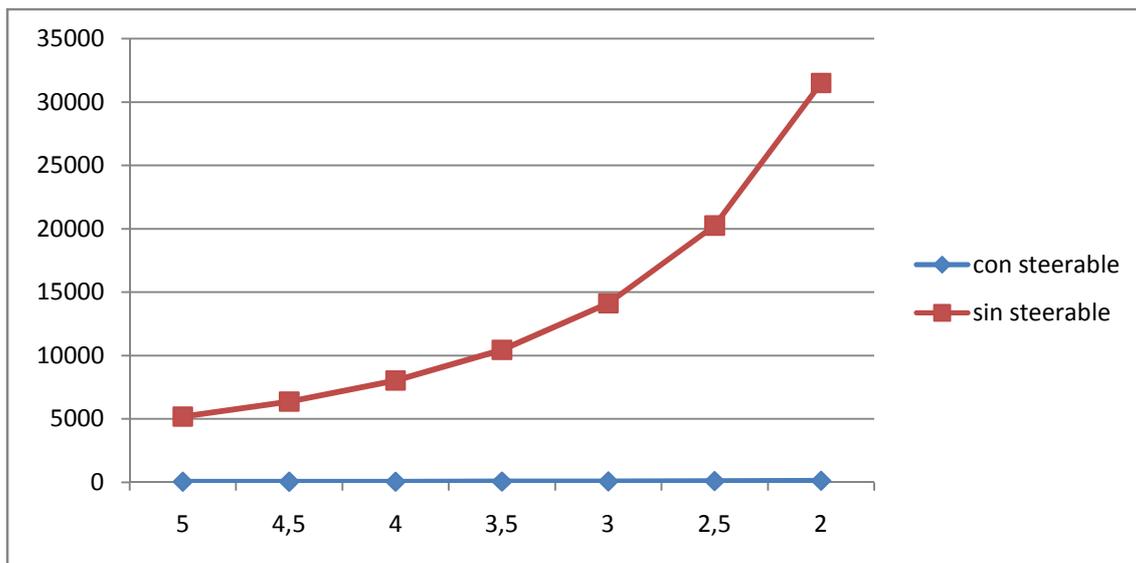
t_1 y t_2 mediante el filtrado de volúmenes en Matlab de tamaño 65x65x65. Posteriormente se aplicarán las formulas que se muestran a continuación para estimar los resultados de los tiempos.

$$Mt_1 + Nt_2.$$

Sistema no orientable: Nt_1 .

Los valores obtenidos son $t_1 = 3$ segundos y $t_2 = 0,01$ segundos.

	5	4,5	4	3,5	3	2,5	2
con steerable	35,24	39,18	44,7	52,76	65,04	85,46	122,94
sin steerable	5172	6354	8010	10428	14112	20238	31482



8.2. Comparativas de tiempos de ejecución con diferentes valores de muestreo orientativo

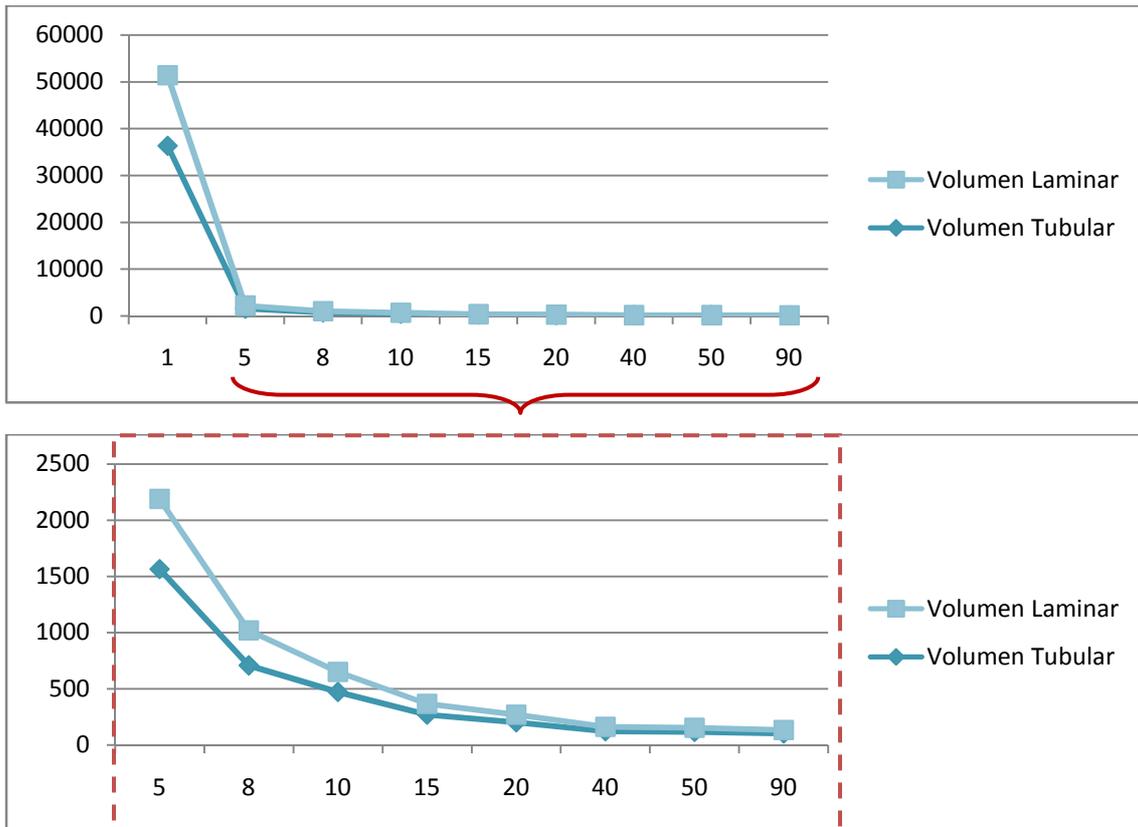
Como se analizó en la sección 6 de este documento, los filtros diseñados realizan un muestreo en diferentes direcciones del espacio para obtener las estructuras de los volúmenes analizados. El número de muestras que se generan en dichos procesos puede calcularse a partir del valor de *deltaAng*. En la siguiente gráfica se muestra dicha información. En principio, la curva que se presenta debe tener la misma forma que la se generaría analizando los tiempos reales de ejecución del proceso de filtrado completo.

	0,5	1	1,5	2	2,5	3	3,5	4	4,5	5
Número de Muestras	165732	41614	18576	10494	6746	4704	3476	2670	2118	1724



A continuación se muestran los datos reales obtenidos para el filtrado de los volúmenes tubulares y la laminares para diferentes valores de *deltaAng*. Los tamaños de los volúmenes son 462x120x212 y 512x512x19 respectivamente.

	1	5	8	10	15	20	40	50	90
Volumen Tubular	36300	1564,125	707,128	470,997	268,444	199,628	120,1498	115,104	100,8909
Volumen Laminar	15035,67	623,654	310,055	177,627	95,461	67,096	38,758	35,97	30,573



Como muestra la figura, para valores inferiores a $\Delta Ang = 15$ el tiempo de ejecución del programa de dispara, por lo que es importante preguntarse cuál es la relación óptima entre el tiempo y la calidad de los resultados. Como se mostrará a continuación para valores de ΔAng comprendidos entre 1 y 20 los resultados en calidad no muestran ninguna diferencia. Esto supone que no es necesario realizar un muestreo con valores muy pequeños de ΔAng para obtener unos buenos resultados. De esta forma se va a mostrar la relación señal/ruido para obtener un análisis preciso de las diferencias de las calidades.

Si consideramos como ruido la diferencia entre la calidad resultante de un volumen filtrado para $\Delta Ang = 5$ con respecto a otro filtrado para $\Delta Ang = 1$. La fórmula que establece la SNR es la que se muestra a continuación.

$$SNR = \frac{S}{N} = \frac{\sigma^2(vol5)}{\sigma^2(diff\ vol5 - vol1)}$$

$$SNR \rightarrow vol5 = vol1 - ruido$$

Para $\Delta Ang = 20$ el valor de SNR es:

$$SNR = \frac{S}{N} = \frac{[(1,195) * 10^{-7}]^2}{[(9,410) * 10^{-9}]^2} = 161,2711$$

$$10 \log_{10}(161,2711) = 22,07 \text{ dB.}$$

Para $\Delta Ang = 10$ el valor de SNR es:

$$SNR = \frac{S}{N} = \frac{[(1,188) * 10^{-7}]^2}{[(3,442) * 10^{-9}]^2} = 1,913 * 10^3$$

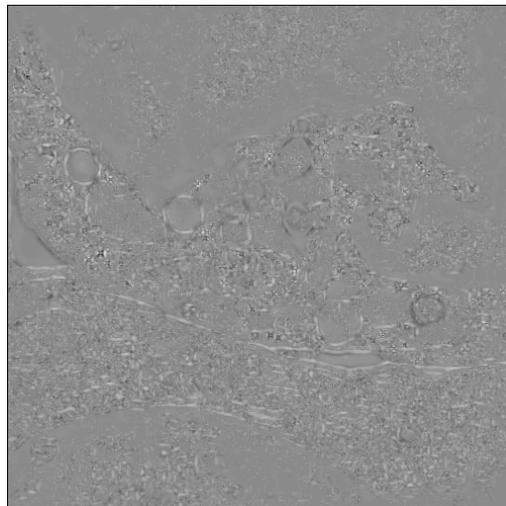
$$10 \log_{10}(1.913 * 10^3) = 30.7601 \text{ dB.}$$

Para $\Delta Ang = 90$ el valor de SNR es:

$$SNR = \frac{S}{N} = \frac{[(1,611) * 10^{-7}]^2}{[(9,896) * 10^{-8}]^2} = 2,6502$$

$$10 \log_{10}(2,6502) = 4,2327 \text{ dB.}$$

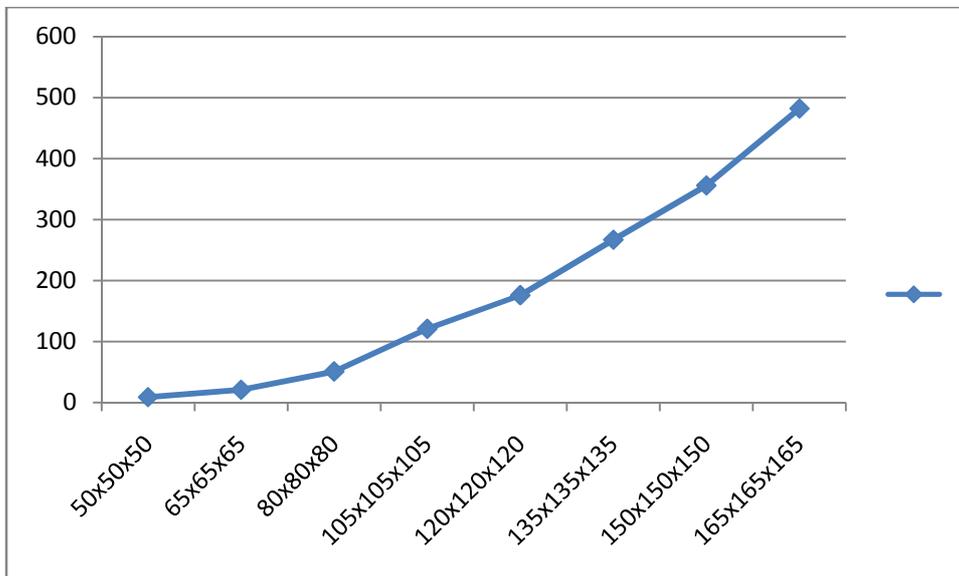
A continuación se muestran la diferencia gráfica entre un volumen filtrado para $\Delta Ang = 90$ y $\Delta Ang = 1$.



La diferencia que se observa de calidades es tan pequeña que demuestra que los valores de ΔAng no influyen sustancialmente a los resultados visuales de las imágenes.

8.3. Comparativa de tiempos entre los diferentes tamaños de los volúmenes

50x50x50	9
65x65x65	21
80x80x80	51
105x105x105	121
120x120x120	176
135x135x135	267
150x150x150	356
165x165x165	482



9. Trabajos Futuros

Finalizado el proyecto, las aplicaciones directas a la vida real que van a tener los filtros desarrollados pueden ser varias. En principio las implementaciones serán añadidas a la librería del Xmipp del CNB (Centro Nacional de Biotecnología), para poder ayudar a posibles investigaciones futuras. Por otra parte, también está previsto que se publiquen estos resultados en revistas de carácter científico para poder mostrar los avances obtenidos en este campo. Ésto es importante dado que la mayoría de los documentos o tesis que se han empleado en el desarrollo de este proyecto han sido seleccionadas desde publicaciones del IEEE o de otras revistas. De esta manera, el hecho de publicar este estudio es un medio para ofrecer a otras personas la oportunidad de seguir el camino de la investigación mediante nuevas técnicas y avances.

10. Conclusiones

Con toda la información aportada en este documento, se puede decir que los filtros implementados presentan avances importantes en el terreno de la biotecnología. La solución que se ha diseñado proporciona una manera rápida y eficaz de determinar las estructuras de volúmenes biológicos, sea cual sea el tamaño de los mismos o de las configuraciones que presente. Los análisis han demostrado que los tiempos de obtención de los resultados son mucho menores que anteriormente y que la calidad, sin duda alguna, no tiene que estar sometida a largos procesos de espera.

11. Bibliografía

- [1] Willian T. Freeman and Edward H. Adelson: “The Design and Use of Steerable Filters”, IEEE Transactions on pattern Analysis and Machine Intelligence, Vol. 13, number 9, September 1991.
- [2] Mathews Jacob, Thierry Blu, Cedric Vaillant, John H. Maddocks and Michael Unser: “3-D Shape Estimation of DNA Molecules From Stereo Cryo-Electron Micro-Graphs Using a Projection-Steerable Snake”, IEEE Transactions on Image processing, Vol. 15, number 1, January 2006.
- [3] Mathews Jacob and Michael Unser: “Design of Steerable Filters for Feature Detection Using Canny-Like Criteria”. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 26, number 8, August 2004.
- [4] Konstantinos G. Derpanis and Jacob M. Gryn: “Three-Dimensional N^{th} Derivative of Gaussian Separable Steerable Filters”. Department of Computer Science and Centre for Vision Research, York University, Toronto, Canada, IEEE, 2005.
- [5] François Aguet, Mathews Jacob and Michael Unser: “Three-Dimensional Feature Detection Using Optimal Steerable Filters” IEEE, 2005.
- [6] Weichuan Yu, Kostas Daniilidis, and Gerald Sommer: “A New 3D Orientation Steerable Filter”. Instiut für Informatik. Christian-Albrechts-Universität.
- [7] ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG INSTITUT FÜR INFORMATIK; Lehrstuhl für Mustererkennung und Bildverarbeitung: “Spherical Derivatives for Steerable Filtering in 3D”. Internal Report 3/07. November 2007.
- [8] Willian T. Freeman: “Steerable Filter and Local Analysis of Image Structure” Massachusetts Institute of Technology, June 1992.
- [9] Prof. David J. Heeger: “Notes on Steerable Filters”, September 15, 1998.
- [10] Derivadas direccionales; http://es.wikipedia.org/wiki/Derivada_direccional

- [11] Retinal Vessel Centerline Extraction Using Multiscale Matched Filter, Confidence and Edge Measures; Michal Sofka, Student member; IEEE and Charles V.Stewart, Member; IEEE; IEEE 2006
- [12] Active shape model segmentation using local edge structures and AdaBoost
Shuyu Li, Litao Zhu, Tianzi Jiang.
- [13] R. Courant and D. Hilbert “Methods of Mathematical Physics”, vol1 New York: Wiley Interscience, 1953.
- [14] “Image Processing and 3D Reconstruction in Electron Microscopy” IEEE SIGNAL PROCESSING MAGAZINE, MAY 2006, Jose-Jesus Fernandez, Carlos Oscar S. Sorzano, Roberto Marabini, and Jose-Maria Carazo
- [16] “Comparison of single-particle analysis and electron tomography approaches: an overview”. Journal of Microscopy, Vol. 232, Pt 3 2008, pp. 562–579; S. JONIĆ, C.O.S. SORZANO & N. BOISSET

12. Anexos (códigos de Matlab y Xmipp)

12.1. Código C++ con librería Xmipp

steerable.h

```
#ifndef STEERABLE_H
#define STEERABLE_H

#include "matrix3d.h"
#include <vector>

class Steerable
{
public:

    std::vector< Matrix3D<double> > basis;

    /*Constructor con parametros*/
    Steerable(double sigma, Matrix3D<double> &Vtomograph,
              double deltaAng, std::string &cadena);

    /*funciones de implementacion del filtro*/
    void generate1DFilters(double sigma,
                          const Matrix3D<double> &Vtomograph,
                          std::vector< Matrix1D<double> > &hx,
```

PROYECTO FINAL DE CARRERA

```
std::vector< Matrix1D<double> > &hy,
std::vector< Matrix1D<double> > &hz);

/* funcion de generacion de volumen*/
void generate3DFilter(Matrix3D<double>& h3D,
std::vector< Matrix1D<double> > &hx,
std::vector< Matrix1D<double> > &hy,
std::vector< Matrix1D<double> > &hz);

void steerableFilter(const Matrix3D<double> &Vtomograph,
double sigma,Matrix3D<double> &Vfiltered);

void singleFilter(const Matrix3D<double>& Vin,
const Matrix1D<double> &hx, const Matrix1D<double> &hy,
const Matrix1D<double> &hz, Matrix3D<double> &Vout);

};
#endif
```

steerable.cpp

```
#include "steerable.h"
#include "fft.h"
#include "volume.h"

#define DEBUG_FILTERGENERATION
#define DEBUG_DIRECTION
#define DEBUG_BASIS

void Steerable::generate1DFilters(double sigma,
const Matrix3D<double> &Vtomograph,
std::vector< Matrix1D<double> > &hx,
std::vector< Matrix1D<double> > &hy,
std::vector< Matrix1D<double> > &hz){

// Inicializacion
hx.clear();
hy.clear();
hz.clear();
Matrix1D<double> aux;
aux.clear();

aux.initZeros(XSIZE(Vtomograph));
aux.setXmippOrigin();
for (int i=0; i<6; i++) hx.push_back(aux);

aux.initZeros(YSIZE(Vtomograph));
aux.setXmippOrigin();
for (int i=0; i<6; i++) hy.push_back(aux);

aux.initZeros(ZSIZE(Vtomograph));
aux.setXmippOrigin();
for (int i=0; i<6; i++) hz.push_back(aux);

double sigma2=sigma*sigma;
```

PROYECTO FINAL DE CARRERA

```

//std::cout<<"sigma2 = "<<sigma2<<std::endl;
double k1 = 1.0/pow((2.0*PI*sigma),(3.0/2.0));
double k2 = -1.0/(sigma2);

FOR_ALL_ELEMENTS_IN_MATRIX1D(hx[0])
{
    double i2=i*i;
    double g = -exp(-i2/(2.0*sigma2));
    hx[0](i) = k1*k2*g*(1.0-(i2/sigma2));
    hx[1](i) = k1*k2*g;
    hx[2](i) = k1*k2*g;
    hx[3](i) = k1*k2*k2*g*i;
    hx[4](i) = k1*k2*k2*g*i;
    hx[5](i) = k1*k2*k2*g;
}
FOR_ALL_ELEMENTS_IN_MATRIX1D(hy[0])
{
    double i2=i*i;
    double g = -exp(-i2/(2.0*sigma2));
    hy[0](i) = g;
    hy[1](i) = g*(1.0-(i2/sigma2));
    hy[2](i) = g;
    hy[3](i) = g*i;
    hy[4](i) = g;
    hy[5](i) = g*i;
}
FOR_ALL_ELEMENTS_IN_MATRIX1D(hz[0])
{
    double i2=i*i;
    double g = -exp(-i2/(2.0*sigma2));
    hz[0](i) = g;
    hz[1](i) = g;
    hz[2](i) = g*(1.0-(i2/sigma2));
    hz[3](i) = g;
    hz[4](i) = g*i;
    hz[5](i) = g*i;
}
}

void Steerable::generate3DFilter(Matrix3D<double>& h3D,
    std::vector< Matrix1D<double> > &hx,
    std::vector< Matrix1D<double> > &hy,
    std::vector< Matrix1D<double> > &hz)
{

    h3D.initZeros(XSIZE(hz[0]),XSIZE(hy[0]),XSIZE(hx[0]));
    h3D.setXmippOrigin();
    FOR_ALL_ELEMENTS_IN_MATRIX3D(h3D)
        for (int n=0; n<6; n++)
            h3D(k,i,j)+=(hz[n](k)*hy[n](i)*hx[n](j));
}

Steerable::Steerable(double sigma, Matrix3D<double> &Vtomograph,
    const double deltaAng, std::string &cadena)
{
    Matrix3D<double> Vfiltered;
    steerableFilter(Vtomograph,sigma,Vfiltered);
}
    
```

PROYECTO FINAL DE CARRERA

```

// seleccion del filtro (a+bx) y c

double a; double b; double c;

if (cadena == "celula"){
    // para la celula
    a = -(1.0/4.0);
    b = 5.0/4.0;
    c = 5.0/2.0;
}
else{
    // para el filamento es por defecto
    a = 1.0;
    b = -(5.0/3.0);
    c = 10.0/3.0;
}

/* inicializacion del volumen de las direcciones*/
#ifdef DEBUG_DIRECTION
    Vtomograph.initZeros();
    std::vector< Matrix3D<double> > filterDirect;
    filterDirect.push_back(Vtomograph);
    filterDirect.push_back(Vtomograph);
    filterDirect.push_back(Vtomograph);
#endif
// esto se hace para que la direccion inicial del filterDirect sea (1,0,0)
filterDirect[0].initConstant(1);

// construccion de un volumen en la direccion (1 0 0)
// muestreo en 0 y 180 grados

double u0=1;
double u1=0;
double u2=0;
FOR_ALL_ELEMENTS_IN_MATRIX3D(Vtomograph){
    Vtomograph(k,i,j) = basis[0](k,i,j) * (a+b*u0*u0) +
        basis[1](k,i,j) * (a+b*u1*u1) +
        basis[2](k,i,j) * (a+b*u2*u2) +
        c*(basis[3](k,i,j) * u0*u1 +
            basis[4](k,i,j) * u0*u2 +
            basis[5](k,i,j) * u1*u2);
}
#ifdef DEBUG_DIRECTION
    VolumeXmipp savePPP1;
    savePPP1() = Vtomograph;
    savePPP1.write("PPPorientacion_0_0_1.vol");
#endif

double Ntilt = round(180.0/deltaAng);
std::cout <<" Ntilt = " << Ntilt <<std::endl;
for (int i=1;i<Ntilt;i++){
    double tilt = deltaAng*i;
    //std::cout <<" tilt = " << tilt <<std::endl;
    //std::cout <<" sin(tilt*(PI/180.0)) = " << sin(tilt*(PI/180.0)) <<std::endl;
    //std::cout <<" abs(sin(tilt*(PI/180.0))) = " << abs(sin(tilt*(PI/180.0))) <<std::endl;
}
    
```

PROYECTO FINAL DE CARRERA

```

double deltaRoti = deltaAng/(sin(tilt*(PI/180.0)));
//std::cout <<" deltaRoti = " << deltaRoti <<std::endl;
double NrotP = round(360.0/deltaRoti);
for (int j=0;j<NrotP;j++){
    double rot = j*deltaRoti;
    double u0 = sin(rot*(PI/180.0))*cos(tilt*(PI/180.0));//x
    double u1 = sin(rot*(PI/180.0))*sin(tilt*(PI/180.0));//y
    double u2 = cos(rot*(PI/180.0)); //z
    double var1 = (a+b*u0*u0);
    double var2 = (a+b*u1*u1);
    double var3 = (a+b*u2*u2);
    /*std::cout <<" u0 = " << u0 << ",
    <<" u1 = " << u1 << ",
    <<" u2 = " << u2
    << std::endl;*/
    FOR_ALL_ELEMENTS_IN_MATRIX3D(Vtomograph)
    {
        double aux = basis[0](k,i,j) * var1 +
            basis[1](k,i,j) * var2 +
            basis[2](k,i,j) * var3 +
            c*(basis[3](k,i,j) * u0*u1 +
            basis[4](k,i,j) * u0*u2 +
            basis[5](k,i,j) * u1*u2);

        if(aux>Vtomograph(k,i,j)){
            Vtomograph(k,i,j) = aux;
            #ifdef DEBUG_DIRECTION
                filterDirect[0](k,i,j) = u0;
                filterDirect[1](k,i,j) = u1;
                filterDirect[2](k,i,j) = u2;
            #endif
        }
    }
}

#ifdef DEBUG_DIRECTION
    VolumeXmipp savep;
    savep()=filterDirect[0]; save.write("PPPDirectx.vol");
    savep()=filterDirect[1]; save.write("PPPDirecty.vol");
    savep()=filterDirect[2]; save.write("PPPDirectz.vol");
#endif
}

void Steerable::steerableFilter(const Matrix3D<double> &Vtomograph,
double sigma, Matrix3D<double> &Vfiltered)
{
    std::vector< Matrix1D<double> > hx, hy, hz;
    generate1DFilters(sigma, Vtomograph, hx, hy, hz);

    // comprobar la imagen
    Matrix3D<double> FILTRO(Vfiltered);
    #ifdef DEBUG_DIRECTION
        VolumeXmipp save;
        generate3DFilter(FILTRO,hx, hy, hz);
        save() = FILTRO; save.write("FILTRO.vol");
    #endif
    Vfiltered.initZeros(Vtomograph);
}

```

PROYECTO FINAL DE CARRERA

```

Matrix3D<double> aux(Vfiltered);
for (int n=0; n<6; n++)
{
    singleFilter(Vtomograph,hx[n],hy[n],hz[n],aux);
    basis.push_back(aux);
}
}

void Steerable::singleFilter(const Matrix3D<double>& Vin,
    const Matrix1D<double> &hx, const Matrix1D<double> &hy, const Matrix1D<double> &hz,
    Matrix3D<double> &Vout){
    Vout.initZeros(Vin);

    // Filter in X

    Matrix1D< std::complex<double> > H, Aux;

    FourierTransform(hx,H);
    FOR_ALL_ELEMENTS_IN_MATRIX1D(H){
        if(i<(XSIZE(H)/2)){
            H(i)*= pow(-1,i);
        }else{
            H(i)*= pow(-1,(XSIZE(H)-i));
        }
    }

    Matrix1D<double> aux(XSIZE(Vin));
    for (int k=0; k<ZSIZE(Vin); k++)
        for (int i=0; i<YSIZE(Vin); i++)
            {
                for (int j=0; j<XSIZE(Vin); j++)
                    DIRECT_VEC_ELEM(aux,j)=DIRECT_VOL_ELEM(Vin,k,i,j);
                FourierTransform(aux,Aux);
                Aux*=H;
                InverseFourierTransform(Aux,aux);
                for (int j=0; j<XSIZE(Vin); j++)
                    DIRECT_VOL_ELEM(Vout,k,i,j)=XSIZE(Aux)*DIRECT_VEC_ELEM(aux,j);
            }

    // Filter in Y
    FourierTransform(hy,H);
    aux.initZeros(YSIZE(Vin));
    FOR_ALL_ELEMENTS_IN_MATRIX1D(H){
        if(i<(XSIZE(H)/2)){
            H(i)*= pow(-1,i);
        }else{
            H(i)*= pow(-1,(XSIZE(H)-i));
        }
    }

    for (int k=0; k<ZSIZE(Vin); k++)
        for (int j=0; j<XSIZE(Vin); j++)
            {
                for (int i=0; i<YSIZE(Vin); i++)
                    DIRECT_VEC_ELEM(aux,i)=DIRECT_VOL_ELEM(Vout,k,i,j);
                FourierTransform(aux,Aux);
                Aux*=H;
            }
    
```

PROYECTO FINAL DE CARRERA

```

InverseFourierTransform(Aux,aux);
for (int i=0; i<YSIZE(Vin); i++)
    DIRECT_VOL_ELEM(Vout,k,i,j)=XSIZE(Aux)*DIRECT_VEC_ELEM(aux,i);
}

// Filter in Z
FourierTransform(hz,H);
aux.initZeros(ZSIZE(Vin));
FOR_ALL_ELEMENTS_IN_MATRIX1D(H){
    if(i<(XSIZE(H)/2)){
        H(i)*= pow(-1,i);
    }else{
        H(i)*= pow(-1,(XSIZE(H)-i));
    }
}

for (int i=0; i<YSIZE(Vin); i++)
    for (int j=0; j<XSIZE(Vin); j++)
    {
        for (int k=0; k<ZSIZE(Vin); k++)
            DIRECT_VEC_ELEM(aux,k)=DIRECT_VOL_ELEM(Vout,k,i,j);
        FourierTransform(aux,Aux);
        Aux*=H;
        InverseFourierTransform(Aux,aux);
        for (int k=0; k<ZSIZE(Vin); k++)
            DIRECT_VOL_ELEM(Vout,k,i,j)=XSIZE(Aux)*DIRECT_VEC_ELEM(aux,k);
    }
}

```

12.2. Código en Matlab

function BASESfiltroSteerable(Xdim, Ydim, Zdim, sigma, deltaAng)

```

[g3xx g3yy g3zz g3xy g3xz g3yz] = filtroSteerable(Xdim, Ydim, Zdim, sigma);

f = zeros(Xdim,Ydim,Zdim);

VolEnerg = f;

XvolDirec = f;

YvolDirec = f;

ZvolDirec = f;

% para la muestra 0 y la muestra 180
u = [1 0 0]

f1 = g3xx*(1-(5/3)*u(1)^2) + g3yy*(1-(5/3)*u(2)^2) + g3zz*(1-(5/3)*u(3)^2) +
(10/3)*(g3xy*u(1)*u(2) + g3xz*u(1)*u(3) + g3yz*u(2)*u(3));

VolEnerg = f1;

disp('Bases Calculadas')

Ntilt = round(180/deltaAng)

for i=1:Ntilt-1

```

PROYECTO FINAL DE CARRERA

```

i
tilt = deltaAng*i
deltaRoti = deltaAng/abs(sind(tilt))
NrotP = round(360/deltaRoti);
for j=0:NrotP-1
    rot = j*deltaRoti;
    x = sind(rot)*cosd(tilt);
    y = sind(rot)*sind(tilt);
    z = cosd(rot);
    u = [x y z];
    f = g3xx*(1-(5/3)*u(1)^2) + g3yy*(1-(5/3)*u(2)^2) + g3zz*(1-(5/3)*u(3)^2) +
(10/3)*(g3xy*u(1)*u(2) + g3xz*u(1)*u(3) + g3yz*u(2)*u(3));
    [VolEnerg XvolDirec YvolDirec ZvolDirec] =
AnalisSteerable(VolEnerg,Xdim,Ydim,Zdim,f,XvolDirec,YvolDirec,ZvolDirec,x,y,z);
end
end
quiverSteerable(Xdim,Ydim,Zdim,VolEnerg,XvolDirec,YvolDirec,ZvolDirec);
end

function [g3xx g3yy g3zz g3xy g3xz g3yz] = filtroSteerable(Xdim, Ydim, Zdim, sigma)
    filename = 'cylinder_0_0_0.vol';
    I = [];
    I = abrirVolumen(filename,Xdim,Ydim,Zdim);
    x0 = -floor(Xdim/2);
    y0 = -floor(Ydim/2);
    z0 = -floor(Zdim/2);
    xF = x0+Xdim-1;
    yF = y0+Ydim-1;
    zF = z0+Zdim-1;
    k1 = 1/((2*pi*sigma)^(3/2));
    k2 = -1/(sigma^2);
    
```

PROYECTO FINAL DE CARRERA

```

p1 = zeros(1,Xdim);
p2 = zeros(1,Ydim);
p3 = zeros(1,Zdim);
Xg3xx = p1;Yg3xx = p2;Zg3xx = p3;
Xg3yy = p1;Yg3yy = p2;Zg3yy = p3;
Xg3zz = p1;Yg3zz = p2;Zg3zz = p3;
Xg3xy = p1;Yg3xy = p2;Zg3xy = p3;
Xg3xz = p1;Yg3xz = p2;Zg3xz = p3;
Xg3yz = p1;Yg3yz = p2;Zg3yz = p3;
% formacion de los filtros separables
for x=x0:xF
    x_x0=x-x0;
    g1 = -exp(-(x^2)/(2*sigma^2));
    Xg3xx(x_x0+1) = k1*k2*g1*(1-(x/sigma)^2);
    Xg3yy(x_x0+1) = k1*k2*g1;
    Xg3zz(x_x0+1) = k1*k2*g1;
    Xg3xy(x_x0+1) = k1*k2^2*g1*x;
    Xg3xz(x_x0+1) = k1*k2^2*g1*x;
    Xg3yz(x_x0+1) = k1*k2^2*g1;
end
for y=y0:yF
    y_y0=y-y0;
    g2 = -exp(-(y^2)/(2*sigma^2));
    Yg3xx(y_y0+1) = g2;
    Yg3yy(y_y0+1) = g2*(1-(y/sigma)^2);
    Yg3zz(y_y0+1) = g2;
    Yg3xy(y_y0+1) = g2*y;
    Yg3xz(y_y0+1) = g2;
    Yg3yz(y_y0+1) = g2*y;
end

```

PROYECTO FINAL DE CARRERA

for z=z0:zF

z_z0=z-z0;

g3 = -exp(-(z^2)/(2*sigma^2));

Zg3xx(z_z0+1) = g3;

Zg3yy(z_z0+1) = g3;

Zg3zz(z_z0+1) = g3*(1-(z/sigma)^2);

Zg3xy(z_z0+1) = g3;

Zg3xz(z_z0+1) = g3*z;

Zg3yz(z_z0+1) = g3*z;

end

g3xx = nconvZ(Zg3xx,nconvY(Yg3xx,nconvX(Xg3xx,I,Xdim,y0,yF,...
z0,zF),Ydim,x0,xF,z0,zF),Zdim,x0,xF,y0,yF);

g3yy = nconvZ(Zg3yy,nconvY(Yg3yy,nconvX(Xg3yy,I,Xdim,y0,yF,...
z0,zF),Ydim,x0,xF,z0,zF),Zdim,x0,xF,y0,yF);

g3zz = nconvZ(Zg3zz,nconvY(Yg3zz,nconvX(Xg3zz,I,Xdim,y0,yF,...
z0,zF),Ydim,x0,xF,z0,zF),Zdim,x0,xF,y0,yF);

g3xy = nconvZ(Zg3xy,nconvY(Yg3xy,nconvX(Xg3xy,I,Xdim,y0,yF,...
z0,zF),Ydim,x0,xF,z0,zF),Zdim,x0,xF,y0,yF);

g3xz = nconvZ(Zg3xz,nconvY(Yg3xz,nconvX(Xg3xz,I,Xdim,y0,yF,...
z0,zF),Ydim,x0,xF,z0,zF),Zdim,x0,xF,y0,yF);

g3yz = nconvZ(Zg3yz,nconvY(Yg3yz,nconvX(Xg3yz,I,Xdim,y0,yF,...
z0,zF),Ydim,x0,xF,z0,zF),Zdim,x0,xF,y0,yF);

end

**function [VolEnerg XvolDirec YvolDirec ZvolDirec] =
AnalisisSteerable(VolEnerg,Xdim,Ydim,Zdim,Vol,XvolDirec,YvolDirec,ZvolDirec,x,y,z)**

for i=1:Xdim

for j=1:Ydim

for k=1:Zdim

if (Vol(i,j,k)>VolEnerg(i,j,k))

VolEnerg(i,j,k) = Vol(i,j,k);

XvolDirec(i,j,k) = x;

YvolDirec(i,j,k) = y;

PROYECTO FINAL DE CARRERA

```

        ZvolDirec(i,j,k) = z;
    end
end
end
end
end
end
function X = nconvX(Vx,I,Xdim,y0,yF,z0,zF)
X = [];
VX = fft(Vx);
SOL = [];
for z=z0:zF
    z_z0=z-z0;
    for y=y0:yF
        y_y0=y-y0;
        vectorX = reshape(I(:,y_y0+1,z_z0+1),1,Xdim);
        VECTORX = fft(vectorX);
        SOL = VX.*VECTORX;
        X(:,y_y0+1,z_z0+1) = fftshift(iff(SOL));
    end
end
end
end
function Y = nconvY(Vy,I,Ydim,x0,xF,z0,zF)
Y = [];
VY = fft(Vy);
SOL = [];
for z=z0:zF
    z_z0=z-z0;
    for x=x0:xF
        x_x0=x-x0;
        vectorY = reshape(I(x_x0+1,.,z_z0+1),1,Ydim);
        VECTORY = fft(vectorY);

```

PROYECTO FINAL DE CARRERA

```

    SOL = VY.*VECTORY;

    Y(x_x0+1, :, z_z0+1) = fftshift(iff(SOL));

end

end

end

function Z = nconvZ(Vz,I,Zdim,x0,xF,y0,yF)

    Z = [];

    VZ = fft(Vz);

    SOL = [];

    for x=x0:xF

        x_x0=x-x0;

        for y=y0:yF

            y_y0=y-y0;

            vectorZ = reshape(I(x_x0+1,y_y0+1,:),1,Zdim);

            VECTORZ = fft(vectorZ);

            SOL = VZ.*VECTORZ;

            Z(x_x0+1,y_y0+1,:) = fftshift(iff(SOL));

        end

    end

end

end

function V = abrirVolumen(filename,Xdim,Ydim,Zdim)

    headerSize = 1024;

    endianness = 'ieee-le';

    fid = fopen(filename, 'rb', endianness);

    fseek(fid, headerSize, 'bof');

    V = fread(fid, Xdim*Ydim*Zdim, 'float', 0, endianness);

    V = reshape(V,Xdim,Ydim,Zdim);

    fclose(fid);

end

```