



Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación

Detección de candidatos a pólipos en colonoscopia virtual mediante análisis de curvatura

Trabajo Especial de Grado presentado por
Br. Eduardo César Buysse El Hennaoni
Br. María Elena Poó Arguello

ante la Ilustre
Universidad Central de Venezuela
para optar al título de
Licenciado en Computación

Tutor:
Prof. Rhadamés Carmona
Co-tutor:
Prof. Ernesto Coto

Caracas, Octubre 2011

Agradecimientos

La presente tesis es un esfuerzo en el que, directa o indirectamente, participaron muchas personas, bien sea leyendo, opinando, corrigiendo, apoyando, criticando, aportando, etc. Agradecemos a aquellos familiares, amigos o profesores que en los momentos de angustia nos tendieron una mano o nos animaron, les damos también las gracias a quienes compartieron con nosotros las alegrías.

Gracias a nuestro tutor el Prof. Rhadamés Carmona por su paciencia y apoyo, gracias al co-tutor el Prof. Ernesto Coto por su gran colaboración, gracias al Prof. Héctor Navarro por dedicar un poco de su tiempo y ayudarnos cuando lo necesitábamos y gracias al Prof. Luis Manuel Hernández por su participación como jurado.

Yo, María Elena Poó, agradezco principalmente a mi compañero Eduardo por ayudarme, apoyarme y soportarme en este largo trabajo especial de grado que finalmente terminamos. Gracias a mi mami hermosa por creer en mí y ser mis muletas cuando me canse de andar. Gracias a mi papi que desde el cielo llena mi camino de luz. Gracias a mi novio, a mis hermanos, a mis amigos y a mis profesores por acompañarme en esta aventura que es la vida y hacer de mí una mejor persona. Finalmente, ¡gracias a la UCV por darme los mejores años de mi vida! ¡GRACIAS a todos!

Yo, Eduardo Buysse, agradezco a mi compañera María, sobre todo por obligarme a seguir en los momentos que no quería. A mis padres, mis hermanos, mi novia, mis suegros, mis familiares, mis amigos y a mis profesores, por siempre tener fe en mí y demostrármelo cuando me hacía falta. A todos ellos, ¡muchas gracias!

Resumen

La detección automática de pólipos es un amplio campo de estudio dentro de la computación gráfica. El principal objetivo es automatizar la detección de pólipos a fin de ayudar al especialista en la tediosa tarea de analizar las imágenes médicas y además minimizar el tiempo de espera para cada paciente. En la actualidad, existen diversas técnicas de detección de pólipos que utilizan diferentes métodos tales como líneas de curvatura, difusión de calor, cálculo del índice de forma, etc. Uno de los mayores problemas de estas técnicas es reducir el tiempo de ejecución debido a que hay que realizar numerosos cálculos. En este trabajo presentamos una técnica de detección automática de pólipos que reduce el tiempo de respuesta. Para este propósito utilizamos programación paralela en CUDA. Los resultados obtenidos muestran una notoria minimización del tiempo de respuesta. La efectividad de nuestro algoritmo es difícil medirla, esto debido a que es necesario el análisis médico de los resultados para conocer el porcentaje de aciertos. Sin embargo, analizándolos en un visualizador del colon, los resultados muestran potenciales candidatos a pólipos por lo que se recomienda entre los trabajos futuros, el análisis clínico de los mismos.

Palabras claves: colonoscopia virtual, detección de pólipos, análisis de curvatura.

Índice General

Agradecimientos	III
Resumen	IV
Introducción.....	VII
CAPÍTULO 1: Marco teórico.....	1
1.1. Adquisición de Imágenes Médicas del Colon	3
1.2. Colonoscopia virtual.....	9
1.2.1. Preparación para el examen de colonoscopia virtual	11
1.2.2. Forma en que se realiza un examen de colonoscopia virtual	11
1.2.3. Cuáles son los riesgos.....	12
1.2.4. Consideraciones especiales.....	13
1.3. Procesamiento de datos.....	14
1.4. Visualización y navegación de los datos.....	15
CAPÍTULO 2: Antecedentes	18
2.1. Técnicas de detección de pólipos.....	18
2.1.1. Detección de pólipos basada en curvatura	19
2.1.2. Detección de pólipos basada en ajuste de superficies.....	24
2.1.3. Detección de pólipos basada en líneas de curvatura.....	25
2.1.4. Detección de pólipos basada en difusión de calor.....	30
2.1.5. Detección de pólipos basada en línea de agua	32
2.1.6. Reducción de falsos positivos.....	36
2.1.7. Combinación de malla, volumen y representación de líneas de curvatura para la detección de pólipos en colonografía por TC.....	39
2.2. Aproximación de la derivada.....	41
2.3. Programación en GPU	44
2.3.1. Arquitectura CUDA.....	46
2.3.2. Organización de hilos (threads).....	46
2.3.3. Jerarquía de memoria	47
2.3.4. Flujo de procesamiento.....	48

CAPÍTULO 3: Detalles de implementación.....	50
3.1. Plataforma de desarrollo.....	50
3.2. Implementación del método de detección de pólipos	50
3.2.1. Extracción de las características geométricas 3D de los pólipos	52
3.2.2. Selección y agrupación de los candidatos a pólipos	58
3.2.3. Extracción de corte planar	60
3.2.4. Análisis selectivo	62
3.2.5. Pólipos detectados	65
CAPÍTULO 4: Pruebas y resultados.....	66
4.1. Descripción del ambiente de pruebas.....	66
4.2. Descripción de las pruebas.....	67
4.2.2. Tablas de resultados y discusiones	67
4.2.3. Sensibilidad y falsos positivos	69
CAPÍTULO 5: Conclusiones y trabajos futuros.....	73
5.1. Conclusiones.....	73
5.2. Trabajos futuros	74
Referencias	75
CAPÍTULO 6: Apéndice	79

Introducción

Hoy en día cientos de personas en el mundo mueren de cáncer de colon, siendo el cáncer el nombre genérico de un grupo de enfermedades que consiste en la división y reproducción anormal y sin control de las células del cuerpo. Particularmente en Venezuela según el Ministerio del Poder Popular para las Salud, en el año 2008 fallecieron 553 mujeres y 498 hombres a causa de esta enfermedad. [URL01].

La técnica para detectar tempranamente los pólipos que ha tomado auge en los últimos años es la colonoscopia virtual, la cual consiste en tomar tomografías del abdomen del paciente para hacer una reconstrucción tridimensional del colon de una forma no invasiva. Luego un especialista debe examinar estas imágenes para determinar la presencia de pólipos. Este proceso de inspección está sujeto a errores ya que es un proceso totalmente humano, es decir que depende de la capacidad del experto para detectar alguna anomalía, además toma mucho tiempo para analizar a cada paciente.

Recientemente se han investigado técnicas con el propósito de ayudar al especialista con la tarea de analizar los datos, resaltando rasgos de la imagen que hacen más fácil su interpretación o llegando incluso a mostrar directamente los pólipos, luego de realizar un análisis computacional de los datos obtenidos. De esta forma se espera disminuir la rata de errores y el tiempo necesario para analizar los datos de un paciente.

En este trabajo se propone un método que ayude al experto en la tarea del análisis de las tomografías y que además sea efectivo y rápido en la detección de pólipos colorrectales. Para ello se plantea el uso del hardware grafico paralelo de la GPU para acelerar los cálculos.

Objetivo General

Implementar un método de detección de pólipos con soporte en GPU a fin de minimizar el tiempo requerido para el análisis de las imágenes médicas y de esta manera ayudar al especialista en el estudio de los datos.

Objetivos específicos

- ✓ Desarrollar e implementar una técnica de detección de pólipos.
- ✓ Paralelizar las estructuras de datos y algoritmos necesarios para que puedan ser ejecutados en la tarjeta gráfica tomando en cuenta las limitaciones de la misma.
- ✓ Generar un archivo con las coordenadas de los candidatos potenciales a pólipos encontrados.
- ✓ Evaluar la aplicación desarrollada.

Alcance de este trabajo

- ✓ La aplicación soporta volúmenes de 16 bits por muestra, en formato crudo (raw).
- ✓ Para efectos de este trabajo se implementa un algoritmo sencillo de segmentación, para extraer el área del colon de la tomografía.
- ✓ La implementación paralela utilizará la librería CUDA que requiere de tarjetas gráficas con GPU de NVIDIA .
- ✓ Las coordenadas de los candidatos potenciales a pólipos se almacenan en un archivo de texto, incluyendo sus respectivas normales, para su posterior visualización.

CAPÍTULO 1: Marco teórico

Los pólipos de colon son precursores importantes del cáncer de colon. Éstos no son otra cosa que un crecimiento anormal de tejido a partir de una membrana mucosa en el intestino grueso (colon). En la Figura 1.1 se puede ver un ejemplo de pólipos dentro del colon.

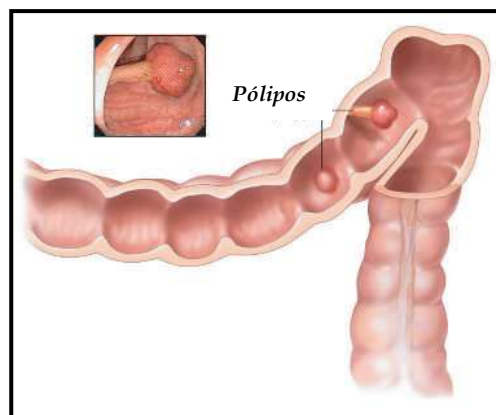


Figura 1.1: Pólipos en el colon.

El *colon* es un órgano del sistema digestivo que forma parte del intestino grueso. La función principal del colon es la formación y almacenamiento temporal de heces fecales. El colon no es esencial para la vida, ya que las personas sobreviven a la remoción del mismo, pero la falta de colon genera condiciones como: Pérdida excesiva de sodio y agua, deshidratación crónica, bajos niveles de sodio en la sangre y una baja producción de orine en los riñones. Las bacterias presentes en el colon juegan también un papel importante ya que pueden convertir algunos productos de desecho (como la urea) en productos que pueden reabsorberse (como el amonio), jugando como un rol en la nutrición. En la Figura 1.2 se puede observar el colon dentro del cuerpo humano y posteriormente en la Figura 1.3 se muestran en detalle las partes que lo componen.

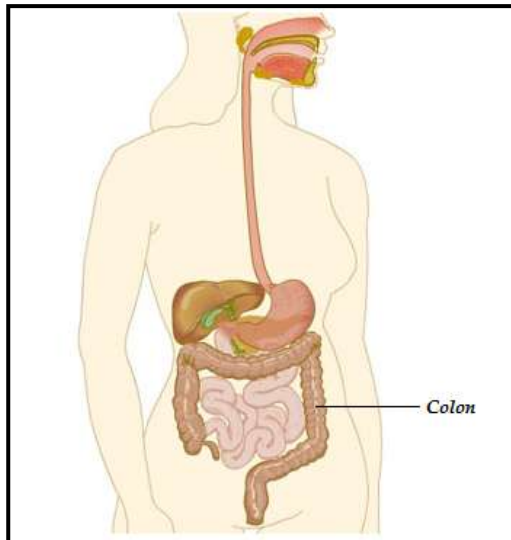


Figura 1.2: El colon en el cuerpo humano.

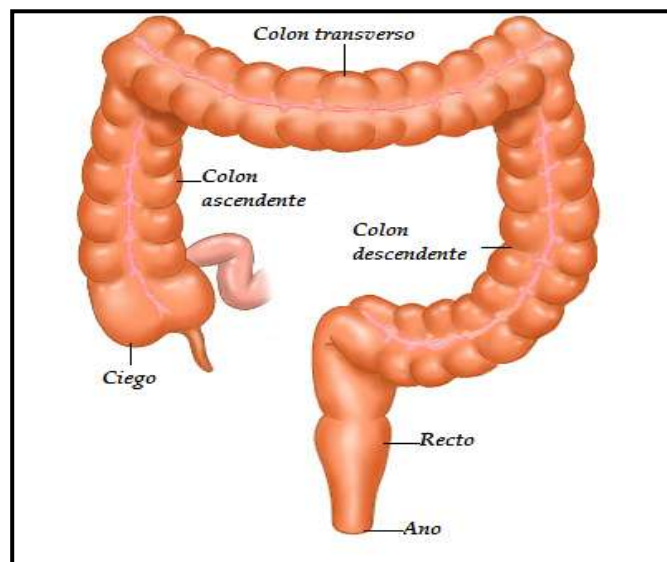


Figura 1.3: Las partes del colon.

En términos generales, los pólipos colorrectales pueden ser *epiteliales neoplásicos*, *epiteliales no neoplásicos* y *no epiteliales*¹. Los primeros, son los que presentan un mayor riesgo en convertirse en cáncer de colon. Sin embargo es imposible diferenciarlos de la mayoría de los pólipos no neoplásicos sin un análisis microscópico del mismo.

¹ La distinción entre epitelial y no epitelial indica si el pólipo es derivado o no del epitelio (en el caso del colon, el epitelio corresponde al revestimiento formado por tejido y mucosa en las paredes del mismo). Por otra parte, un pólipo es neoplásico si se produce por una proliferación anormal de células, mientras que la formación de un pólipo no neoplásico es de otra naturaleza (alteración en el proceso de renovación celular, agrupación desorganizada de células maduras o reparación de la mucosa posterior a una ulcera).

El proceso de degeneración de un pólipo en cáncer de colon suele tardar muchos años¹. No obstante, la detección tardía de los mismos disminuye la probabilidad de recuperación. Por esta razón, aunque no todos los pólipos degeneren en cáncer, es importante la detección precoz y la extracción de estos; es la manera más efectiva de prevenir el cáncer de colon.

Las personas de más de 50 años tienen más probabilidades que los jóvenes de tener cáncer de colon. Es por eso que los expertos recomiendan que todas las personas de 50 años o más que tienen un riesgo promedio de cáncer de colon (como antecedentes familiares) se hagan una prueba de detección para buscar pólipos de colon.

A pesar de los avances de la ciencia, la colonoscopia sigue siendo el método de detección de pólipos más usado en el mundo. Este tipo de examen puede resultar incómodo y algunas veces doloroso para las personas. En algunos casos la colonoscopia puede ser llevada a cabo bajo sedación, en la cual se aplican medicamentos intravenosos y el paciente entra en un estado que permite realizar el estudio en donde no experimenta molestias durante el procedimiento. Aún así, se ha hecho necesaria la utilización de métodos alternativos que resulten cómodos para garantizar además que las personas con riesgos de tener pólipos se hagan los exámenes pertinentes sin temor al dolor y demás implicaciones.

En los últimos años se ha hecho un esfuerzo para la detección temprana del cáncer de colon de una forma más económica, rápida, precisa e incluso menos invasiva e incómoda para el paciente. A través del uso de tomografías computarizadas es posible ahora inspeccionar el interior del colon de una manera no invasiva.

1.1. Adquisición de Imágenes Médicas del Colon

La adquisición de imágenes médicas se refiere a las técnicas y procesos usados para la creación de imágenes del cuerpo humano con propósitos clínicos. También se emplea frecuentemente para referirse a las técnicas no-invasivas utilizadas para producir imágenes del

¹ En el capítulo 23 del libro *Gastroenterología y Hepatología*, el Dr. Felipe Franco B. describe este proceso, afirmando que dura alrededor de unos 10 años.

cuerpo. Sin embargo, la adquisición de imágenes médicas también incluye la endoscopia, el cual es un conjunto de técnicas ‘mínimamente-invasivas’. Estas técnicas son llamadas así debido a que se llevan a cabo a través de una cavidad del cuerpo o a través de una abertura en lugar de realizar una cirugía abierta. Endoscopia significa ‘mirar adentro’, lo cual usualmente se refiere a mirar dentro del cuerpo por razones médicas utilizando un instrumento llamado Endoscopio¹ (ver Figura 1.4).

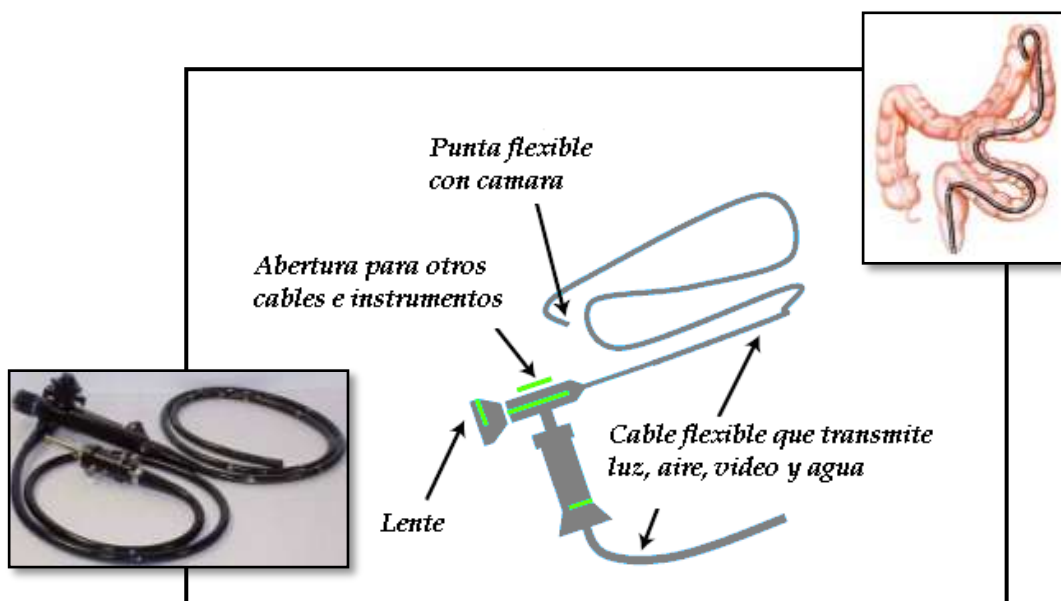


Figura 1.4: Ilustración de un endoscopio típico. El emplace izquierdo muestra la fotografía de un endoscopio común. En el emplace derecho se puede observar como el cable flexible navega por el colon.

Se pueden realizar dos tipos de endoscopia, una sigmoidoscopia o una colonoscopia, las cuales difieren en la extensión de la cavidad del colon que es examinada. La **sigmoidoscopia** es el examen mínimamente-invasivo del intestino grueso desde el recto hasta donde termina el colon descendente usando un endoscopio. Si algo inusual es encontrado en el recto o en el colon, como un pólipo o tejido inflamado, el médico puede tomar una muestra usando un instrumento insertado en el endoscopio. Luego, el médico envía la muestra de tejido al laboratorio para determinar con un microscopio si es cancerosa o no. La **colonoscopia**, también llamada **colonoscopia óptica**, es el examen mínimamente-invasivo del colon y la parte distal del intestino delgado con una cámara en un endoscopio flexible (también llamado **colonoscopia** en este examen) insertado a través del ano. Este examen provee un diagnóstico visual y provee la oportunidad de tomar una biopsia o remover una

¹ El **endoscopio** es un instrumento (producto sanitario) en forma de tubo, que contiene una luz y una óptica que permite la visualización del interior de un órgano hueco o una cavidad corporal, introduciéndolo mediante un agujero natural o una pequeña incisión quirúrgica.

lesión sospechosa. La colonoscopia es similar pero no igual a una sigmoidoscopia. La sigmoidoscopia le permite al médico inspeccionar solo la parte final del colon, mientras que la colonoscopia le permite examinar el colon completo. Con frecuencia una sigmoidoscopia es usada como una exploración previa a la colonoscopia.

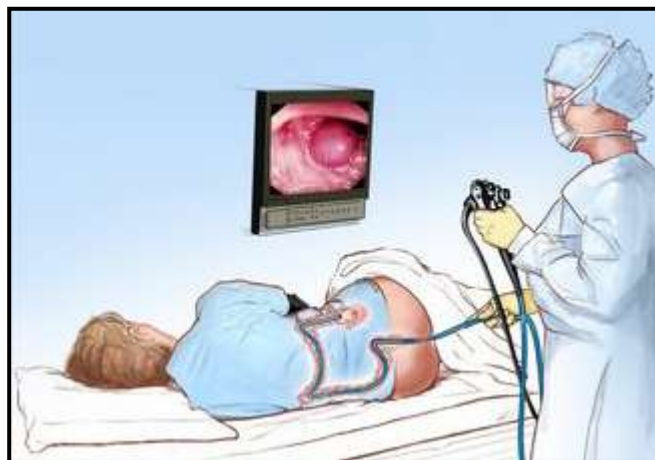


Figura 1.5: Imagen de una colonoscopia tradicional.

La Figura 1.5 muestra la forma como se lleva a cabo una colonoscopia tradicional u óptica para obtener una vista del interior del colon.

El **Enema de Bario** se utiliza para resaltar el colon y el recto. Es otra técnica mínimamente-invasiva en la cual un tubo de enema bien lubricado es insertado en el recto (ver Figura 1.6). El bario¹ es inyectado a través de este tubo en dirección al colon y el recto [URL02]. Un pequeño balón en la punta del tubo es inflado para mantener el bario en el interior. Se cambiará su ubicación varias veces para garantizar que el bario cubra las paredes del colon y del recto. Después, se inyecta una pequeña cantidad de aire dentro del colon. Se toma una serie de radiografías, seguidamente se extrae el tubo de enema y se le indicará que vaya al baño para expulsar el bario y el aire.

En un colon sano, el bario debe llenar el colon uniformemente. El intestino debe mostrar un contorno normal y debe lucir libremente abierto. Los Enemas de Bario son usados frecuentemente para comprobar la salud del intestino. Un pólipo pre-canceroso o canceroso puede detectarse si se

¹ El **bario** es un elemento químico de la tabla periódica cuyo símbolo es Ba y su número atómico es 56. Metal alcalinotérreo, el bario es el 14º elemento más común, ocupando una parte de 2.000 de la corteza terrestre. Su masa atómica es 137,34. Su punto de fusión está a 725 °C, su punto de ebullición a 1.640 °C, y su densidad relativa es 3,5. Reacciona con el cobre y se oxida rápidamente en agua.

observa alguna anomalía en las radiografías. Los pólipos pueden ser observados, pero no removidos durante el examen.

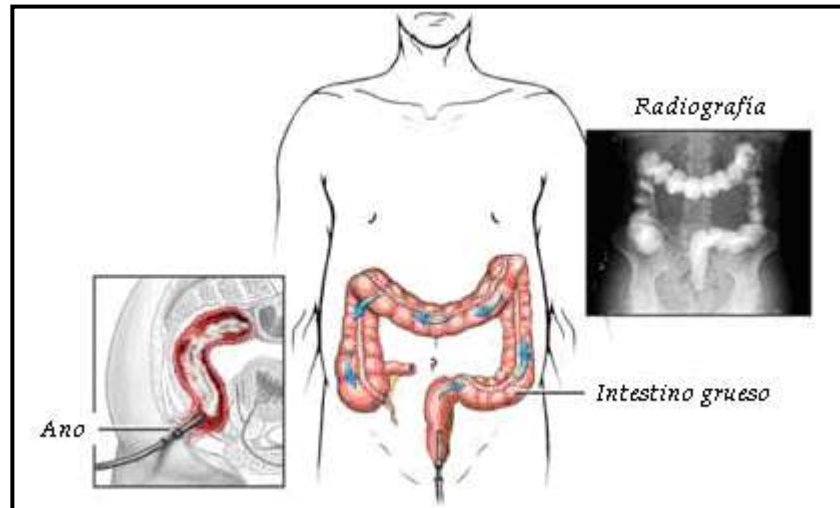


Figura 1.6: Enema de bario.

La Tomografía Computarizada (TC), originalmente conocida como **tomografía axial computarizada (TAC)**, es un método de adquisición de imágenes médicas. La TAC es una tecnología de exploración de rayos X que produce imágenes detalladas de cortes axiales del cuerpo. En lugar de obtener una imagen como la radiografía convencional, la TAC obtiene múltiples imágenes al rotar alrededor del cuerpo. Una computadora combina todas estas imágenes en una imagen final que representa un corte del cuerpo como si fuera una rodaja. La máquina de TAC crea múltiples imágenes en rodajas (cortes) de la parte del cuerpo que está siendo estudiada. Cada imagen de TC resultante de un barrido consiste en una matriz de valores de intensidades del área barrida expresadas en Unidades Hounsfield¹ (UH). Los barridos de TC del colon son usados para un procedimiento llamado *Colonografía de TC*. El procedimiento consiste en inflar el colon con aire o anhídrido carbónico y luego realizar un barrido de TC del abdomen y pelvis del paciente, para luego transferir las imágenes resultantes a un computador para su procesamiento y análisis. Estas imágenes deben adquirirse preferiblemente mientras el paciente retiene la respiración, para evitar artefactos en la imagen debido al movimiento. Los tomógrafos modernos hacen esto perfectamente posible, ya que son capaces de barrer largas secciones del cuerpo con alta resolución en pocos segundos.

¹ **Unidad Hounsfield:** información numérica contenida en cada píxel en la tomografía computarizada. Se utiliza para representar la densidad del tejido.

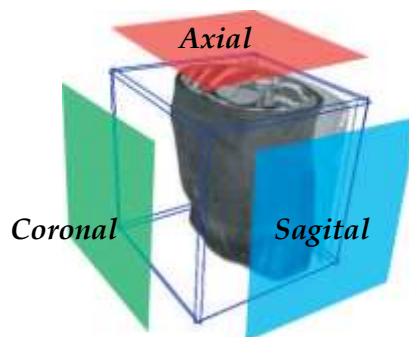


Figura 1.7: Planos tridimensionales del cuerpo.

Una vez que el barrido de TC es transferido al computador, la estrategia más sencilla para explorarlo y analizarlo es el *slicing*, una técnica en la que el médico explora el barrido mediante la visualización de cada uno de los cortes, uno por uno. Sin embargo, la mayoría de las estaciones de trabajo actuales puede procesar el barrido de TC para producir una Reconstrucción Multi-Planar (RMP) de las imágenes. La técnica de RMP consiste en desplegar las imágenes producidas por el proceso original de reconstrucción, en una orientación distinta a la de las imágenes originales. Si, por ejemplo, las imágenes de TC originales son axiales, es posible producir una vista sagital y una vista coronal de los datos. Usando RMP, las estaciones de trabajo actuales pueden desplegar las imágenes en dirección axial, coronal, sagital y cualquier otra orientación (ver Figura 1.7). Por lo tanto, el barrido de TC puede ser explorado completamente en cualquiera de los planos, usando la técnica de *slicing*. Sin embargo, el procedimiento completo de colonografía de TC puede incluir el uso de técnicas de Despliegue de Volúmenes (DV) para proveer un examen virtual del colon llamado Colonoscopia Virtual (CV).



Figura 1.8: Máquina para TAC (Tomografía Axial Computada).

La Figura 1.8 muestra una fotografía de una máquina para TAC helicoidal con un programa de reconstrucción 3D y programa de Colonoscopia virtual.

La **Resonancia Magnética (RM)** es un método no-invasivo usado para obtener imágenes tridimensionales del interior del cuerpo. Similar a la TC, se producen una serie de imágenes llamadas cortes de RM. Sin embargo, RM es la mejor para adquirir imágenes de tejidos sin calcificaciones. En general, la RM es superior a la TC al distinguir las diferencias entre dos tejidos similares pero no idénticos, lo que contribuye a la delineación precisa de la extensión de los tumores. Sin embargo, un barrido de RM puede tomar varios minutos en ser adquirido, provocando la pérdida de la calidad de la imagen debido a artefactos causados por el movimiento del paciente.

El Examen Volumétrico Interpolado con Respiración Retenida (EVIRR) es una Tecnología de barrido de RM relativamente nueva que permite adquirir las imágenes en un tiempo reducido sin degradar la calidad de la imagen. La colonografía de RM es un método aún más reciente que la colonografía de TC y por lo tanto no se ha definido aún un protocolo final. Las primeras investigaciones describen barridos de EVIRR que cubren la completitud del colon relleno de enema en el plano coronal. Hasta ahora se han utilizado el aire y el anhídrido carbónico como agentes de distensión colónica, aunque los enemas líquidos también pueden utilizarse. Actualmente, tres diferentes tipos de enema líquido se han utilizado: el enema brillante, el enema oscuro y el etiquetado fecal. Si se utiliza el enema brillante, la cavidad del colon aparece brillante en las imágenes y los pólipos se ven como defectos en el relleno. Utilizando el enema oscuro, la cavidad de colon aparece oscura en las imágenes y se puede ver una mejora en las paredes del colon, haciendo posible observar pólipos y cambios inflamatorios. Cuando se utiliza el etiquetado fecal, el paciente ingiere un medio de contraste para darle a la materia fecal un valor de intensidad distinto a los pólipos y las paredes del colon, y frecuentemente se remueve de las imágenes para una mejor interpretación de las mismas. De manera similar a la colonografía de TC, la técnica más sencilla para explorar la colonografía de RM es la técnica de *slicing*, aunque la mayoría de las estaciones de trabajo puede producir una RMP también.

1.2. Colonoscopia virtual

La técnica alternativa que ha tomado auge en los últimos años es la colonoscopia virtual, la cual consiste en tomar tomografías del abdomen del paciente para hacer una reconstrucción tridimensional del colon de una forma no invasiva. Luego un especialista debe examinar estas imágenes para determinar la presencia de pólipos.

La Colonoscopia Virtual (CV) es una técnica en la que se produce una recreación tridimensional de las paredes internas del colon para así explorarlas en búsqueda de posibles lesiones. La idea es básicamente la simulación de una colonoscopia óptica, por esto, su nombre de colonoscopia virtual.

Originalmente, la entrada al procedimiento de CV son los datos de TC producidos durante un examen de colonografía de TC, pero actualmente se pueden utilizar también imágenes de RM, correspondientes a una colonografía de RM, luego que las imágenes son producidas son transferidas a una estación de trabajo con software especializado en colonoscopia virtual. El software carga las imágenes de entrada y las procesa, produciendo imágenes correspondientes a la parte interna del colon desde el punto de vista de un colonoscopio. Con la ayuda del software de CV, el médico analiza las imágenes de CV y produce un reporte médico.

Una CV no muestra una imagen del colon completo sino lo que puede ser visto por el colonoscopio “virtual”. Este puede ser trasladado dentro del colon, tal como el colonoscopio real en una colonoscopia óptica. Cada vez que cambia de posición la imagen de CV es actualizada para simular el efecto de mover un colonoscopio real. A diferencia de una colonoscopia tradicional u óptica, en la colonoscopia virtual no es necesario moverse manualmente; este movimiento a través del colon puede ser calculado automáticamente a partir de las imágenes de entrada para así realizar una navegación virtual siguiendo un camino calculado. Esto se conoce como *navegación virtual o vuelo virtual*. El camino aproxima el eje central del esqueleto unidimensional del colon y usualmente es llamado *línea central* porque atraviesa el colon desde un extremo hacia el otro en una trayectoria que es central a la estructura del colon. La Figura 1.9 ilustra como se ve la línea central del colon. Como las imágenes de entrada usualmente contienen el colon y el recto es posible entonces calcular

una línea central a través del intestino grueso completo, cubriendo la misma superficie que una colonoscopia óptica.

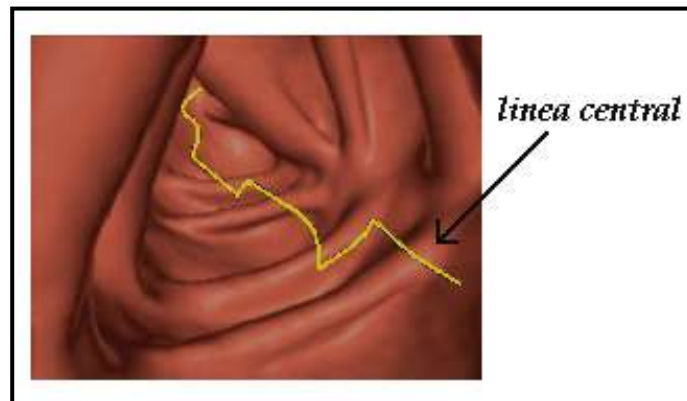


Figura 1.9: Línea central del colon (línea amarilla).

Algunos sistemas también proveen una vista tridimensional del exterior del colon, indicando la posición actual del colonoscopio virtual para proveer una idea tridimensional precisa de la posición del colonoscopio dentro del colon. Las estaciones de trabajo de CV permiten al médico detener el vuelo virtual y mover el colonoscopio virtual libremente dentro del colon, o simplemente mover el colonoscopio virtual a una posición específica dentro del colon. En la Figura 1.10 podemos observar tanto la navegación interna como la vista tridimensional exterior del colon.

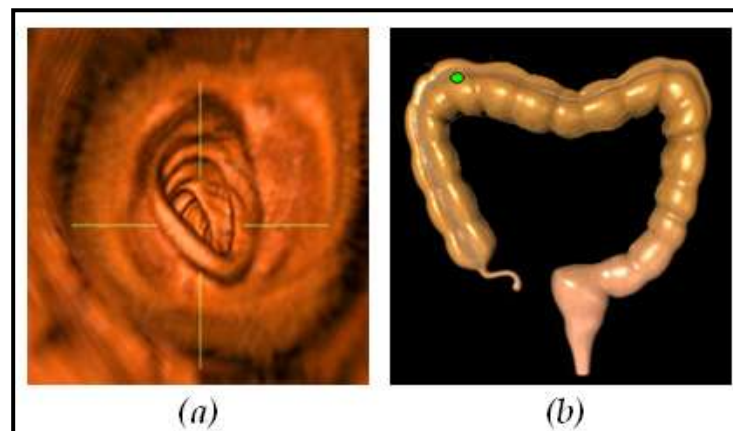


Figura 1.10: (a) vuelo virtual. (b) vista exterior del colon (el punto verde indica la posición del colonoscopio virtual).

El fluido residual limita la colonoscopia virtual porque este luce como parte de la pared del colon en el vuelo virtual, debido a su alto contraste. Por lo tanto, el fluido residual

puede esconder pólipos; este fluido puede ser removido por el software de CV luego de que las imágenes de TC son cargadas y antes de que se calcule la línea central. Se emplea un procedimiento llamado *limpieza electrónica* el cual limpia el colon para el vuelo virtual y puede revelar pólipos escondidos bajo el fluido residual. La limpieza electrónica tiene la ventaja de limpiar el colon para el vuelo virtual, pero puede producir artefactos en las imágenes que pueden parecer falsos pólipos llevando a un diagnóstico equivocado. En el caso de la colonografía de RM la cantidad de fluido residual es de poca importancia, porque no afecta las imágenes obtenidas. La mayor desventaja de la CV es que no es posible realizar una biopsia, y por lo tanto una colonoscopia óptica debe realizarse si hay una sospecha de cáncer.

1.2.1. Preparación para el examen de colonoscopia virtual

La preparación puede variar dependiendo del médico. Toda persona que se someta a cualquier tipo de colonoscopia tiene que vaciar completamente los intestinos antes del examen, lo cual se puede hacer utilizando un enema o laxantes combinados con una dieta líquida el día antes del procedimiento. Se deben evacuar completamente los intestinos hasta que no quede ninguna materia sólida. La persona debe asegurarse de comentarle al médico acerca de cualquier medicamento que esté tomando y se le dirá cuáles puede tomar y cuáles debe suspender temporalmente. Los escáneres para IRM (Imágenes por Resonancia Magnética) y TC son muy sensibles a los metales, por lo tanto, no se deben llevar joyas puestas el día del examen. Para el procedimiento, a la persona se le solicitará que lleve puesta una bata hospitalaria.

1.2.2. Forma en que se realiza un examen de colonoscopia virtual

Los pacientes tienen que seguir un procedimiento estricto de evacuación intestinal el día anterior, igual que lo harían para una colonoscopia estándar [URL03]. Este examen se realiza en la sala de radiología de un hospital o un centro médico y, a diferencia de la colonoscopia convencional, no se requieren sedantes y no se utiliza ningún colonoscopio.

En principio, la persona se acuesta sobre el lado izquierdo en una mesa estrecha que está conectada a una máquina para IRM o TC y las rodillas se doblan contra el pecho. Se inserta un tubo

pequeño y flexible dentro del recto y se insufla¹ aire través de dicho tubo para que el colon esté más grande y sea más fácil de visualizar. Después de hacer esto, se le pide a la persona que se acueste boca arriba.

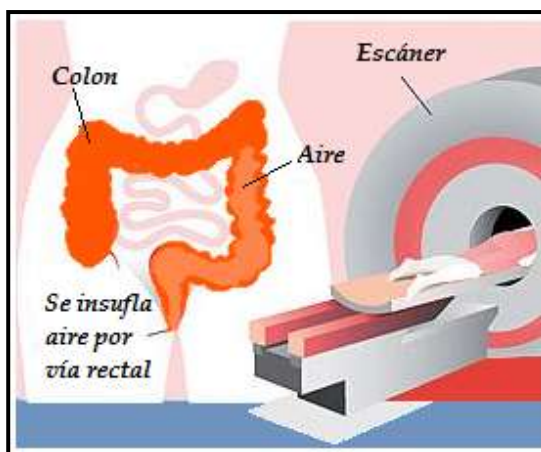


Figura 1.11: Colonoscopia virtual.

Seguidamente, se desliza la mesa dentro de un túnel grande a través de la máquina, donde los rayos X crean imágenes del colon. La persona debe permanecer bastante inmóvil durante este procedimiento, dado que los movimientos pueden hacer que las imágenes salgan borrosas. Es posible que se le solicite a la persona contener el aliento brevemente mientras se toma cada imagen. Finalmente, una computadora combina todas las imágenes para formar imágenes tridimensionales del colon, las cuales se observan en un monitor de video. Las imágenes también se toman mientras la persona está boca abajo. La Figura 1.11 muestra bastante bien el procedimiento anteriormente descrito. El examen dura aproximadamente 20 minutos.

1.2.3. Cuáles son los riesgos

- ✓ Exposición a la radiación.
- ✓ Los medicamentos de preparación para el examen pueden provocar náuseas, vómitos, distensión o irritación rectal.

¹ Introducir, a soplos o inyectados, un gas, un vapor o una sustancia en polvo dentro de una cavidad u órgano del cuerpo.

1.2.4. Consideraciones especiales

Las diferencias entre la colonoscopia virtual y la convencional son, entre otras [URL03]:

- ✓ Ambos procedimientos pueden localizar pólipos que han alcanzado un tamaño grande y preocupante. Sin embargo, la colonoscopia virtual no es tan detallada como la colonoscopia convencional y puede pasar por alto pólipos de menos de 10 milímetros de diámetro o lesiones planas.
- ✓ Si se encuentra un pólipo con la colonoscopia virtual, el paciente tiene que someterse a una colonoscopia convencional para eliminar la anomalía. La colonoscopia convencional permite la extirpación inmediata de pólipos.
- ✓ Con la colonoscopia virtual, se puede visualizar el colon desde muchos ángulos diferentes, lo cual no es fácil con la colonoscopia convencional.
- ✓ En la colonoscopia virtual, no se utilizan sedantes y los pacientes, en general, pueden reanudar sus actividades normales inmediatamente después del examen. La colonoscopia convencional implica el uso de sedantes y usualmente amerita un día de reposo.

La colonoscopia virtual es una de las herramientas de detección sistemática para cáncer de colon recomendadas por la Sociedad Estadounidense para el Cáncer (*American Cancer Society*). En la Figura 1.12 se puede observar claramente una vista del colon mediante colonoscopia virtual y otra de colonoscopia tradicional u óptica.

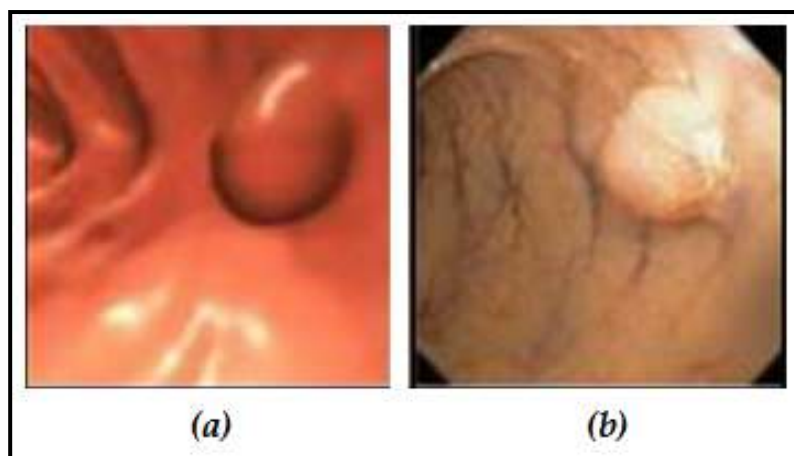


Figura 1.12: Ejemplos del colon visto desde el interior usando. (a) Colonoscopia virtual.
(b) Colonoscopia tradicional.

1.3. Procesamiento de datos

Una vez adquirida la imagen tridimensional del colon a través de un barrido de TC o RM, correspondientes a una colonografía de TC o una colonografía RM, respectivamente, se procede a realizar el procesamiento de los datos de la misma.

El procesamiento de datos comprende diversas etapas [COT08]; inicialmente la **carga de datos** que consiste en leer cada uno de los archivos y cargar los datos de la imagen en una estructura de datos en la memoria de la estación de trabajo de CV. El limpiador electrónico solo es necesario en la colonografía de TC si se utiliza etiquetado fecal durante la preparación del paciente. La forma más simple de remover el fluido residual de las imágenes TC es la umbralización¹, removiendo así todas las regiones con altos valores de intensidad como el fluido residual. Sin embargo, la umbralización también remueve secciones del hueso y no puede tratar el Efecto de Volumen Parcial (EVP), un artefacto en las imágenes producido en el borde entre dos regiones con valores de intensidad muy diferentes, el cual produce efectos de escalonado en el borde interno del colon. Un método más veloz para el limpiador electrónico es el *rayo de segmentación* que se describe en [COT08] y [LAK00], el cual puede remover el fluido residual así como tratar el EVP, reparando los bordes de aquellas regiones con intensidades muy diferentes.

Para desplegar la vista externa del colon es necesario segmentar la pared interna del mismo. Dicha segmentación también es necesaria para determinar la línea central del colon, detectar pólipos y optimizar la generación de las imágenes de CV. La **segmentación del colon** puede hacerse de diversas formas, por ejemplo con el algoritmo de Región Creciente basado en valores de intensidad, métodos estadísticos que usan Campos Aleatorios de Markov, Modelos deformables, entre otros. Las técnicas de región creciente son las más rápidas por lo tanto las más apropiadas para la práctica clínica pero dependen de umbrales muy estrictos que pueden variar entre diferentes barridos y diferentes escáneres [COT03].

¹ La **umbralización** permite etiquetar la imagen resultante de los procesos de filtrado del ruido y realce de los bordes a fin de obtener los píxeles en los que se produce un borde.

Una vez finalizada la segmentación del colon, es posible iniciar la etapa de **planificación de ruta**. La planificación de ruta es la base de una navegación eficiente y es bien sabido que una buena ruta puede mejorar la detección de pólipos. El objetivo es encontrar un camino que facilite la navegación dentro del colon virtual. Por lo tanto, la ruta debe permanecer dentro del colon en todo momento, evitar colisiones con la pared del colon, proveer la mejor vista posible y ser tan suave como sea posible. Se han desarrollado varias técnicas que tratan de cumplir con estos criterios; algunas de ellas son demasiado costosas en cómputo para ser consideradas. Idealmente, la ruta debe permitirle al usuario explorar el colon en su completitud en un solo vuelo virtual. Sin embargo, frecuentemente es necesario calcular más de una ruta debido a la *desconexión del colon*. Esto es, que la etapa de segmentación del colon no logró segmentar el colon completo sino que se obtuvieron varias secciones desconectadas del mismo; esto puede ocurrir debido a una falla en el limpiado electrónico o la segmentación del colon. Un método completamente automático para el obtener la ruta se propone en [WAN02], el cual es capaz de combinar secciones desconectadas del colon.

Finalmente sigue la etapa de **detección de pólipos**; en esta etapa se detectan los candidatos a pólipos y posteriormente se eliminan aquellos que fueron detectados erróneamente como pólipos (falsos pólipos). La detección de pólipos será discutida en detalle en el capítulo 2 de este documento.

1.4. Visualización y navegación de los datos

Antes de hablar del proceso de detección de pólipos es importante tomar en cuenta el despliegue de las imágenes que se van a utilizar. En el contexto médico, cuando se habla de despliegue de imágenes, la calidad de las mismas es la principal inquietud ya que es poco probable que un médico acepte un sistema si los resultados que arroja están basados en análisis de imágenes poco confiables. El objetivo es el de proyectar los datos tridimensionales en el espacio bidimensional de la pantalla 2D de una forma que sea fiel a los datos.

Existen dos formas de llevar a cabo el despliegue de las imágenes, una de ellas es *el despliegue de superficie* de un mallado de triángulos que aproxime la superficie del colon. La otra forma es desplegar los datos tridimensionales usando técnicas de *despliegue de volúmenes*. La primera de ellas puede ser más rápida, pues el hardware gráfico está optimizado para desplegar

geometrías; sin embargo requiere de un proceso previo de reconstrucción de la superficie, en donde se ignoran detalles presentes en otras capas del tejido del colon. Cuando se utiliza el despliegue de volúmenes, típicamente se utiliza *ray-casting*¹ perspectivo [WAN00] [YOU97] para producir imágenes de alta calidad. La técnica de ray-casting suele ser costosa computacionalmente, pues puede requerir información adicional a la superficie en el despliegue. Sin embargo, existen numerosas técnicas de aceleración que permiten producir imágenes de ray-casting a velocidades interactivas. [YOU97][WAN99][KAU01].

La navegación dentro del colon virtual es un aspecto muy importante del despliegue de las imágenes. El paradigma de navegación utilizado determina la usabilidad del sistema de colonoscopia virtual. Las técnicas hoy existentes pueden clasificarse en tres clases: *Navegación automática*, *navegación manual o libre* y *navegación guiada*. En la navegación automática se genera una animación fuera de línea de vuelo virtual y luego se calcula la ruta de navegación. Este tipo de navegación reduce la interacción a la funcionalidad de un reproductor de video por lo que requiere, si las estructuras de interés no son cubiertas correctamente, recalculer la animación y la ruta de la cámara. La principal ventaja de la navegación automática es que no requiere una técnica de despliegue interactiva, ya que no hay interacción del usuario que provoque retraso en la navegación. La navegación manual o libre es otra opción popular; desafortunadamente la complejidad de las estructuras anatómicas comúnmente encontradas en las imágenes de colonografía es bastante alta por lo que es difícil la navegación. Además la prevención de colisiones requiere usualmente operaciones costosas. A diferencia de la navegación automática, este tipo de navegación requiere una técnica de despliegue altamente interactiva ya que los retrasos en el despliegue de la imagen en respuesta a la interacción del usuario dificultan la navegación. Finalmente la navegación guiada permite una navegación altamente flexible, combinada con la guía del usuario a través de una ruta de navegación y un esquema eficiente de prevención de colisiones. Este tipo de navegación se beneficia de un despliegue interactivo debido a que la interacción del usuario juega un papel importante en el control de la cámara.

¹ **Ray-casting** es una técnica que consiste en trazar rayos desde el ojo a través de cada pixel de una imagen. Ray-casting es una técnica para el despliegue de volúmenes 3D. Esta consiste en lanzar rayos desde el ojo a través de los vóxeles del volumen y acumular el color de cada pixel de acuerdo al color y transparencia de cada vóxel atravesado por el rayo.

Durante la navegación del colon puede que el usuario requiera emplear herramientas de diagnóstico, como medir un pólipo en 3D, determinar el espesor de una pared del colon, u otra funcionalidad que le sea útil. Luego de realizado el diagnóstico, las herramientas de reporte le permiten al usuario transmitir el diagnóstico a los receptores (como al cirujano que va a remover los pólipos).

Una vez calculada la ruta de navegación, los datos son inspeccionados por un experto para así determinar la posible presencia de pólipos o de cáncer de colon. Este proceso de inspección está sujeto a errores ya que es un proceso totalmente humano, es decir que depende de la capacidad del experto para detectar alguna anomalía, además toma mucho tiempo para analizar a cada paciente.

A fin de ayudar al experto en el análisis de los datos, se han desarrollado técnicas de detección automática de candidatos a pólipos. A continuación se explicarán algunas de estas técnicas así como otros aspectos importantes para la comprensión del método propuesto en el presente trabajo.

CAPÍTULO 2: Antecedentes

A partir del nacimiento de la colonoscopia virtual, son muchas las investigaciones que se han llevado a cabo con el propósito de desarrollar nuevas técnicas para la detección automática de pólipos asistida por el computador, así como para incluir mejoras en dichas técnicas. Por otra parte, los avances realizados en otras áreas de interés del procesamiento digital de imágenes, y la incorporación de nuevo hardware gráfico, que permite ejecutar procesos paralelos de propósito general en múltiples núcleos de procesamiento, son elementos que pueden contribuir a mejorar los resultados de dichas técnicas de detección así como disminuir considerablemente el tiempo de respuesta de las mismas.

2.1. Técnicas de detección de pólipos

Como lo mencionamos anteriormente, en la actualidad existen numerosas técnicas para la detección de pólipos en datos volumétricos. Estas técnicas en general suelen implementar dos fases [NAV08]:

- *Fase de detección de candidatos a pólipos:* En esta primera fase se detectan las estructuras que pueden ser pólipos, aunque generalmente se detectan erróneamente muchas estructuras que en realidad no lo son debido a limitaciones del detector.
- *Fase de eliminación de falsos positivos:* Consiste en eliminar los falsos positivos, es decir aquellos casos en que se detecta un supuesto pólipo que realmente no lo es. A pesar de que fueron obtenidos como candidatos en la fase anterior, éstos se eliminan generalmente usando métodos estadísticos o de clasificación. Algunas veces esta fase está mezclada con la primera y es difícil distinguirla como una fase separada.

A continuación se presentan algunas de las técnicas hasta ahora expuestas para la detección de pólipos.

2.1.1. Detección de pólipos basada en curvatura

Existen varios métodos propuestos basados en curvatura para detectar pólipos. La mayoría de estos extraen la superficie de la pared del colon para realizar los cálculos. Sin embargo, aunque estos métodos funcionaron bien en conjuntos de datos con pólipos simulados, cuando fueron aplicados a casos clínicos reales presentaron un elevado incremento en la tasa de falsos-positivos o bien un descenso en la sensibilidad⁹.

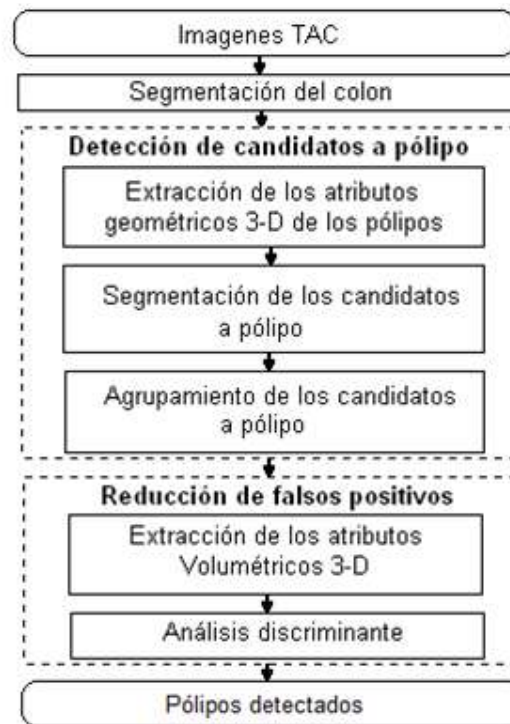


Figura 2.1: Esquema propuesto por Yoshida y Nappi.

En el año 2001 Yoshida y Nappi [YOS01] lograron buenos resultados aplicando un método para la detección automática de pólipos que, a diferencia de otros métodos basados en curvatura,

⁹ **Sensibilidad** se refiere a la probabilidad de clasificar correctamente a un individuo enfermo, es decir, la probabilidad de que para un sujeto enfermo se obtenga en la prueba un resultado positivo.

extrae una región gruesa de la pared del colon y no sólo la superficie. Esto permite extraer el pólipo completo (incluyendo su estructura interna) lo cual ayuda a reducir los falsos-positivos. El esquema propuesto por Yoshida y Nappi se muestra en la Figura 2.1. A continuación se describe cada una de las fases:

- **Segmentación del colon.** La segmentación del colon consta de 2 fases:

Fase de extracción basada en anatomía: En esta fase se aplica suavizado de Gauss, el análisis de histograma, umbralización, detección de bordes, región creciente, análisis de componentes conexas y la morfología matemática. El resultado es una segmentación con una alta sensibilidad, pero puede contener estructuras extra-colónicas.

Fase de análisis basado en el colon: Primero se hace un análisis de componentes conexas para identificar el recto en el volumen segmentado en la primera fase. Después se aplica un algoritmo de volumen creciente y se realizan una serie de chequeos (como la simetría del colon ascendente y descendente y posición del recto) para verificar que el colon se ha segmentado correctamente.

- **Detección de candidatos a pólipo.** Consta de tres fases que se muestran a continuación:

Fase de extracción de los atributos geométricos 3-D de los pólipos: Para caracterizar a los pólipos, se calculan dos atributos geométricos para cada vóxel. Uno de ellos es el índice de la forma volumétrica y el otro es el índice de curvatura. Sus respectivas formulas son:

$$SI(p) = \frac{1}{2} - \frac{1}{\pi} \arctan \frac{\kappa_1(p) + \kappa_2(p)}{\kappa_1(p) - \kappa_2(p)}$$

$$CV(p) = \sqrt{\frac{\kappa_1(p)^2 + \kappa_2(p)^2}{2}}$$

donde $\kappa_1(p)$ y $\kappa_2(p)$ representa las curvaturas principales en la vecindad del punto p . El índice de la forma volumétrica indica el tipo de forma, tal como se ve en la Figura 2.2. El índice de curvatura indica que tan suave es la curva de la superficie.

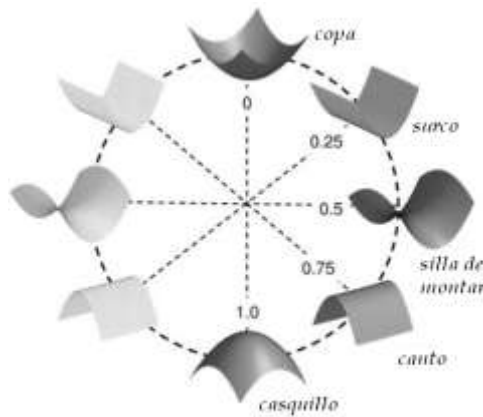


Figura 2.2: Relación de superficies comunes con los valores de índice de forma.

Fase de segmentación de los candidatos a pólipo: Para realizar la segmentación de los vóxeles candidatos a pólipo se realiza una umbralización con histéresis¹⁰. Se definen los valores máximo y mínimo del índice de forma volumétrica e índice de curvatura para seleccionar los vóxeles que servirán como semilla. Luego se definen nuevos valores máximo y mínimo y se hallan los vóxeles dentro de este rango que se encuentren conectados con las semillas.

Fase de agrupamiento de los candidatos a pólipo: Se define una distancia de combinación de manera tal de agrupar los vóxeles que se encuentren a una distancia menor que esta. Luego se aplica un algoritmo de c-medias difusas (*fuzzy c-means algorithm*) para eliminar los candidatos a pólipo que fueron marcados debido al ruido.

- **Reducción de falsos positivos.** Se divide en 2 etapas:

Extracción de los atributos Volumétricos 3D: Se determinan seis atributos para cada vóxel de cada grupo. Estos atributos son: El índice de forma, índice de curvatura, valor de TC, magnitud del gradiente del valor de TC, la concentración del gradiente y la concentración del gradiente direccional. Luego, por cada grupo obtenido en la fase de agrupamiento se obtienen 9 valores estadísticos de estas variables que son: Media, valor máximo, valor mínimo, varianza,

¹⁰ Una **umbralización con histéresis** o histéresis de umbral (Hysteresis thresholding) utiliza 2 umbrales. Primero uno más restrictivo para realizar una primera selección. Luego se realiza un ciclo en el cual se usa un segundo umbral menos restrictivo para seleccionar elementos cercanos a los de la primera selección.

asimetría estadística (skewness), curtosis¹¹, entropía, contraste y la media aritmética de los 10 valores máximos.

Análisis discriminante: Se utiliza una fórmula de análisis discriminante lineal o una fórmula de análisis discriminante cuadrática cuyos parámetros se obtienen de un conjunto de datos de entrenamiento en el cual los pólipos y los falsos positivos son conocidos. Aplicando el análisis discriminante se obtienen dos clases, una con los falsos positivos y otra con los pólipos.

En el 2008, Huang et al. [HUA10] investigaron el planteamiento del Frente de Pareto¹² para evaluar y mejorar los algoritmos de región creciente basado en curvatura, comúnmente usados para la detección de candidatos a pólipos. En principio, implementan dos tipos de algoritmos de región creciente basada en curvatura para localizar los candidatos a pólipos y los comparan. El primero de ellos, llamado AR, agrupa los vértices que cumplan con un conjunto de criterios basados en curvatura, en una superficie de malla triangular. En el segundo algoritmo, denominado RA+RB, los vértices se agrupan según dos conjuntos separados de criterios basados en curvatura.

A fin de ser coherentes con las convenciones del análisis frente de Pareto, Huang et al. [HUA10] describen el rendimiento de un algoritmo de detección de candidatos a pólipo como: Tasa de falsos negativos (FN) y de falsos positivos (FP). El número de pólipos verdaderos localizados en el colon que no fueron detectados por CAD (diseño asistido por computadora) se define como FN. El número de detecciones localizadas en superficies del colon sano se define como FP. No hay una única solución que optimice el rendimiento en ambos, las soluciones generalmente aceptadas de un problema multi-objetivo se definen como frente de Pareto.

A partir del análisis, sugieren un nuevo algoritmo de región creciente de dos pasos diferente al diseño original de Yoshida y Nappi [YOS01] en dos aspectos principales. En primer lugar, el diseño original utiliza un conjunto de criterios más selectivos en el primer paso para formar puntos de semillas para la detección del tope de los pólipos y utiliza criterios menos selectivos en el segundo paso para determinar el cuerpo de los pólipos. En segundo lugar, el diseño original utiliza un paso

¹¹La **curtosis** es una medida de la forma o apuntamiento de las distribuciones. Así las medidas de curtosis (también llamadas de apuntamiento o de concentración central) tratan de estudiar la mayor o menor concentración de frecuencias alrededor de la media y en la zona central de la distribución.

¹²La eficiencia de Pareto es aquella situación en la cual se cumple que no es posible beneficiar a más elementos de un sistema sin perjudicar a otros. El conjunto de decisiones que son eficientes según Pareto son llamadas **Frente de Pareto**.

adicional para combinar detecciones fragmentarias. Había dos razones principales para diseñar un nuevo algoritmo de dos criterios. En primer lugar, el paso de la fusión requiere la información de la ubicación del vértice. Esto aumentaría significativamente el tiempo de cálculo debido a que los cálculos de distancia requieren mucho tiempo. En segundo lugar, querían saber si en el algoritmo de detección de pólipos podía ser ajustado de forma automática por el enfoque de Frente de Pareto si se descartaban condiciones innecesarias. Al eliminar las relaciones de selectividad entre el primer grupo de criterios y el segundo, encontraron que más de 85% de las soluciones del Frente de Pareto habían hecho un primer criterio menos selectivo y no se requería realizar la fusión en la mayoría de los casos. En la Figura 2.3 mostrada a continuación se pueden observar los resultados de esta técnica basada en el Frente de Pareto.

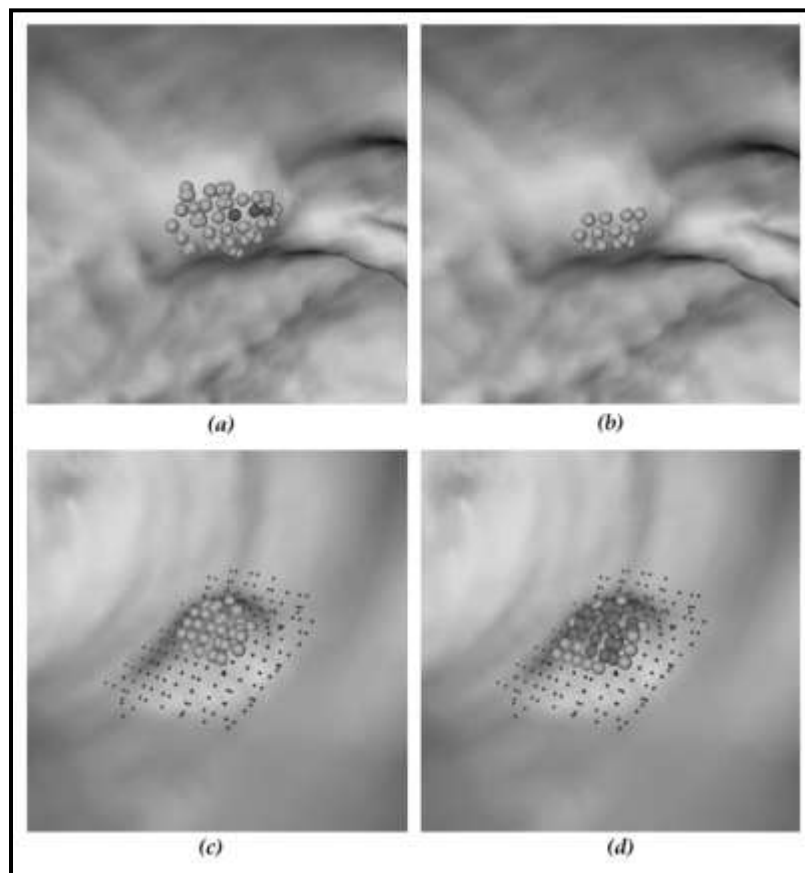


Figura 2.3: (b,c) Grupo de vértices resultantes utilizando el algoritmo de AR.

(a,d) Grupo de vértices resultantes utilizando RA+RB. En (a,d) las esferas gris claras representan los vértices que satisfacen un conjunto de criterios menos selectivos.

2.1.2. Detección de pólipos basada en ajuste de superficies

En 2005, Kitasaka et al. [KIT05] proponen detectar pólipos calculando la curvatura en base al ajuste de superficies en lugar de hacer análisis de curvatura basado en la aproximación de la primera y segunda derivada, lo cual está muy influenciado por el ruido contenido en las imágenes de tomografía. El método propuesto consiste de varios pasos, primero la **extracción de las paredes del colon** usando *región creciente*¹³ sobre las imágenes de tomografía, seguido de una operación de dilatación y substrayéndole la cavidad del colon. Luego es necesario un **ajuste de superficie**. Para cada punto de la pared colónica (punto objetivo), se localizan puntos de intensidad mediante el uso de interpolación lineal en la imagen original, alrededor del área cuyo centro coincide con el punto objetivo. La ecuación cuadrática mostrada a continuación ha sido ajustada para calcular los puntos de intensidad:

$$f(x, y, z) = a_1x^2 + a_2y^2 + a_3z^2 + a_4xy + a_5yz + a_6zx + a_7x + a_8y + a_9z = 0$$

En consecuencia se obtienen nueve parámetros ($a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$). Los parámetros a_i se encuentran aplicando mínimos cuadrados. Cuando la forma de la región de estudio es muy irregular el error cuadrático tiende a ser grande, y se toma una vecindad más pequeña para que el área de estudio sea menor y más fácil de manejar. De esta forma el algoritmo puede adaptarse a distintos tamaños de pólipos.

Luego de esto, para el **cálculo de curvatura** por cada punto objetivo, se calculan la primera y segunda derivada de la superficie. Con estos valores se calcula el índice de forma y el valor de curvatura. Estas derivadas son utilizadas para determinar la forma de la pared del colon. Finalmente, los **candidatos a pólipos** se obtienen extrayendo aquellos puntos que muestran formas convexas en base al índice de forma y valor de curvatura.

Usando este método, Kitasaka et al. [KIT05] reportan menos falsos positivos que los obtenidos aplicando el método basado en curvaturas. La Figura 2.4 muestra una comparación entre resultados obtenidos con el método propuesto por Yoshida (detección de pólipos basada en curvatura) y el propuesto por Kitasaka.

¹³ **Región creciente** (región growing) es una técnica para extraer regiones de la imagen que están conectadas según cierto criterio predefinido. Este criterio puede estar basado en información de intensidades y/o bordes de la imagen.

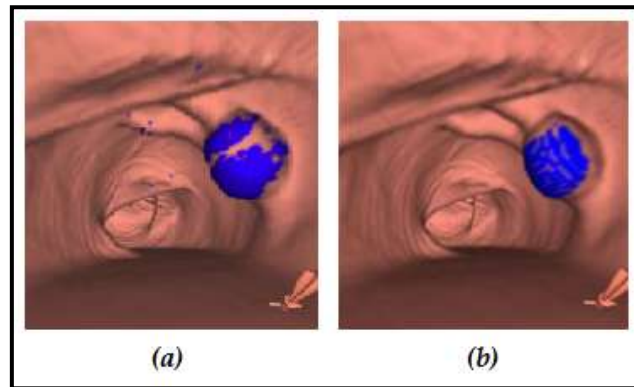


Figura 2.4: Ejemplos de pólipos marcados. (a) usando método propuesto por Yoshida. (b) usando el método propuesto por Kitasaka.

2.1.3. Detección de pólipos basada en líneas de curvatura

En el 2006 Zhao et al. [ZHA06] introdujeron una técnica para caracterizar y visualizar las líneas de curvatura en la superficie del colon con el fin de detectar pólipos. La idea básica consiste en mostrar la superficie del colon con iluminación y mejorarla desplegando las líneas de curvatura de estas superficies. Este método trabaja sobre la superficie del colon extraída con la técnica de *marching cubes*¹⁴, y con iso-superficies implícitas en el volumen extraídas directamente con Ray Casting.

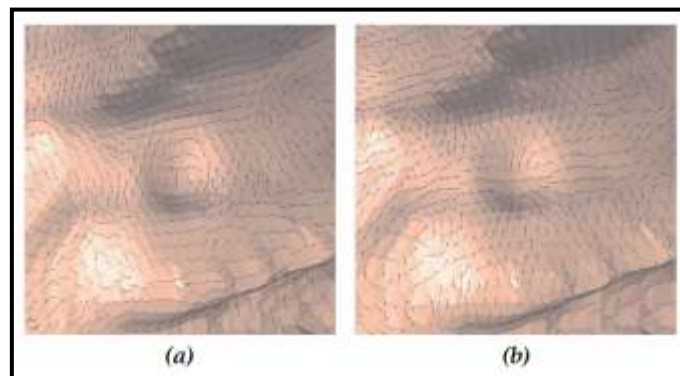


Figura 2.5: Ejemplo de líneas de curvatura resaltando un pólipo. (a) patrón circular en las direcciones de máxima curvatura. (b) patrón de enfoque en las direcciones de curvatura mínima.

¹⁴ **Marching cubes** se trata de un algoritmo para la poligonalización de cualquier tipo de isosuperficies que consiste en determinar el paralelepípedo que encierra un blob (**B**inary **L**arge **O**bjects, *objetos grandes binarios*) haciendo uso de la información de la posición y tamaño de las primitivas que lo componen y seguidamente se divide dicho paralelepípedo en cubos de tamaño constante.

La superficie de un pólipo consta de dos partes, la parte superior y la parte inferior (ver Figura 2.6). Llamamos *tope* a la parte superior de un pólipo y a la parte inferior la llamamos *cuello*. El tope del pólipo es el área más sobresaliente; es ideal representarlo como una superficie esférica o elipsoidal. El cuello de un pólipo es la zona de transición desde la raíz hasta el tope del pólipo. Es típicamente un anillo cerrado y una superficie anticlástica, es decir, todos los puntos en el cuello del pólipo son hiperbólicos, sin embargo, hay que tener en cuenta que la geometría de los pólipos no es siempre la misma ni presenta un patrón específico.

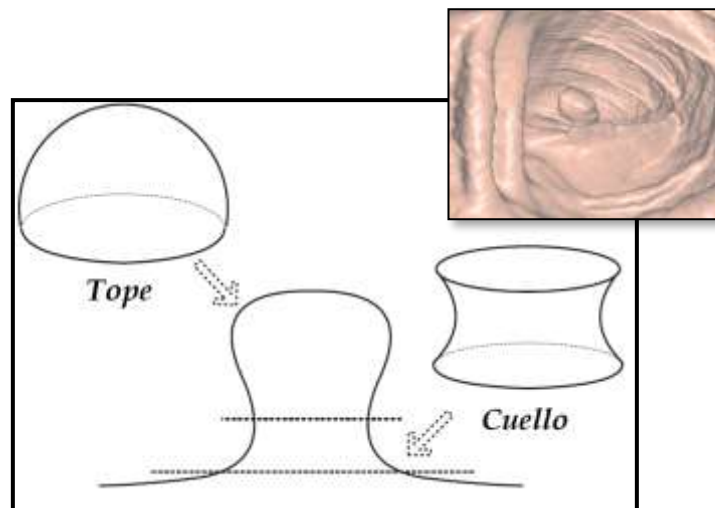


Figura 2.6: La superficie de un pólipo tiene dos partes tal como se ilustra en esta figura.

Existen también pólipos planos. Los pólipos planos son aquellos que tienen una altura inferior a la mitad del diámetro de la lesión. En general no poseen un cuello muy definido. (ver Figura 2.7)



Figura 2.7: Imagen de un pólipo plano.

Cuando un pólipo no puede ser fácilmente identificado sólo de acuerdo a su tope, se hace necesario el análisis del cuello. Cada pólipo tiene un cuello siempre que este sobresale de la pared del colon. Alrededor de una de estas zonas anticlásticas, las direcciones de las curvaturas principales presentan patrones particulares de los pólipos, es decir, patrones circulares o de enfoque.

El primer paso del método consiste en tener un estimado de la dirección de las curvaturas tanto principal como secundaria. Patrones específicos en líneas de corriente de curvatura puede ayudar a discriminar los pólipos del fondo. El patrón circular en direcciones de máxima curvatura alrededor el cuello indica la posibilidad de pólipos. Como el cuello de un pólipo es normalmente un área de anillo cerrado, se espera que líneas de corriente aproximadamente cerradas representen la parte estrecha del cuello del pólipo.

La forma de obtener este estimado varía según el tipo de despliegue de la imagen que se realice. Una forma es utilizando líneas de curvatura en superficies explícitas con mallado triangular. El primer paso para generar líneas de corriente de curvatura en mallas triangulares es la estimación de curvatura. Para esto, se utiliza el método NVV desarrollado por Page et al. [PAG02] el cual utiliza la vecindad geodésica, ya que esta garantiza un muestreo direccional balanceado alrededor de cada vértice del mallado, lo cual es muy útil para reducir los efectos de la irregularidad de malla y el ruido. El método NVV ofrece una estimación robusta para las direcciones principales de la curvatura. La corriente calculada en cada paso de integración es proyectada sobre el mallado.

Para distribuir las corrientes a lo largo del mallado de triángulos se utilizan semillas que se disponen a lo largo de toda la superficie. La distribución de estas semillas controlada por curvatura contribuye mucho a la caracterización del pólipo.

Otra forma es utilizando líneas de curvatura en Iso-superficies implícitas. Las superficies implícitas son superficies definidas a partir de una relación del tipo $f(x,y,z)=0$. Las tomografías computarizadas son representaciones en puntos discretos de diferentes partes del interior del cuerpo. Estos puntos representan la solución de una ecuación del tipo $f(x_i, y_i, z_i)=0$, aunque f es desconocida en este caso. Como veremos más adelante, se pueden hallar las derivadas direccionales

de f en cada punto de la Iso-superficie, aún desconociendo la formula de f mediante convolución¹⁵. En comparación con una red triangular explicita, una superficie implícita provee una visualización de la pared del colón más pareja y clara. Usualmente se visualiza usando técnicas de ray-casting de volumen. En la técnica propuesta se describe un nuevo enfoque para generar líneas de corriente de curvatura en iso-superficies implícitas.

En la estimación de curvatura en Iso-superficies utilizan el método propuesto por Van Vliet y Verbeek [VAN98] para estimar curvaturas principales en una iso-superficie de un volumen 3D. El algoritmo consta de tres pasos. En primer lugar, se calcula el vector gradiente \mathbf{g} y la matriz hessiana \mathbf{H} en un punto p de la iso-superficie. En un volumen 3D, estas derivadas se pueden aproximar realizando la convolución con la derivada de una función gaussiana con ancho específico del núcleo σ . En segundo lugar, a la matriz \mathbf{H} se le multiplica por una matriz de rotación para alinear un eje con la dirección del gradiente \mathbf{g} local, lo cual da origen a una matriz \mathbf{H}_r que tiene la siguiente forma:

$$\mathbf{H}_r = \begin{pmatrix} f_{gg} & \dots & \dots \\ \vdots & f_{uu} & f_{uv} \\ \vdots & f_{vu} & f_{vv} \end{pmatrix} = \begin{pmatrix} f_{gg} & \dots \\ \vdots & \mathbf{H}_t \end{pmatrix}$$

\mathbf{H}_t es la matriz Hessiana en el eje 2D local con centro en el punto p y paralelo a la superficie en dicho punto. Finalmente, se hace el análisis de autovalores y autovectores en \mathbf{H}_t . El valor local de las curvaturas principales en P puede ser calculado basándose en la magnitud del gradiente local y los autovalores λ_1 y λ_2 :

$$k_1 = \frac{-\lambda_1}{\|\mathbf{g}\|} \quad \mathbf{y} \quad k_2 = \frac{-\lambda_2}{\|\mathbf{g}\|}$$

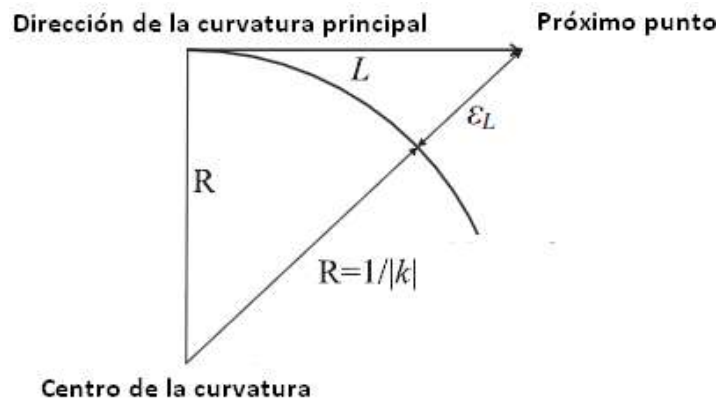
Los cuales deben ser transformados al sistema de coordenadas originales en 3D.

En cuanto al trazado de las corrientes de líneas de curvatura en Iso-superficies implícitas, se restringen los puntos de las corrientes de líneas a la iso-superficie, de manera implícita cuando estas se trazan a través del volumen 3D. La línea de corriente se traza mediante la integración gradual

¹⁵ Una **convolución** es una operación matemática que transforma dos funciones f y g en una tercera función que en cierto sentido representa la magnitud en la que se superponen f y una versión trasladada e invertida de g y se denota $f * g$.

desde el punto de semilla hacia adelante y hacia atrás. Para cada paso de integración, se calculan primero las direcciones principales de curvatura en el punto actual de la línea de corriente de la iso-superficie usando el método propuesto por Van Vliet y Verbeek. Se utiliza interpolación lineal para estimar el gradiente, el Hessiano y los valores de los datos en una posición arbitraria en el volumen 3D. Luego la corriente es propagada un paso siguiendo la dirección de la curvatura principal a una longitud L controlada por la curvatura local utilizando la siguiente fórmula:

$$L = \sqrt{\varepsilon_L \left(\frac{2}{|k|} + \varepsilon_L \right)}$$



donde ε_L es definida por el usuario para indicar cuán lejos puede alejarse el punto de la iso-superficie. Luego la corriente es proyectada en la iso-superficie y se repite iterativamente este proceso hasta que la corriente se encuentre muy cerca de otra corriente o cuando la corriente haya recorrido una distancia determinada.

Para la siembra de semillas controlada por la curvatura, la idea básica es utilizar el valor de la curvatura mínima para determinar el espacio entre corrientes de máxima curvatura y vice-versa. El primer paso es iniciar una corriente desde un punto determinado de la superficie. En cada paso del trazado, se colocan un par de semillas, una a cada lado, en dirección perpendicular a la corriente y a la distancia controlada por la curvatura local dada por la siguiente fórmula:

$$d_s = \sqrt{\varepsilon_1 \left(\frac{2}{|k|} + \varepsilon_1 \right)}$$

Donde ε_1 es definida por el usuario (similar a ε_L). Posteriormente estas semillas se colocan en una cola de semillas. Iterativamente se van seleccionando semillas y se van empezando nuevas corrientes.

Cada vez que se inicia una nueva corriente, se eliminan las semillas inválidas (las que están muy cerca de otra corriente). Este proceso se detiene cuando ya no quedan semillas válidas.

Luego de que todas las corrientes se han trazado, se desea seleccionar las corrientes que se encuentran alrededor del cuello de los pólipos preseleccionados. Para esto se seleccionan las corrientes con una mayor cantidad de puntos hiperbólicos en el área del pólipo (es decir, donde las curvaturas principales tienen signos opuestos). Posteriormente, se quiere encontrar corrientes (casi) cerradas. Para lograr esto, se utiliza una adaptación de los métodos desarrollados por Portela [POR97] y Sadarjoen et al. [SAD99] para hallar corrientes con patrones de remolino.

La principal virtud de esta técnica es permitir mejorar las imágenes de la colonoscopia virtual agregando las líneas de curvatura que permiten al radiólogo caracterizar fácilmente pólipos. Sin embargo, este método aun no ha sido adaptado para permitir la detección automática de los mismos.

2.1.4. Detección de pólipos basada en difusión de calor

En 2007 Konukoglu et al. [KON07] observaron que si la cavidad del colon se calienta a un nivel constante, el proceso de difusión de calor generaría singularidades (pozos) en los patrones de difusión del calor, cerca de los centros de estructuras protuberantes. La difusión del calor es más rápida en regiones en donde las superficies son cóncavas que en donde son convexas.

Konukoglu et al. [KON07] proponen el campo de difusión de calor (HDF - *Heat Diffusion Fields*), para detectar y caracterizar una estructura sobresaliente como una superficie en 3D. El proceso de difusión tiene un efecto de realce ya que el HDF utiliza la estructura de colisión del calor propagado que lleva a la información estructural de toda la superficie de los pólipos; en otras palabras, el HDF hace un resumen de la superficie del colon en un barrido local. El método se basa en la detección y caracterización de los hundimientos en conjunto con la distribución local de los vóxeles en torno a él. El único parámetro del método es el coeficiente constante de difusión.

Inicialmente el colon es segmentado para obtener el interior del mismo al cual se le asigna una temperatura alta constante. A todos los tejidos no identificados como pertenecientes al interior

del colon se les asigna una temperatura baja constante. A continuación en la Figura 2.8 se simula el proceso de enfriamiento de la cavidad del colon.

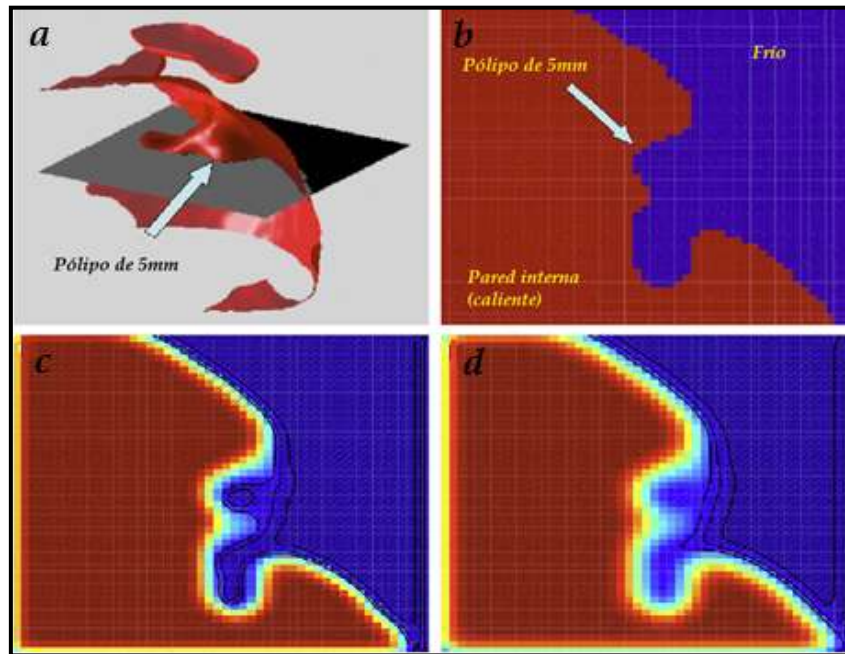


Figura 2.8: Proceso de difusión de calor, (a) Renderizado de superficie de la pared del colon, (b) el campo de la temperatura inicial, (c, d) los campos de temperatura a medida que avanza el proceso de difusión, el lado rojo es la cavidad del colon (caliente) y el lado azul es la pared del colon (frío).

Un pólipo perfectamente esférico daría lugar a una singular esfera simétrica en el centro del pólipo. Esto no siempre ocurre debido a las variantes formas de los pólipos del colon. Para determinar que formas podrían representar pólipos Konukoglu et al. [KON07] utilizan el parámetro *FA Fractional Anisotropy*. *FA* es igual a cero ($FA=0$) en caso de una perfecta simetría esférica, por lo tanto mientras más bajo sea el valor de *FA*, mayor es la probabilidad de que dicha estructura represente un pólipo. Sin embargo, a pesar de que el parámetro *FA* es especialmente útil en la discriminación entre pliegues y pólipos con cuello de botella, por sí sólo no es suficiente para eliminar todos los falsos pólipos. Es necesario realizar un análisis local de la pared del colon en torno a los aciertos y calcular la triangulación uniforme (TU) de los parámetros. La idea detrás de TU es realizar una triangulación de Delaunay¹⁶ de una superficie esférica alrededor del pólipo y verificar la similitud entre las áreas de

¹⁶ Una **triangulación de Delaunay** es una red de triángulos que cumple la condición de Delaunay. Esta condición dice que la circunferencia circunscrita de cada triángulo de la red no debe contener ningún vértice de otro triángulo.

todos los triángulos. TU se usa junto con la FA para la identificación ya que ninguno de ellos por sí sólo garantiza la detección de un pólipo verdadero.

El método de HDF proporciona inmunidad contra las irregularidades de la superficie del colon producto por ejemplo de ruido en la segmentación. Los resultados sugieren además una mejora respecto al método de Yoshida [YOS01], pero no se pudo hacer una comparación directa ya que trabajaron con datos distintos. A pesar de esto, su principal desventaja es el alto costo computacional requerido. Sin embargo, Konukoglu et al. [KON07] sugieren que esto podría mejorarse cambiando un poco la implementación. Por otro lado sólo los pólipos de tamaño igual o mayor a 8 mm se consideraron, es decir, todos los resultados asociados con pólipos pequeños fueron excluidos.

2.1.5. Detección de pólipos basada en línea de agua

En general, los pólipos tienen formas distintas, son bulbosos y con formaciones elípticas que sobresalen en las paredes del colon. La forma de un pólipo es casi convexa (a pesar de las irregularidades de menor importancia). A partir de esto se diseñó un método para determinar los grupos de vértices que mostraron la forma característica de un pólipo “casi convexo” llamado el método del plano de agua. Este método fue planteado por Bitter et al. [BIT05] en el año 2005.

El método se basa en simular que la pared del colon se llena de agua a partir de un vértice cualquiera, desde el exterior del colon hacia el interior, hasta detectar el momento a partir del cual si se agrega más agua esta se derrama. Los vértices pueden tener dos estados: los vértices por debajo de la línea de agua se consideran húmedos y los que están por encima se consideran secos (ver Figura 2.9).

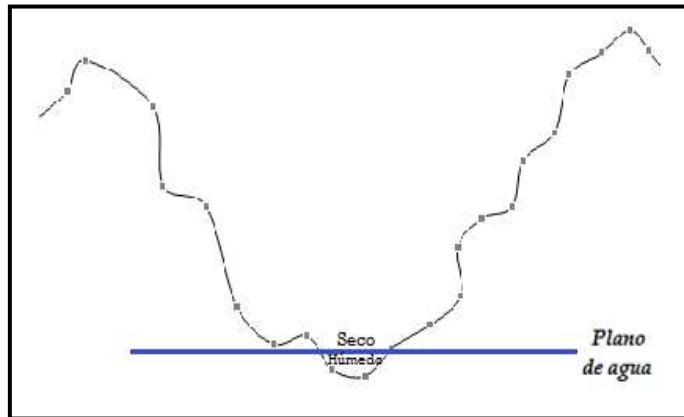


Figura 2.9: Ilustración de los vértices mojados y secos.

En la Figura 2.8 el método comienza en un vértice cualquiera y utiliza la información espacial en los vértices vecinos para generar un plano de agua.

El plano de agua hará que el vértice de partida sea “húmedo” pero deja sus vértices vecinos “secos”. El objetivo del algoritmo es encontrar el plano de agua óptimo tal que el número máximo de vértices este del lado “húmedo” del plano de agua, definiendo al grupo de vértices que identificarán un candidato a pólipo. El método seguirá colocando vértices “húmedos” uno por uno y luego a sus vecinos y así sucesivamente hasta que el plano de agua no pueda moverse más sin causar fugas de agua. Este método pondrá a prueba a todos los vértices para ver si el plano de agua puede ser creado y movido confiablemente definiendo el grupo de vértices “húmedos” que son los principales candidatos a pólipos. Todos los grupos candidatos serán coloreados. El primer plano de agua se genera de manera tal que sea perpendicular a la media normal de todos los triángulos adyacentes al vértice de partida, o vértice semilla. Todos los vértices vecinos se insertan en el grupo límite (de vértices que se encuentran justo por encima del nivel de agua actual). Una vez que el agua inicial se genera en el plano, este puede ser movido por cuatro modos diferentes. Estos modos son: (1) Incremento de la altitud, (2) reorientar el plano con un punto de anclaje, (3) reorientar el plano con dos puntos de anclaje, y (4) ningún movimiento más.

El movimiento es discreto, moviendo el plano lo suficiente como para causar sólo un límite único para el conjunto de vértice a mojarse.

1) *Incremento de la altitud*: El plano de agua incrementa solo su altitud sin cambiar la dirección del plano de agua.

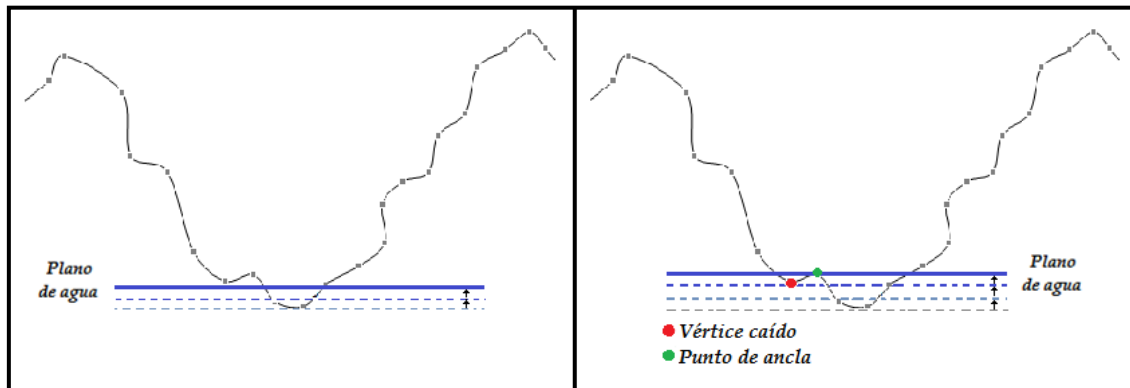


Figura 2.10: En la imagen izquierda se incrementa solo la altitud, en la derecha se cambian las fronteras del plano de agua.

En la Figura 2.10 podemos ver que el algoritmo eleva el plano de agua hasta el vértice más cercano al agua en el grupo límite y añade a sus vecinos secos a dicho grupo. El algoritmo se queda en este modo siempre que cada vértice seleccionado mantenga la propiedad de que todos sus vértices vecinos estén en el lado seco del plano de agua. El plano continuará incrementando su altitud hasta encontrar un vértice que tenga un vecino que no esté del lado seco del plano de agua (y que no se encontrara en el lado mojado en el paso anterior). Elevar el plano de agua más arriba permitirá que el agua se filtre en ese vecino que todavía no estaba incluido en el grupo. Por lo tanto tendríamos que cambiar del modo 1 al 2.

2) *Reorientar el plano con un punto de anclaje*: En el modo 2, elevar el plano de agua hará que la dirección del plano cambie. Esto ocurre cuando un cambio anterior del plano de agua provoca que el vecino de un vértice seleccionado se convierta en un vértice vecino “caído”, o un vértice vecino que no está del lado seco del plano de agua. Un vértice es llamado punto de “ancla” si tiene un vértice vecino caído, tal como vemos en la imagen derecha de la Figura 2.10. El plano de agua debe ser levantado por encima de un punto de anclaje para evitar fugas. El plano de agua se ancla a este punto, inclinándolo para recuperar el vértice vecino caído.

La inclinación es una rotación alrededor de un eje que se encuentra en el plano de agua, el cual encaja en el punto de anclaje y es perpendicular al segmento de línea del punto de anclaje del

vecino caído, así como se muestra en la Figura 2.11. Esta dirección de inclinación minimiza el ángulo que el plano de agua tendrá que inclinar cuando el vértice vecino sea considerado seco de nuevo. A continuación, el plano se inclina lo suficiente para mojar el ángulo límite más cercano al conjunto de vértices. La inclinación se repite hasta que el vértice caído este en la región seca o hasta que el segundo vértice vecino caído es encontrado. En el primer caso el algoritmo va a volver al modo 1, incrementando la altitud solamente. En este último caso, el algoritmo va a caer en el modo 3, incrementando el plano de agua con respecto a dos puntos de anclaje.

3) *Reorientar el plano con dos puntos de anclaje*: Si el segundo vértice vecino caído ocurre, el plano de agua debe reorientarse con respecto a dos puntos de anclaje como vemos en la Figura 2.11. Por lo tanto, la línea de inclinación es el vector diferencia entre los dos puntos de anclaje, que ahora definen el eje de rotación del plano de agua y que serán usados para reorientarlo. Cuando está en este modo, en función de la recuperación de los vértices vecinos caídos, el algoritmo puede entrar en el modo 1, 2, o permanecer en el modo 3. Si otro vértice experimenta una caída de un vértice vecino además de los dos puntos de anclaje que todavía están tratando de recuperar sus vértices vecinos caídos, entonces, la máxima reorientación del plano debe ser alcanzada y el algoritmo está ahora en el modo 4, en el cual ningún movimiento mas del plano de agua es posible sin causar fugas.

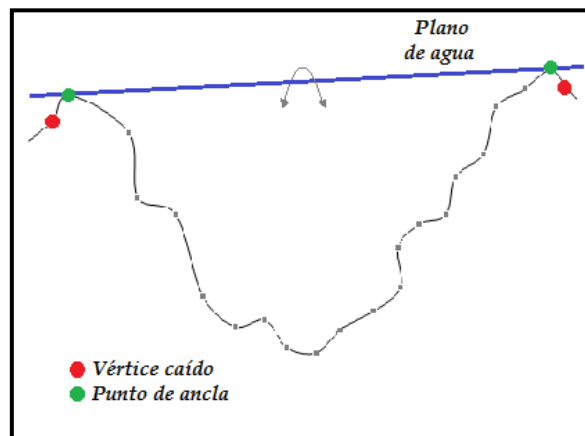


Figura 2.11. Reorientación del plano.

4) *Ningún movimiento más*: En el modo 4, la máxima reorientación del plano de agua esta alcanzada y todos los vértices por debajo del plano de agua son considerados el “grupo húmedo”, tal como se observa en la Figura 2.12. Este grupo es evaluado para ver si es un buen candidato a pólipo. Se considera bueno cuando este genera un plano de agua que consta de más de un número mínimo

de vértices. En este algoritmo, han definido el umbral como un conjunto de 20 o más vértices. Estos grupos son buenos candidatos a pólipos y se seguirán procesando, y posiblemente serán rechazados en otros análisis.

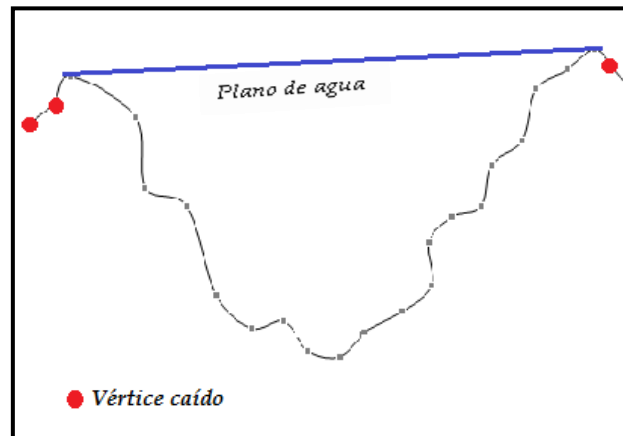


Figura 2.12. Ningún movimiento más.

Bitter et al. [BIT05] reportan mejoras con respecto a métodos de análisis de curvatura que ellos mismos han implementado para una aplicación de colonoscopia virtual, aunque no especifican el algoritmo particular utilizado. Entre otros, permiten detectar protuberancias que no eran detectadas por el método anterior y permiten detectar regiones continuas para candidatos a pólipos, cuando los métodos basados en curvatura solo detectaban algunos vóxeles del pólipo.

2.1.6. Reducción de falsos positivos

Después de haber detectado los candidatos a pólipos se hace necesario clasificarlos para determinar cuáles son los falsos positivos. En el 2001 Gokturk y Tomasi [GOK01] usan una Máquina de Vectores de Soporte (SVM - *Support Vector Machine*) para eliminar estos falsos positivos. Las máquinas de soporte vectorial o máquinas de vectores de soporte son un conjunto de algoritmos de aprendizaje supervisado desarrollados por Vladimir Vapnik y su equipo en los laboratorios AT&T¹⁷.

En principio utilizan vistas perpendiculares del volumen en vez de reconstrucción 3D. Esto lo hacen para imitar la forma en la que trabajan muchos radiólogos. La idea consiste en tomar una

¹⁷ La Corporación AT&T (siglas de su antiguo nombre, **American Telephone and Telegraph**; NYSE: AT&T) es una compañía estadounidense de telecomunicaciones. Provee servicios de voz, video, datos, e internet a negocios, clientes y agencias del gobierno.

cantidad considerable de muestras aleatorias de planos mutuamente ortogonales, es decir que cada volumen de candidato a pólipo se corta con tres planos perpendiculares que formaran tripletas. Seguidamente se calculan ciertos atributos geométricos sobre estas tripletas. Para calcular los atributos a usar como rasgos para la SVM se ajustan varias formas geométricas a cada imagen de la tripleta, entre otras figuras se ajustan circunferencias, curvas cuadráticas y líneas rectas al borde de la imagen previamente binarizada¹⁸ según un cierto umbral para detectar el borde (ver Figura 2.13).

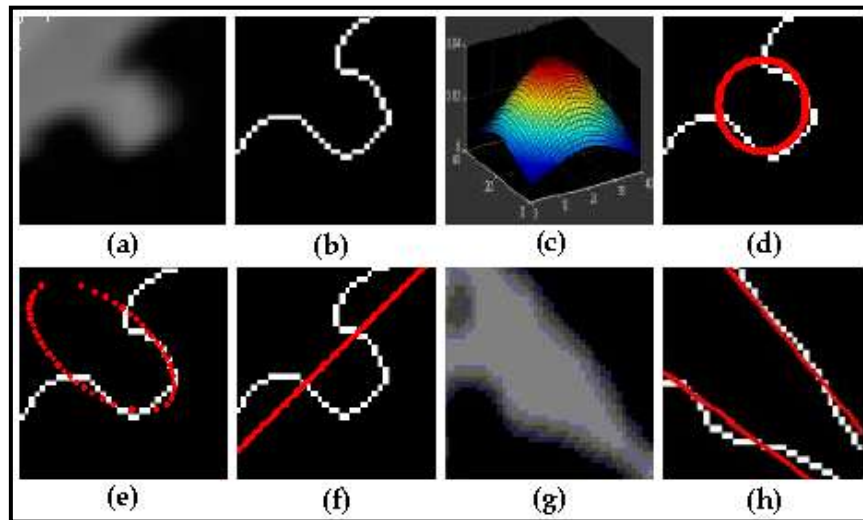


Figura 2.13: Ilustración de las formas primitivas: (d) Círculo, (E) cuádrlica (f) de línea (h) líneas paralelas. Cada forma se pondera por la gaussiana dada en (c).

Se hace también un análisis de líneas paralelas a fin de determinar si existen pliegues ya que la proyección de un pliegue sobre un plano contendrá probablemente líneas casi paralelas. Además de los atributos basados en forma, se extraen rasgos de intensidad de la parte de la imagen correspondiente a tejidos (media y desviación estándar de la intensidad del tejido). Todos estos atributos son calculados para cada una de las imágenes de la tripleta, y luego se calcula su histograma. Seguidamente a partir de este histograma se seleccionan para cada una de las tripletas muestras que pertenezcan a tejido sano y otras que pertenezcan a pólipos y se usan para entrenar una SVM para así clasificar pólipos y tejidos que no son pólipos.

¹⁸ La **binarización** de una imagen digital consiste en convertir la imagen digital en una imagen en blanco y negro, de tal manera que se preserven las propiedades esenciales de la imagen.

Dado un conjunto representativo de vectores de pólipos y de tejido sano, un clasificador de algoritmo de aprendizaje lleva este conjunto de entrenamiento como entrada y produce una salida. El objetivo es encontrar una función de separación entre los pólipos y el tejido sano. El clasificador SVM tiene entonces el objetivo de buscar la hipersuperficie óptima diferencial entre ambas clases. Esta hipersuperficie óptima no sólo clasifica correctamente los datos sino que también maximiza el margen de los puntos más cercanos al hiperplano. Matemáticamente, consideremos el problema de separar el conjunto de entrenamiento S de puntos $x_i \in \mathbb{R}^2$ con $i = 1, 2, \dots, N$. Cada punto x_i pertenece a cualquiera de las clases y en consecuencia se le asigna una etiqueta $y_i \in \{-1, 1\}$. La SVM implícitamente transforma la función dada de vectores x en nuevos vectores $\phi(x)$ en un espacio con más dimensiones, de manera que la hipersuperficie que separa la x se convierte en un hiperplano en el espacio de $\phi(x)$. Encontrar el hiperplano óptimo es un problema de optimización en el que la distancia de los puntos de margen al plano es maximizada. En el problema de optimización, sólo los productos internos de la forma $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$, llamados núcleos, necesitan ser recalculados. En el proceso de clasificación, sólo los vectores que están muy cerca de la hipersuperficie de separación deben ser considerados. Estos vectores son llamados vectores de soporte y básicamente son los puntos más cercanos al hiperplano óptimo de separación. Una descripción explicativa de los vectores de soporte sería entonces, "los puntos de datos que llevan a la diferenciación característica entre las dos clases". Una vez obtenidos estos vectores, la distancia de un vector x desde el clasificador óptico determina la clasificación de la SVM.

A diferencia de otros métodos de clasificación, SVM minimiza el riesgo estructural, dado que es baja la probabilidad de clasificar mal los datos. Además de que la SVM comprime toda la información relevante en un conjunto de entrenamiento en un pequeño número de vectores de soporte y utiliza sólo estos vectores para clasificar nuevos datos. Así, las características distintivas de ambas clases están implícitamente identificadas. Usar SVM en lugar de hacer la clasificación heurística a mano, explota toda la información en el entrenamiento haciéndola óptima y elimina las conjeturas de la tarea de definir los criterios adecuados de discriminación.

2.1.7. Combinación de malla, volumen y representación de líneas de curvatura para la detección de pólipos en colonografía por TC.

Se han presentado muchos enfoques diferentes para la detección de pólipos a través de los últimos años [RAV09]. También se ha demostrado que los pólipos se pueden detectar igualmente bien como protuberancias en una representación explícita (superficie de malla) o como en una representación implícita (nivel gris) de la pared del colon. En el 2009 V.F. van Ravesteijn et al. [RAV09] presentan un patrón para el reconocimiento de pólipos, que combina los dos enfoques y revela que en cierta manera son complementarios. Por otra parte, el análisis de una línea de corriente, utilizada originalmente en el campo de la visualización de datos, es incorporado en su marco.

Los pólipos pueden caracterizarse por la condición de que la curvatura principal más pequeña es mayor que cero. En otras palabras son como topes, mientras que los pliegues del colon son típicamente alargados con una curvatura positiva y la otra cerca de cero o (ligeramente) negativa. Debido a la naturaleza global de la forma cilíndrica de colon, las regiones con dos curvaturas positivas son relativamente escasas. Las curvaturas principales de la pared del colon o bien se pueden calcular mediante la diferenciación de una representación de malla o directamente de los datos de la imagen subyacente.

Van Ravesteijn et al. [RAV09] concluyen que los pólipos son salientes, tanto en la representación explícita como en la implícita. La detección de falsos pólipos es un problema para ambas representaciones. El enfoque explícito actúa directamente en una representación de la pared del colon, mientras que el método implícito interactúa con los datos subyacentes. Van Ravesteijn et al. [RAV09] conciben un enfoque combinado en la medida en que la protuberancia es mejor representada en el método explícito, mientras que las propiedades de intensidad y volumen lo son por el método implícito.

A fin de captar la información esencial de la forma de la superficie, líneas de corriente se esparcen en toda la superficie dependiendo de las magnitudes principales de la curvatura local. En las superficies menos curvas, se generaron menos líneas de corriente que en las superficies altamente curvadas. Las líneas de corriente que forman parte de la pared del colon tienen la característica útil

de que tienden a rodear el cuello de los pólipos. El “ángulo de liquidación” se deriva de esta característica. Se define como la variación acumulada de la dirección a lo largo de la línea de corriente. En cada punto de muestreo, el cambio diferencial de la dirección se determina sobre la base de la superficie normal en ese punto. Líneas de curvatura cerradas, como los cuellos de pólipos, tienen un ángulo de liquidación de al menos 2π . Los primeros experimentos mostraron los valores máximos del ángulo de liquidación están estrechamente relacionados con detecciones de pólipos verdaderos y por lo tanto podría ser útil para reducir el número de falsos positivos (PM) que han encontrado sistemas CAD.

El método explícito implica dos características. En primer lugar, utiliza una función que se deriva del campo de desplazamiento de la malla. Esta función mide el porcentaje de los candidatos con un desplazamiento mayor que un cierto umbral T , además se denota como ϕT . Van Ravesteijn et al. [RAV09] tomaron un umbral de 0,6mm. Este diseño favorece a los candidatos con bordes empinados y formas compactas. La segunda característica que utilizaron fue la intensidad media de los candidatos, calculado como la suma ponderada de todos los tonos vóxeles incluidos en una máscara de segmentación. Este último consiste en la zona comprendida entre el original y la malla desplazada. Para analizar el rendimiento de un sistema combinado, las correspondencias entre los candidatos determinaron que ambos métodos debían establecerse. El método implícito actúa sólo en las regiones que fueron elegidas como candidatos por el método explícito. Estas regiones se obtienen dilatando la máscara de segmentación 10 veces. Un área de segmentación correspondiente para el método implícito se deriva de la imagen deformada por el umbral de la diferencia de intensidad a un valor de 100 unidades Hounsfield (HU). Así, cada candidato a partir del método implícito es inherentemente vinculado a la detección correspondiente de la malla.

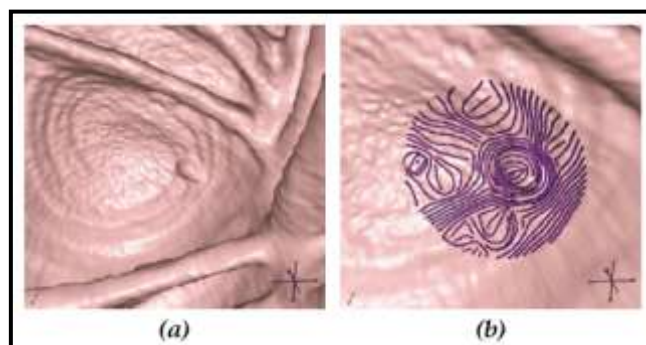


Figura 2.14: (a) Un pólipo con su entorno. (b) Ejemplo de líneas de curvatura generadas en las cercanías del pólipo.

En la Figura 2.14 se muestran las líneas de curvaturas generadas por este método para identificar los pólipos. En resumen, V.F. van Ravesteijn et al. [RAV09] presentan un sistema que detecta pólipos basado en cuatro funciones intuitivas. Inicialmente, el sistema se basaba en dos características: ΦT (porcentaje de los candidatos con un desplazamiento mayor que un cierto umbral T) y la intensidad media. Luego, se agregó el método implícito por medio de la función de diferencia de intensidad máxima. Por último, el ángulo de liquidación se incluyó para ambas configuraciones anteriores.

La detección se realiza por medio de dos enfoques diferentes. Cada enfoque caracteriza diferentes aspectos de los candidatos. En efecto, los objetos que sobresalen son detectados por medio de deformar una representación explícita de la superficie del colon o mediante la modificación de los datos de intensidad que contiene una representación implícita. También han añadido una forma-descriptora derivada del análisis de las líneas de curvatura. Se demostró que la característica basada en el campo de desplazamiento de malla, la intensidad media, la diferencia de intensidad máxima y el ángulo de liquidación de la línea de corriente son suficientes para un buen desempeño. Se analizaron 56 imágenes de 28 pacientes y se encontraron más del 92% de los pólipos con menos de 1,6 falsos positivos por análisis.

2.2. Aproximación de la derivada

Existen varios filtros clásicos que permiten aproximar la primera derivada de una imagen. Por ejemplo, operador de Sobel, Prewitt y Cross. Todos están basados en un kernel que se aplica a cada píxel de la imagen. El problema de estos operadores es que generalmente no son lo suficientemente aproximados, y si se utilizan para calcular la segunda derivada el error es aún mayor.

No es suficiente pedir que la función reconstruida sea precisa, también es necesario que sea continua ya que el ojo humano sea capaz de detectar fácilmente estas discontinuidades. También es importante la eficiencia del proceso de reconstrucción, sobre todo cuando hablamos de datos volumétricos muy pesados. Es primordial entonces una función que utilice tan pocas muestras como sea posible.

En 1998, Torsten Möller et al [MOL98], por primera vez, introducen un criterio de diseño para filtros de interpolación y de derivadas que arrojan un mínimo error numérico. Para lograr esto, utilizan expansión de Taylor de la suma de convoluciones.

Para reconstruir una función continua $f(t)$ o su derivada $f'(t)$ a partir de puntos de muestreo $f[k]$, convolucionamos con un filtro continuo w . El resultado de esta operación lo llamaremos $f_r^w(t)$:

$$f_r^w(t) = \sum_{k=-\infty}^{\infty} f[k] \cdot w\left(\frac{t}{T} - k\right)$$

en donde T es la distancia de muestreo.

Ahora, $f[k] = f(kT)$ puede expandirse con series de Taylor de $N + 1$ términos alrededor de t :

$$f[k] = \sum_{n=0}^N \frac{f^{(n)}(t)}{n!} (kT - t)^n + \frac{f^{(N+1)}(\xi_k)}{(N+1)!} (kT - t)^{(N+1)}$$

donde $f^{(n)}(t)$ es la derivada n -ésima de f y $\xi_k \in [t, kT]$.

Ahora, podemos substituir $f[k]$ en $f_r^w(t)$, para obtener una representación alternativa para el valor reconstruido en el punto t :

$$f_r^w(t) = \sum_{n=0}^N a_n^w(\tau) f^{(n)}(t) + r_N^w, i(\tau)$$

con:

$$a_n^w(\tau) = \frac{T^n}{n!} \sum_{k=-\infty}^{\infty} (k - \tau)^n w(\tau - k)$$

La única hipótesis requerida de la función original es que debe ser suave y debe tener una continuidad C^n .

En general, concluyen que para el diseño de un filtro es importante responder a tres preguntas:

- ✓ ¿Qué derivada de la función original necesitamos reconstruir?
- ✓ ¿Qué precisión necesitamos en el proceso de reconstrucción?
- ✓ ¿A cuál espacio C^n debe pertenecer la función reconstruida?

Las dos primeras preguntas pueden controlarse haciendo que a_n^w sea cero para $n < k$ (si queremos reconstruir la derivada k -ésima), igualmente requerimos que el coeficiente para la k -ésima derivada sea uno.

Condición 1: $a_n^w = 0 \forall n < k$ y $a_k^w = 1$

El objetivo principal del diseño del filtro es su precisión. Ganamos precisión numérica al requerir que los coeficientes de error a_n^w mayores a k sean cero. Esto lleva a filtros $N - EF$ (función de error de orden N).

Condición 2: $a_n^w = 0 \forall k < n < N + k - 1$

Podemos entonces considerar que el filtro w está compuesto por elementos w_k definidos por:

$$w_k = w_k(t) = w(\tau + k)$$

Un ejemplo de un filtro definido por piezas es el siguiente:

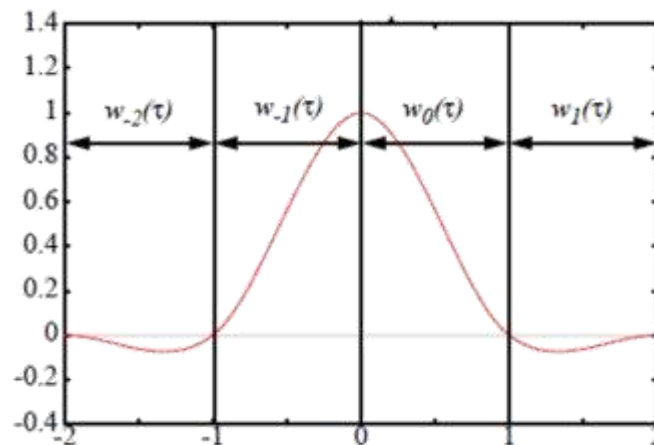


Figura 2.15: En esencia, el filtro tiene p trozos. Para los filtros simétricos y anti-simétrico p es par.

Las dos primeras condiciones van entonces a originar un sistema de ecuaciones lineal con incógnitas w_k . La solución a ese sistema de ecuaciones define el filtro w . Sin embargo, el filtro todavía no cumple la tercera condición, debe pertenecer a un espacio de funciones continuas C^M , esto es, la función w y sus M derivadas deben ser continuas.

Condición 3:

Como las condiciones 1 y 2 originan un filtro por piezas, la tercera condición puede expresarse como:

$$w_k(\tau) \in C^M \text{ y } w_k^{(m)}(1) = w_{k+1}^{(m)}(0) \forall k, m < M$$

Aunque no es necesario restringir el filtro a filtros polinomiales por piezas, los polinomios son más fáciles de utilizar e implementar. Así, las condiciones 1, 2 y 3 se traducen en un sistema de ecuaciones lineal para encontrar los coeficientes de estos polinomios.

Resumiendo, el proceso completo para diseñar un filtro incluye los pasos siguientes:

Paso 1: Resolver un sistema de ecuaciones lineales creado por las condiciones 1 y 2, sobre las piezas w_k del filtro w .

Paso 2: Escoger un conjunto de funciones base para la representación de w_k .

Paso 3: Resolver los coeficientes de las funciones base, considerando la condición 3, y la solución del Paso 1.

2.3. Programación en GPU

GPU computing, o GPGPU, es el uso de la GPU¹⁹ (unidad de procesamiento gráfico) para realizar operaciones de cálculo científico o técnico de propósito general [URL08].

El modelo empleado para esta tecnología se basa en el uso combinado de una CPU y una GPU en un sistema de coprocesamiento heterogéneo. La parte secuencial de la aplicación se ejecuta en la CPU y las partes de mayor carga computacional se aceleran en la GPU. Para el usuario, la aplicación

¹⁹ La unidad de procesamiento gráfico o **GPU** (acrónimo del inglés *graphics processing unit*) es un procesador dedicado al procesamiento de gráficos u operaciones de coma flotante, para aligerar la carga de trabajo del procesador central en aplicaciones como los videojuegos y o aplicaciones 3D interactivas.

simplemente se ejecuta más rápido porque utiliza la gran capacidad de la GPU para multiplicar el rendimiento.

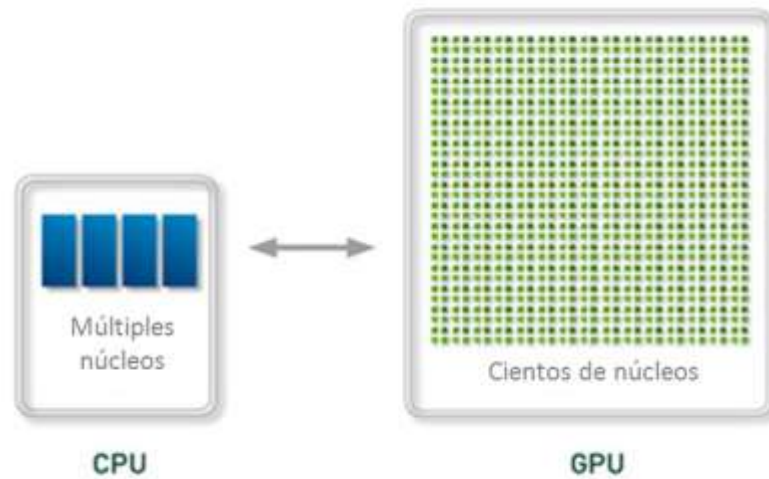


Figura 2.16. CPU vs GPU [URL08]

A lo largo de los años, la GPU ha evolucionado hasta alcanzar teraflops de rendimiento en las operaciones de cálculo en punto flotante. NVIDIA revolucionó la GPGPU y aceleró el mundo de la informática durante los años 2006-2007 con la introducción de su arquitectura de cálculo paralelo masivo “CUDA” siglas de *Compute Unified Device Architecture*. Esta arquitectura consta de cientos de núcleos de procesamiento que operan de forma conjunta para manejar los datos de la aplicación a mayor velocidad.

El éxito de las GPU en los últimos años reside en la facilidad del modelo de programación paralela CUDA asociado. En este modelo, el desarrollador de la aplicación modifica ésta para asignar a la GPU los kernels más complejos desde el punto de vista computacional. El resto de la aplicación permanece en la CPU. Asignar una función a la GPU implica reescribir ésta para aprovechar el paralelismo del procesador gráfico y agregar palabras clave de “C” para transferir los datos hacia y desde la GPU. El programador se encarga de lanzar decenas de miles de procesos (threads) de forma simultánea. El hardware de la GPU maneja estos procesos y programa su ejecución.

Debido a las grandes diferencias entre las arquitecturas de la GPU y la CPU, todos los problemas no pueden beneficiarse de programación paralela en la GPU. El acceso a memoria plantea

las mayores dificultades. Las CPU están diseñadas para el acceso aleatorio a memoria, en cambio, en una GPU el acceso a memoria está mucho más restringido.

El reto del programador consiste entonces en adaptar los accesos a memoria y las estructuras de datos a las características de la GPU. De esta forma, aquellos algoritmos con un alto grado de paralelismo obtendrán mayores beneficios de su implementación en la GPU.

2.3.1. Arquitectura CUDA

CUDA intenta aprovechar el gran paralelismo, y el alto ancho de banda de la memoria en las GPUs en aplicaciones con un gran coste aritmético frente a realizar numerosos accesos a memoria principal, lo que podría actuar de cuello de botella. Viene incorporada en todas las GPU NVIDIA desde la serie G8X. Su característica fundamental es que puede ser programada directamente en C más algunas extensiones.

CUDA se basa en un arreglo de multiprocesadores. Cada multiprocesador posee funciones especiales, instrucciones y memoria compartida. Para manejar cientos de hilos, utiliza una estructura llamada SIMT (Single Instruction Multiple Thread). Dicha estructura crea, maneja, organiza y ejecuta los grupos de hilos en grupos de 32 hilos paralelos llamados *warps*.

2.3.2. Organización de hilos (threads)

Los hilos se organizan en bloques, que se distribuyen sobre un grid con un máximo 512 hilos distintos. Cada hilo está identificado con un identificador único, que se accede con la variable `threadIdx`. Esta variable es muy útil para repartir el trabajo entre distintos hilos. `threadIdx` tiene 3 componentes (x, y, z), coincidiendo con las dimensiones de bloques de hilos. Así, cada elemento de una matriz, por ejemplo, lo podría tratar su homólogo en un bloque de hilos de dos dimensiones. Los hilos de un bloque se ejecutan concurrentemente. La ejecución de dichos bloques se puede alterar dependiendo del uso de la memoria compartida.

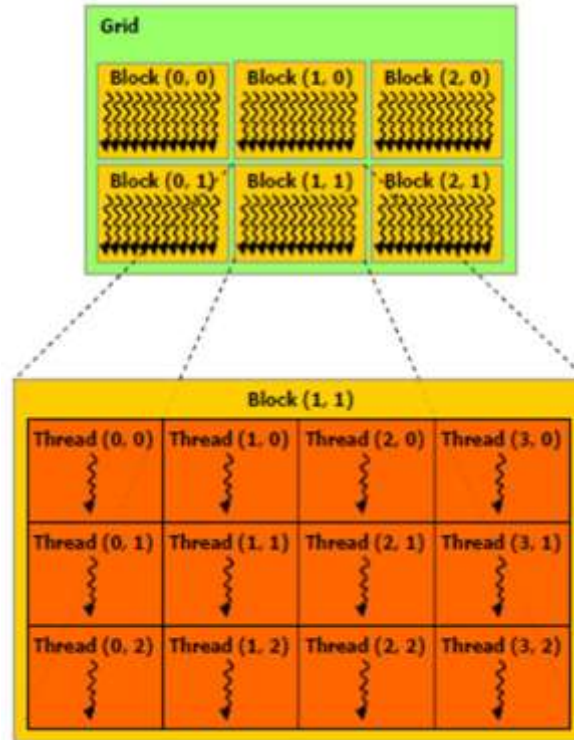


Figura 2.17. Red de bloques de hilos de dos dimensiones [URL09].

2.3.3. Jerarquía de memoria

CUDA dispone principalmente de tres memorias (ver figura 2.18):

- Memoria local: A la cual puede acceder cada hilo (thread).
- Memoria compartida: La cual es común a todos los hilos de un bloque. Es la más explotada debido a que es la que tiene mayor velocidad.
- Memoria global: Viene a ser una memoria "RAM" dentro del dispositivo GPU. Las transferencias de memoria entre CPU y GPU se realizan hacia y desde esta memoria. Luego, es posible transferir dicha información a la memoria compartida.

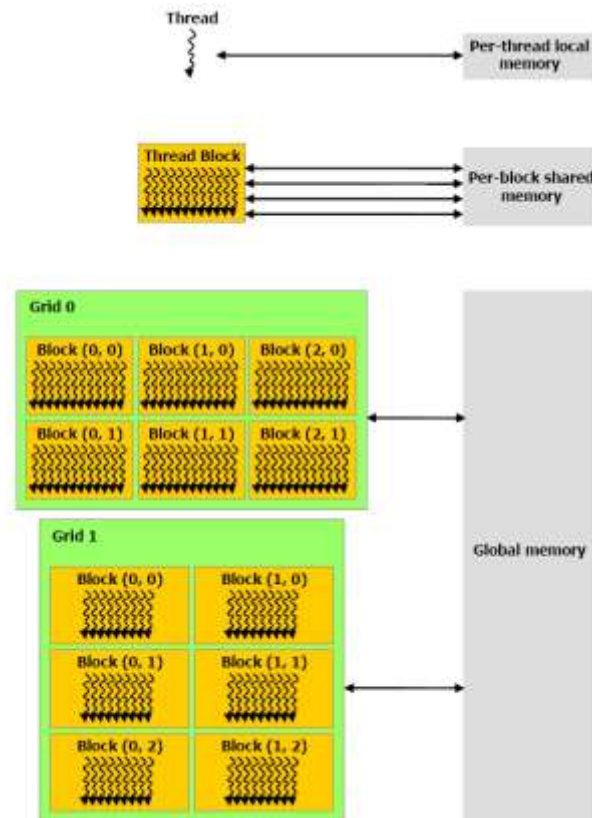


Figura 2.18. Jerarquía de memoria [URL10].

2.3.4. Flujo de procesamiento

Para procesar datos en la GPU es necesario efectuar una serie de trasposos de memoria. El flujo de procesamiento se describe a continuación:

- Copiar la información desde la memoria RAM a la memoria de la GPU.
- Ejecutar en la GPU las instrucciones que indique la CPU.
- Se tiene entonces la información procesada en la memoria de la GPU. Resta transferir desde la GPU a la RAM estos resultados.

En la Figura 2.19 se muestra un ejemplo de flujo de procesamiento CUDA [URL04].

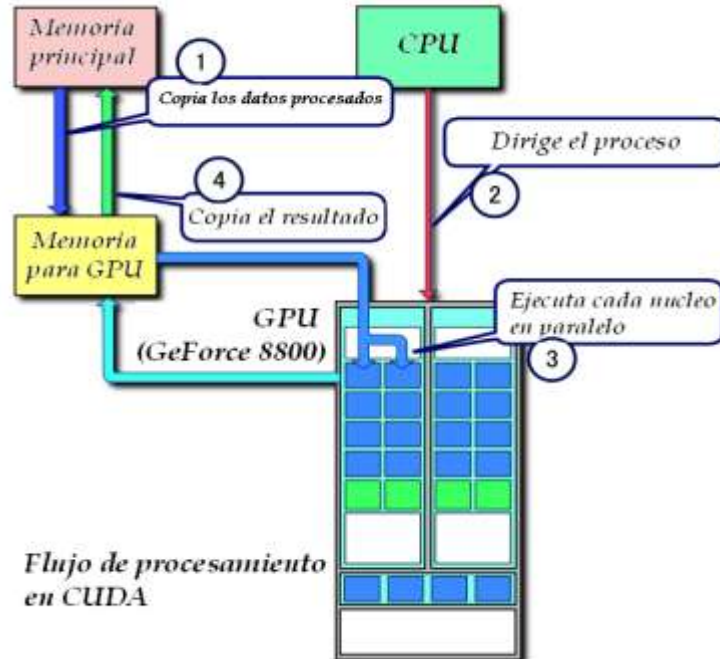


Figura 2.19: Ejemplo de flujo de procesamiento CUDA. (1) Se copian los datos de la memoria principal a la memoria de la GPU. (2) La CPU encarga el proceso a la GPU. (3) La GPU lo ejecuta en paralelo en cada núcleo. (4) Se copia el resultado de la memoria de la GPU a la memoria principal

[URL04].

Luego de investigar diversas técnicas de detección automática de pólipos, el diseño de filtros desarrollado por Möller y la herramienta de programación paralela CUDA, diseñamos un nuevo método de detección de candidatos a pólipos basado en los antecedentes. La implementación de dicho método se muestra en el capítulo a continuación.

CAPÍTULO 3: Detalles de implementación

A continuación se muestra en detalle la implementación del método propuesto. La técnica combina el criterio de Möller para la extracción de las características geométricas de los pólipos, el índice de forma propuesto por Yoshida y Nappi para la selección de los candidatos, la extracción de un corte planar para cada grupo de candidatos y un análisis de la forma resultante del mismo para la reducción de falsos positivos.

3.1. Plataforma de desarrollo

El entorno de programación que se utilizó fue Visual Studio 2008. Es un entorno de desarrollo integrado para sistemas operativos Windows. Utilizamos el lenguaje de programación C sobre la arquitectura CUDA, llamado CUDA C, para realizar operaciones que se ejecutan de forma paralela en la GPU. La aplicación fue desarrollada sobre el sistema operativo Windows y es soportada por Windows XP® o superior. No ha sido probada en versiones anteriores.

Una limitante que representa el uso de CUDA, es la utilización de un hardware específico. Para utilizar CUDA es necesario que el computador posea una tarjeta gráfica multi-núcleo NVIDIA compatible con CUDA [URL07]. Por otra parte, es importante contar con un procesador y memoria RAM suficiente (2GB o superior) para evitar un “cuello de botella”. El desarrollo de la aplicación se realizó en un computador con procesador Intel Core 2 Quad 2.40 Ghz con 3 GB de memoria RAM y una tarjeta gráfica GeForce 8800 GTS.

3.2. Implementación del método de detección de pólipos

Partiendo de la segmentación del colon en el volumen, el primer paso del método propuesto consiste en extraer las características 3D en los diferentes puntos de la pared del colon. Para esto, se calcula una aproximación de las primeras y segundas derivadas direccionales utilizando filtros diseñados siguiendo el criterio de Möller et al. [MOL98] para posteriormente calcular las curvaturas principales utilizando las formulas propuestas en Olivier Monga et al. [MON92]. En el segundo paso,

se seleccionan diferentes puntos tomando como base el índice de forma volumétrica propuesto por Yoshida y Nappi [YOS01] y se agrupan los puntos cercanos entre sí. Para el cuarto paso, se propone extraer un corte planar para cada grupo formado en el tercer paso tomando como referencia el centro del grupo y el promedio de las normales. Finalmente, en el quinto paso se realiza un análisis de forma en dichos cortes, buscando figuras cerradas en las cuales el perímetro y el área guarden una relación determinada (ver figura 3.1).

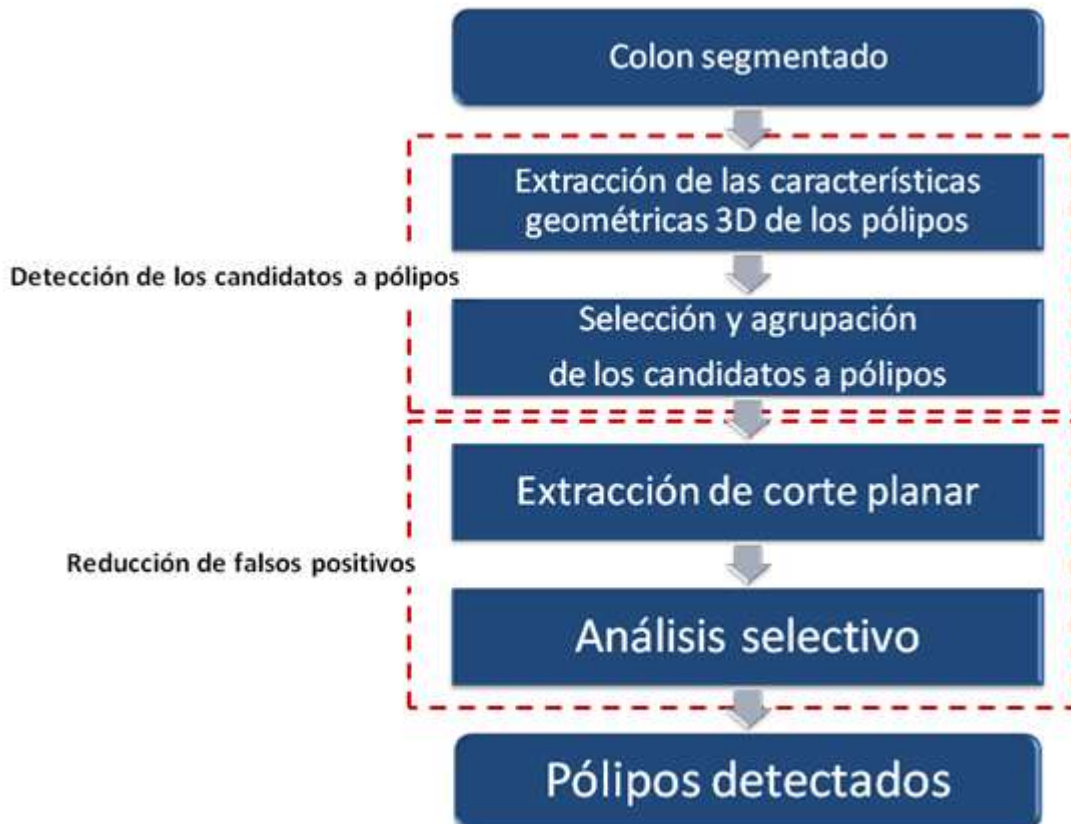


Figura 3.1: Esquema propuesto.

Para la detección de pólipos, se requiere de un colon previamente segmentado. En este trabajo se implementó un algoritmo de relleno o *floodfill*. Para el floodfill utilizamos una función `segmentar` que básicamente parte de un vóxel perteneciente al lumen del colon llamada semilla. Se verifica el valor de intensidad de sus vóxeles vecinos y posteriormente se van agregando a una cola aquellos cuyo valor sea menor a cierto umbral y no hayan sido verificados previamente. Cuando un vóxel tiene un valor de intensidad mayor al umbral, este es considerado parte de la pared de colon y su posición es guardada en un vector. Luego, se van tomando cada uno de los elementos de la cola y

se repite el mismo procedimiento hasta que la cola este vacía. Finalmente, se obtiene un vector con las posiciones de todos los vóxeles pertenecientes al colon.

3.2.1. Extracción de las características geométricas 3D de los pólipos

La extracción de las características geométricas de los pólipos comprende tres etapas. La primera consiste en **obtener una aproximación de la primera y segunda derivada**. Para obtener una aproximación utilizamos el criterio de Möller et al. [MOL98] descrito anteriormente en la Sección 2.2. Respondiendo a las tres preguntas propuestas en dicho paper:

1) ¿Qué derivada de la función original necesitamos reconstruir?

Para nuestro propósito vamos a necesitar tanto el filtro de interpolación como el de la primera y segunda derivadas.

2) ¿Qué precisión necesitamos en el proceso de reconstrucción?

Función de error de orden 3.

3) ¿A cuál espacio C^n debe pertenecer la función reconstruida?

C^2

El filtro de interpolación y el de la primera derivada los tomamos del apéndice en [MOL98]. Para el filtro de la segunda derivada realizamos una serie de operaciones descritas en el apéndice 1 de este trabajo. Elegimos la precisión y la continuidad más altas que pudimos calcular para la segunda derivada.

Si expresamos el filtro de la forma:

$$\begin{bmatrix} w_{-4} \\ w_{-3} \\ w_{-2} \\ w_{-1} \\ w_0 \\ w_1 \\ w_2 \\ w_3 \end{bmatrix} = M \begin{bmatrix} \tau^3 \\ \tau^2 \\ \tau \\ 1 \end{bmatrix}$$

La matriz M correspondiente al filtro de la segunda derivada, error de orden 3 y continuidad 2 queda expresada así:

$$\begin{bmatrix} 1 & & & \\ -\frac{1}{24} & 0 & 0 & 0 \\ \frac{11}{24} & -\frac{1}{8} & -\frac{1}{8} & -\frac{1}{24} \\ \frac{41}{24} & \frac{5}{8} & 1 & \frac{1}{6} \\ -\frac{1}{24} & \frac{4}{8} & 1 & \frac{1}{6} \\ \frac{25}{24} & -\frac{31}{8} & -\frac{13}{8} & \frac{17}{24} \\ \frac{8}{25} & -\frac{8}{11} & -\frac{8}{13} & \frac{24}{5} \\ -\frac{8}{25} & \frac{12}{11} & 0 & -\frac{3}{17} \\ \frac{41}{24} & -\frac{31}{8} & \frac{13}{8} & \frac{17}{24} \\ \frac{24}{11} & -\frac{8}{5} & -1 & \frac{24}{1} \\ -\frac{1}{24} & \frac{4}{8} & -1 & \frac{1}{6} \\ \frac{1}{24} & -\frac{1}{8} & \frac{1}{8} & -\frac{1}{24} \end{bmatrix}$$

Los filtros quedan entonces expresados de la siguiente manera:

$$\begin{bmatrix} -1 & \frac{5}{2} & -\frac{3}{2} & 0 & 0 & 0 \\ 3 & -\frac{15}{2} & \frac{9}{2} & \frac{1}{2} & \frac{1}{2} & 0 \\ -3 & \frac{15}{2} & -\frac{9}{2} & -1 & 0 & 1 \\ 1 & -\frac{5}{2} & \frac{3}{2} & \frac{1}{2} & -\frac{1}{2} & 0 \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{6} & \frac{1}{4} & 0 & 0 & 0 \\ -\frac{5}{6} & \frac{17}{2} & \frac{1}{4} & -\frac{1}{12} & \frac{1}{12} \\ \frac{5}{3} & -\frac{19}{6} & -\frac{1}{2} & \frac{4}{3} & \frac{2}{3} \\ -\frac{5}{3} & \frac{7}{2} & 0 & -\frac{5}{2} & 0 \\ \frac{5}{6} & -\frac{23}{12} & \frac{1}{2} & \frac{4}{3} & -\frac{2}{3} \\ \frac{1}{6} & \frac{5}{12} & -\frac{1}{4} & -\frac{1}{12} & \frac{1}{12} \end{bmatrix}$$

$$\begin{bmatrix} 1 & & & \\ -\frac{1}{24} & 0 & 0 & 0 \\ \frac{11}{24} & -\frac{1}{8} & -\frac{1}{8} & -\frac{1}{24} \\ \frac{41}{24} & \frac{5}{8} & 1 & \frac{1}{6} \\ -\frac{1}{24} & \frac{4}{8} & 1 & \frac{1}{6} \\ \frac{25}{24} & -\frac{31}{8} & -\frac{13}{8} & \frac{17}{24} \\ \frac{8}{25} & -\frac{8}{11} & -\frac{8}{13} & \frac{24}{5} \\ -\frac{8}{25} & \frac{12}{11} & 0 & -\frac{3}{17} \\ \frac{41}{24} & -\frac{31}{8} & \frac{13}{8} & \frac{17}{24} \\ \frac{24}{11} & -\frac{8}{5} & -1 & \frac{24}{1} \\ -\frac{1}{24} & \frac{4}{8} & -1 & \frac{1}{6} \\ \frac{1}{24} & -\frac{1}{8} & \frac{1}{8} & -\frac{1}{24} \end{bmatrix}$$

Interpolación

Primera derivada

Segunda derivada

Una vez calculados todos los w_k , lo siguiente consiste en obtener el kernel unidimensional. Recordemos que w_k está definido de la siguiente manera:

$$w_k = w_k(\tau) = w(\tau + k)$$

En la Figura 4.2 podemos observar cómo está definido el filtro.

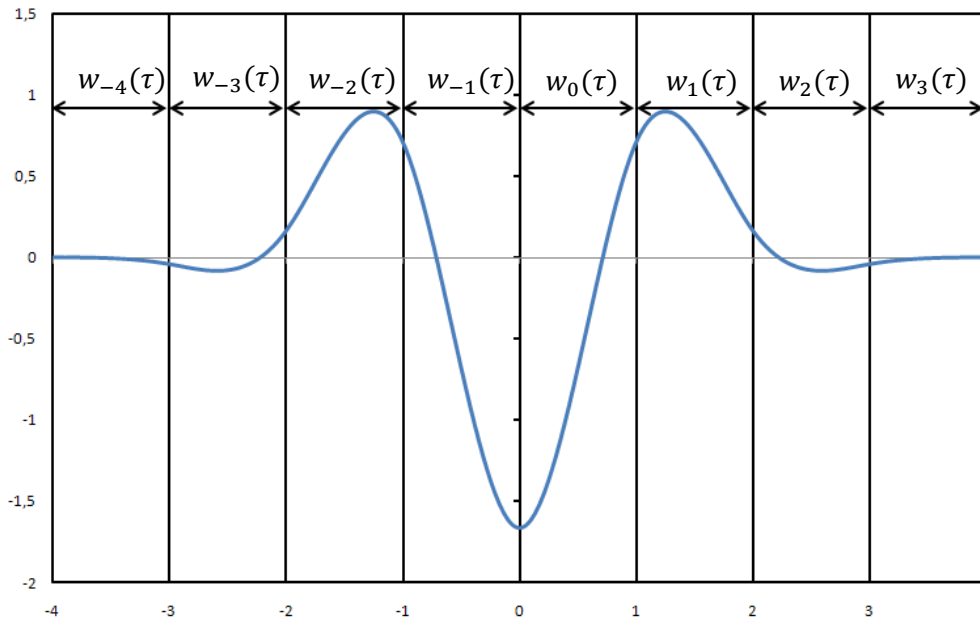


Figura 3.2: El filtro w está dividido en ocho trozos w_k . Si consideramos que $w_k = w_k(\tau) = w(\tau + k)$, entonces cada trozo w_k queda evaluado entre 0 y 1.

Tomando en cuenta lo anterior, construimos entonces el kernel unidimensional. Para ello se toman n muestras equidistantes del filtro w . En detalle, primero definimos una función `obtener_filtro` explicada a continuación:

Algoritmo 1

```

función obtener_filtro(tamaño)
//calculamos la distancia entre las muestras. Se suma uno para descartar los
extremos.
    Distancia ← cantidad_de_divisiones/(tamaño +1);
//se toma una muestra y se calcula la siguiente
    Para i←0 hasta tamaño hacer
        posición_muestra_sig ← posición_muestra_ant + distancia;
        k ← parte_entera(posición_muestra_sig);
        t ← parte_decimal(posición_muestra_sig);
        muestra[tamaño-i] ←  $w_k(t)$ 

```

Es importante tener en cuenta por la definición de convolución que estas muestras se van a colocar en el kernel de forma invertida, es decir, la última se va a colocar en la primera posición y así sucesivamente.

Una vez obtenidos los kernels unidimensionales de los filtros de interpolación, primera derivada y segunda derivada, construimos a continuación el kernel 3D. En principio es necesario decidir qué operación se quiere hacer en cada una de las direcciones x , y , z . Por ejemplo, para construir el filtro que calcula la derivada parcial con respecto a x , en la dirección de x se debe aplicar la derivada, en la dirección de y se debe aplicar la interpolación así como también en la dirección de z . Luego, se hallan los valores del kernel de convolución W que será una matriz tridimensional $n \times n \times n$ construida de la siguiente forma:

$$\forall i, j, k \in \mathbb{N}, \quad 1 \leq i, j, k \leq n : W[i, j, k] = w_x[i] \cdot w_y[j] \cdot w_z[k]$$

Donde w_x , w_y y w_z son vectores contruidos con los valores de los kernels unidimensionales de las operaciones (interpolación, primera derivada o segunda deriva) que se requieren en cada dirección (x , y ó z). Véase el siguiente algoritmo:

Algoritmo 2

```

función construir_kernel(filtro_unidimensional_x, filtro_unidimensional_y,
filtro_unidimensional_z)
//cada filtro está guardado en una clase que contiene su tamaño y un arreglo con el
filtro
    suma←0
    Para z←0 hasta z< tamaño_filtro hacer
        Para y←0 hasta y< tamaño_filtro hacer
            Para x←0 hasta x< tamaño_filtro hacer
                matriz_kernel[z][y][x]← filtro_unidimensional_z.arreglo[z] ×
                filtro_unidimensional_y.arreglo[y] ×
                filtro_unidimensional_x.arreglo[x]
                suma←suma + valor_absoluto(matriz_kernel[z][y][x]);
    Para z←0 hasta tamaño_filtro hacer
        Para y←0 hasta tamaño_filtro hacer
            Para x←0 hasta tamaño_filtro hacer
                matriz_kernel[z][y][x] ← matriz_kernel[z][y][x]/suma;

```

Después de obtener el kernel, se procede a realizar la convolución. En el caso discreto la fórmula de la convolución se define de la siguiente forma:

$$f[t] * w[t] = \sum_k f[k] \cdot w[t - k]$$

Computacionalmente, la convolución se realiza utilizando 2 matrices. Llamaremos a la primera matriz f y a la segunda kernel de convolución. Primero, se hace coincidir el centro del kernel con la posición de la matriz a la cual se le quiere hallar el valor de la función de convolución. Luego, se multiplica cada elemento del kernel por su correspondiente en la matriz f y se suman todos los resultados. Esta suma es el valor de la convolución en ese punto en particular de la matriz f . Este proceso se repite con todos los puntos en la matriz f de los cuales se desea conocer el valor de convolución. Esta parte del proceso requiere de muchos cálculos, por lo que hemos utilizado programación paralela en CUDA para reducir el tiempo de respuesta. Para lograr esto, se debe dividir el proceso total en procesos más pequeños (hilos) con la misma estructura de manera tal que se puedan ejecutar paralelamente sin afectar el resultado final. Cada uno de estos hilos será ejecutado en un núcleo de procesamiento diferente. Se debe tener en cuenta que CUDA restringe el tiempo de ejecución de cada hilo, por lo que estos no deberían tener ciclos muy largos. Teniendo en cuenta todos estos aspectos, cada hilo calculará un término de la convolución (la multiplicación de un elemento en el kernel por un elemento en el volumen) para cada vóxel perteneciente a la pared del colon y lo sumará al total. Para estos efectos, las posiciones de los vóxeles que se encuentran en la superficie de la pared del colon estarán guardadas en un vector llamado C . La paralelización se realizará a lo largo de los elementos en el vector C , es decir, se estarán ejecutando paralelamente los hilos que van a calcular el mismo término de convolución para todos los componentes en C (ver Figura 3.3).

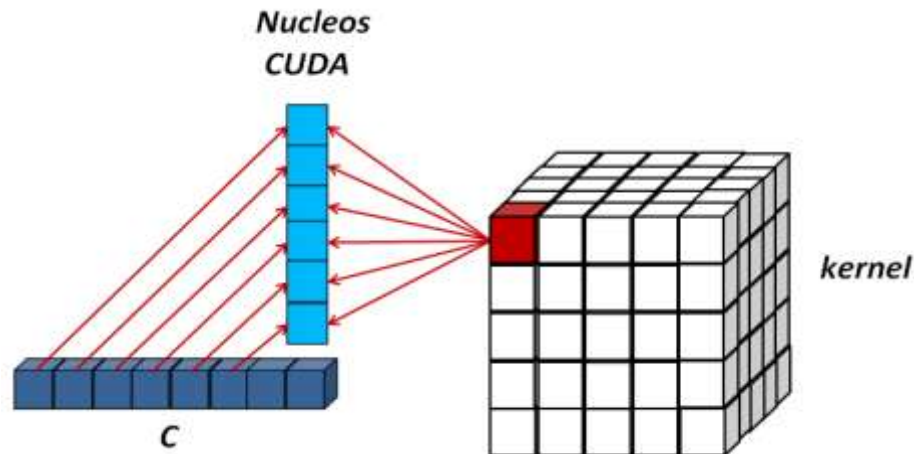


Figura 3.3: C es el conjunto de vóxeles que pertenecen al colon, cada núcleo CUDA tiene un vóxel que pertenecen a C y calcula el término n-ésimo de la convolución para dicho vóxel.

Todos los hilos que calculan el mismo término de convolución deben finalizar su ejecución para empezar a calcular el siguiente término para todos los componentes de C. Este proceso se repite hasta hallar el valor total de la sumatoria de todos los términos de la convolución.

Algoritmo 3

//x y z son coordenadas de un vóxel perteneciente a la pared del colon

//la función calcula el termino de la convolución i,j,k con respecto a las coordenadas x y z

función calcular_derivadas_paralelo<<x,y,z>>(i,j,k)

 sumatoria[x][y][z] ← kernel [i][j][k] × volumen[x-(tamaño_kernel/2)+i] [y-(tamaño_kernel/2)+j][z-(tamaño_kernel/2)+k] + sumatoria[x][y][z];

fin función

En la implementación del método propuesto, se calculan nueve convoluciones en total. Tres de ellas corresponden al cálculo de las derivadas parciales de primer orden ($\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$) y las otras

seis al cálculo de las derivadas parciales de segundo orden ($\frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial y^2}, \frac{\partial^2 f}{\partial z^2}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial y \partial x}, \frac{\partial^2 f}{\partial x \partial z} =$

$$\frac{\partial^2 f}{\partial z \partial x}, \frac{\partial^2 f}{\partial y \partial z} = \frac{\partial^2 f}{\partial z \partial y})$$

Una vez que tenemos las primeras y segundas derivadas de orden parcial, la segunda etapa consiste en **calcular la primera y segunda curvatura**. Para esto, usamos la forma propuesta por

Olivier Monga et al. [MON92] donde sugieren la siguiente fórmula para calcular la curvatura principal k_1 y k_2 :

$$K_i = \frac{\mathbf{h}^T \mathbf{H} \mathbf{h} + \mathbf{f}^T \mathbf{H} \mathbf{f} \pm \sqrt{(\mathbf{h}^T \mathbf{H} \mathbf{h} - \mathbf{f}^T \mathbf{H} \mathbf{f})^2 + 4(\mathbf{h}^T \mathbf{H} \mathbf{f})^2}}{2}$$

\mathbf{H} es la matriz hessiana; \mathbf{h} y \mathbf{f} son columnas de la matriz de rotación P definida de la siguiente manera:

$$P = \begin{pmatrix} \frac{f_x}{\delta} & \frac{f_y}{\gamma} & \frac{f_z f_x}{\gamma \delta} \\ \frac{f_y}{\delta} & -\frac{f_x}{\gamma} & \frac{f_y f_z}{\gamma \delta} \\ \frac{f_z}{\delta} & 0 & -\frac{\gamma}{\delta} \end{pmatrix} = (\mathbf{g} \mathbf{h} \mathbf{f})$$

$$\gamma = \sqrt{f_x^2 + f_y^2} \quad \gamma \delta = \sqrt{f_x^2 + f_y^2 + f_z^2}$$

En la tercera etapa del cálculo de los atributos geométricos, se calcula para cada vóxel en la superficie de la pared del colon el **índice de forma volumétrica** propuesto por Yoshida y Nappi [YOS01]. Su fórmula es la siguiente:

$$SI(p) = \frac{1}{2} - \frac{1}{\pi} \arctan \frac{\kappa_1(p) + \kappa_2(p)}{\kappa_1(p) - \kappa_2(p)}$$

Donde $\kappa_1(p)$ y $\kappa_2(p)$ representa las curvaturas principales en la vecindad del punto p . El índice de la forma volumétrica indica el tipo de forma, tal como se ve en la Figura 2.2 del capítulo 2.

3.2.2. Selección y agrupación de los candidatos a pólipos

Los vóxeles pertenecientes a la pared del colon están almacenados en un vector con sus respectivas curvaturas principales calculadas. Para seleccionar cuales son candidatos a pólipos se definen los valores máximo y mínimo del índice de forma volumétrica para seleccionar los vóxeles

que se sospecha pudieran encontrarse sobre la superficie de un pólipo. La posición de los vóxeles seleccionados se guardará en un vector, al cual llamaremos *candidatos*.

Sobre la superficie de un pólipo, es natural que se encuentren varios vóxeles seleccionados. Sin embargo es posible que no todos ellos se encuentren conectados. A continuación se formaran grupos de los vóxeles seleccionados. Se espera que cada grupo contenga todos los vóxeles seleccionados que se encuentren sobre la superficie de un mismo pólipo (aunque no todos estén conectados). Para iniciar el proceso de formación de un grupo (al cual llamaremos *grupo*), tomamos un vóxel del vector *candidatos*, lo eliminamos del mismo y lo agregamos a una cola auxiliar que llamaremos *cola*, lo agregamos al *grupo* y lo marcamos como revisado. Luego, comenzará el ciclo de búsqueda de vóxeles seleccionados que se encuentren cercanos entre sí. Para esto, se toma el primer elemento de la *cola* y se la región cúbica de tamaño *lado x lado x lado* que rodea dicho elemento (siendo *lado* un valor dado). Si dentro de esta región se encuentra algún otro voxel que se encuentre en *candidatos* (y que no haya sido revisado previamente), este se agrega a *cola* y se elimina de *candidatos*. Seguidamente, se van tomando cada uno de los elementos de la cola y se realiza el mismo procedimiento hasta que la *cola* este vacía. Esto significa que el primer grupo se ha formado y se agrega a un vector de grupos que llamaremos *candidatos_agrupados*. Para formar los demás grupos, se repite el proceso de formación de grupos hasta que vector *candidatos* quede vacío.

Para comprenderlo mejor, obsérvese el siguiente algoritmo:

Algoritmo 4

función agrupar_candidatos(candidatos, lado)//recibe el vector candidatos y el tamaño del lado

Para $i \leftarrow 0$ **hasta** $\text{colon}_s \rightarrow \text{tamaño}()$ **hacer**

v ← candidatos[i]

Si (!v.revisado) **entonces**

cola → **encolar**(v) //se introduce la semilla a la cola

grupo → **incluir**(v);

v.revisado ← **verdadero** //se marca como revisado el vóxel

```

Mientras (cola.no_vacia) hacer
    n ← cola→frente() //aquí se obtiene el vóxel al frente
    de la cola
    cola→desencolar() //se saca el vóxel de la cola

    Si (índices_dentro_de_bordes) entonces

        Para i← -lado hasta i ≤ lado hacer
            Para j← -lado hasta j ≤ lado hacer
                Para k← -lado hasta k ≤ lado hacer
                    aux.x←n.x+i;
                    aux.y←n.y+j;
                    aux.z←n.z+k;

                    Si (aux.seleccionado^!aux.revisado)
                    entonces
                        grupo→incluir(aux);
                        cola→encolar(aux);

    candidatos_agrupados→incluir(grupo);

```

3.2.3. Extracción de corte planar

El corte planar se extrae con el fin de analizar la forma del mismo para cada grupo de candidatos y en base a esto seleccionar los mejores candidatos a pólipos y descartar la mayor cantidad de falsos positivos. En primer lugar, se calcula el promedio de las normales de los vóxeles que forman un grupo y el centro volumétrico del mismo. Luego, hallamos la ecuación del plano que pasa por el centro volumétrico y es perpendicular a la media normal.

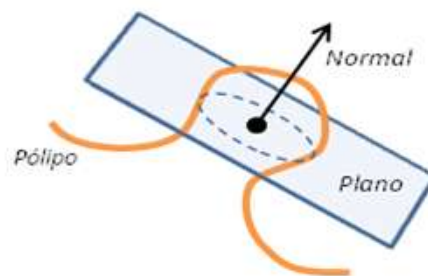


Figura 3.4. Corte planar.

Para obtener las muestras en el plano perpendicular a \vec{n} que pasa por el punto p , necesitamos hallar 2 vectores \vec{v}_1, \vec{v}_2 tal que $(\vec{n}, \vec{v}_1, \vec{v}_2)$ formen una base ortonormal de \mathbb{R}^3 . Entonces cualquier punto $p_{\alpha,\beta}$ en el plano anteriormente mencionado se puede expresar de la siguiente manera:

$$p_{\alpha,\beta} = p + \alpha\vec{v}_1 + \beta\vec{v}_2$$

Tomemos como referencia el vector canónico \vec{j} . Si $\vec{n} = \vec{j}$ entonces podemos tomar

$$\vec{v}_1 = \vec{i} \quad \text{y} \quad \vec{v}_2 = \vec{k}$$

Si $\vec{n} \neq \vec{j}$ entonces podemos tomar

$$\vec{v}_1 = \frac{\vec{n} \times \vec{j}}{\|\vec{n} \times \vec{j}\|} \quad \text{y} \quad \vec{v}_2 = \frac{\vec{n} \times \vec{v}_1}{\|\vec{n} \times \vec{v}_1\|}$$

Una vez hallados \vec{v}_1 y \vec{v}_2 , se toman las muestras en todos los puntos $p_{\alpha,\beta}$ tal que

$$-N \leq \alpha, \beta \leq N$$

De tal forma que obtenemos una porción cuadrada del plano alrededor del punto p de lado $2N + 1$. Para hallar el valor de la muestra es necesario realizar una interpolación trilineal. Seguidamente, el valor de cada muestra se guarda en un arreglo bidimensional de tamaño $2N + 1 \times 2N + 1$.

Algoritmo 5

//la siguiente función se realiza para cada grupo de candidatos

```
función calcular_direcciones_plano2()
    Si (normal= vector_canonico_j) entonces
        direccion1←vector_canonico_i
        direccion2←vector_canonico_k
    sino
        direccion1← normal × vector_canonico_j
        direccion2← direccion1 × normal
```

Algoritmo 6

```

función obtener_planos_corte(lado)
  para x←0 hasta x<lado hacer
    para y←0 hasta y<lado hacer
      c←lado/2
      x_relativo←x-c;
      y_relativo←y-c;
      //se halla el punto donde se va a muestrear
      p ← centro_candidato+x_relativo×direccion1
        +y_relativo×direccion2
      // si ninguno de los puntos a evaluar
      //para realizar la interpolación trilinear
      //se sale de los bordes
      Si (dentro_de_bordes) entonces
        //se hace una interpolación trilinear evaluando
        // en los 8 vóxeles alrededor de p
        plano[y][z]←interpolación_lineal(p)

```

3.2.4. Análisis selectivo

El análisis selectivo consiste en seleccionar aquellos candidatos que verdaderamente son pólipos y descartar los falsos positivos. Para ello, partimos de la suposición de que si un grupo de candidatos está sobre la superficie de un pólipo, el corte planar del mismo forma una figura cerrada, por lo contrario, los cortes planares sobre los grupos de falsos positivos tomaran diversas formas (ver Figura 3.5).

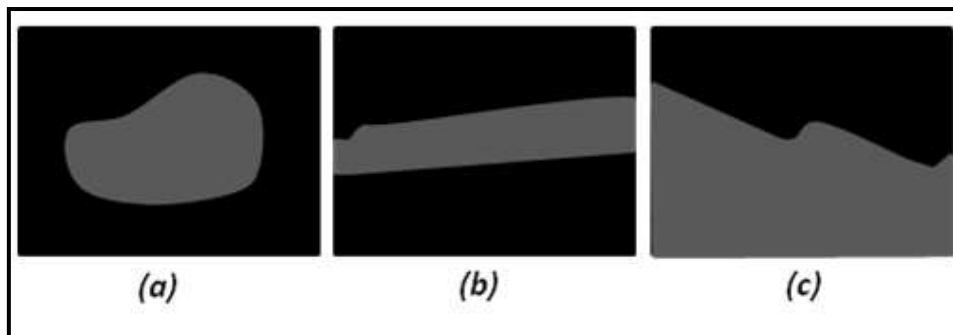


Figura 3.5: (a) Es fuerte candidato a pólipo. (b) es un pliegue. (c) es la pared del colón. El área negra corresponde al interior del colón o lumen, mientras que la gris representa la figura sólida resultante del corte planar.

Tomando como punto de partida el centro del corte, se realiza el recorrido de todas las muestras mayores a un cierto umbral (tono claro) que se encuentran adyacentes entre sí. Si una vez recorridas todas las muestras no se alcanza el borde del corte planar, entonces forma una figura cerrada y por tanto un posible candidato a pólipo. Si por el contrario el recorrido alcanza el borde, entonces la figura representa un pliegue o una pared del colon por lo que es descartada.

Una figura cerrada no representa necesariamente un buen candidato a pólipo. Para complementar este criterio, tengamos en cuenta la siguiente proposición:

Consideremos una figura bidimensional de perímetro p y área A . Razonemos la relación $R_{(p,A)}$ dada por la siguiente fórmula:

$$R_{(p,A)} = \frac{p^2}{A}$$

El valor de esta fórmula se hace relativamente pequeño cuando el área de la figura es grande con relación a su perímetro. Cuando el área es pequeña en comparación al perímetro, el valor de la fórmula se hace más grande. Observemos los siguientes ejemplos.

$$\text{Para un círculo, } R_{(p,A)} = \frac{p^2}{A} = \frac{(2\pi r)^2}{\pi r^2} = 4\pi \approx 12,56$$

$$\text{Para un cuadrado, } R_{(p,A)} = \frac{p^2}{A} = \frac{(4l)^2}{l^2} = 16$$

$$\text{Para un rectángulo con } L=3 \text{ y } l=2, R_{(p,A)} = \frac{p^2}{A} = \frac{(2L+2l)^2}{L \cdot l} = \frac{(2 \cdot 3 + 2 \cdot 2)^2}{3 \cdot 2} = \frac{100}{6} \approx 16,66$$

$$\text{Para un rectángulo con } L=10 \text{ y } l=1, R_{(p,A)} = \frac{p^2}{A} = \frac{(2L+2l)^2}{L \cdot l} = \frac{(2 \cdot 10 + 2 \cdot 1)^2}{10 \cdot 1} = \frac{484}{10} = 48,4$$

Como se puede observar, esta relación es un buen indicador de qué tan alargada es la figura. Mientras mayor sea el valor de la relación, mas alargada es la figura (ver la figura 3.6).

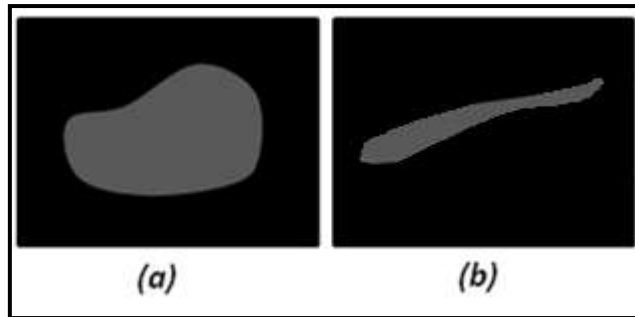


Figura 3.6: (a) Fuerte candidato a pólipo. (b) Candidato a descartar.

Podemos utilizar esta relación para descartar los candidatos cuya figura central en el corte planar sea muy alargada. Basta contar los píxeles internos a la figura y los píxeles en el borde de la figura. Se toma la suma de píxeles internos como el área de la figura y la suma de los píxeles en el borde como el perímetro. Luego se calcula $R_{(p,A)}$ y se descartan los candidatos con $R_{(p,A)}$ mayores a un valor dado. El valor utilizado en este caso fue 30(ver apéndice 7).

El algoritmo para realizar el análisis selectivo quedaría entonces de la siguiente manera:

Algoritmo 5

```

función recorrer_candidato(pixel_central)
    area←0
    perimetro←0
    cola→encolar(pixel_central)
    mientras (cola_no_vacia) hacer
        n←cola→frente() //se obtiene el primer pixel de la cola
        cola→desencolar()
        Si (esta_dentro_de_bordes=verdadero) entonces
            Si (esta_revisado=falso) entonces
                esta_revisado←verdadero
                Si (valor_pixel≥umbral) entonces
                    cola→encolar(vóxeles_vecinos)
                    area++
                sino
                    perimetro++
            sino
                ← falso //llego al borde
    RpA←(perimetro x perimetro)/area
    si (RpA>30)

```

```
        ← falso //forma muy alargada
sino
        ← verdadero //fuerte candidato
```

3.2.5. Pólipos detectados

Una vez hecho el análisis selectivo, quedan descartados muchos de los falsos positivos. La aplicación guarda en un vector las coordenadas del centro y el vector normal de cada uno de los candidatos a pólipos encontrados.

Luego de guardar los resultados, durante un análisis de los mismos pudimos notar que la selección del tamaño del filtro afectaba directamente el tamaño de las protuberancias que se detectaron. Idealmente, decidimos elegir entonces dos tamaños de filtro, uno relativamente pequeño (13) y otro un poco más grande (21). De esta manera son detectados diversos tamaños de protuberancias. En definitiva, luego de ejecutar el proceso con ambos filtros, los vectores de resultados son comparados para eliminar redundancias y finalmente el programa arroja un archivo con las coordenadas del centro y los respectivos vectores normales de los pólipos encontrados.

CAPÍTULO 4: Pruebas y resultados

Una vez terminada la etapa de diseño e implementación del método de detección, es necesario poner a prueba la aplicación para de esta forma evaluar el rendimiento, la eficiencia y precisión, y así determinar si se cumplieron o no los objetivos anteriormente establecidos. En este capítulo presentaremos las pruebas realizadas y los resultados obtenidos.

4.1. Descripción del ambiente de pruebas

Para realizar las pruebas, requerimos de un hardware particular que pudiera ejecutar la aplicación. A continuación, en la Tabla 4.1 se muestran las especificaciones del hardware utilizado:

Arquitectura	Procesador	RAM	Sistema Operativo
1	Intel Core 2 Quad 2.40 Ghz	3.00 GB	Windows XP 32 bits
2	Intel Core i3 3.07 Ghz	4.00 GB	Windows 7 64 bits

Tabla 4.1: Arquitecturas utilizadas en el ambiente de pruebas

Recordemos que para utilizar CUDA es necesario que el computador posea una tarjeta gráfica multi-núcleo NVIDIA compatible con CUDA [URL07]. En la Tabla 4.2 se muestran las tarjetas gráficas utilizadas.

Arquitectura	Tarjeta gráfica	# Cores	Ancho de banda	Memoria	Capacidad
1	GeForce 8800 GTS	96	64 GB/s	640 MB	2.0
2	GeForce GTX 470	448	133.9 GB/s	1280 MB	2.0

Tabla 4.2: Tarjetas gráficas de cada arquitectura usada en el ambiente de pruebas

Los volúmenes utilizados para las pruebas tienen el formato de archivo RAW. La precisión de muestreo varía. Los datos provienen de tomografías. Las características de estos volúmenes son presentadas en la Tabla 4.3 mostrada a continuación:

Volumen	Dimensiones (WxHxD)	Tamaño MB
1	512x512x442	221
2	512x512x544	272
3	512x512x548	274
4	512x512x469	234
5	512x512x463	231
6	512x512x426	213

Tabla 4.3: Características de los volúmenes utilizados en las distintas pruebas.

4.2. Descripción de las pruebas

La aplicación se corrió varias veces para cada volumen, esto a fin de comparar resultados con diversos tamaños de filtros (entre 11-29) y determinar en general cual era el más acertado. Para medir la eficiencia del código ejecutado en CUDA, realizamos pruebas en distintos hardware. De esta manera determinamos que tan rápido podía ejecutarse el código de forma paralela según la capacidad de la tarjeta gráfica utilizada. Para las mediciones de tiempo se utilizó la biblioteca timer.h tanto para las implementaciones hechas en CUDA como para CPU.

4.2.2. Tablas de resultados y discusiones

En la Tabla 4.4 se muestra para cada volumen la cantidad de vóxeles que se encuentran sobre la superficie de la pared del colon, así como la cantidad de vóxeles seleccionados por cada tamaño de kernel utilizado.

Volumen	Vóxeles en la pared del colon	Vóxeles seleccionados	
		Kernel tamaño 13	Kernel tamaño 21
1	351.483	5.197	4.327
2	394.514	5.314	4.826
3	1.024.677	12.073	11.789
4	357.356	3.325	3.300
5	393.000	7.389	7.032
6	369.910	6.986	7.211

Tabla 4.4: Conteo de vóxeles por volumen.

Volumen	Arquitectura	Tamaño del kernel	Tiempo	
			CPU	GPU
1	1	13	110	34
		21	537	136
	2	13	89	3
		21	484	10
2	1	13	102	37
		21	470	123
	2	13	78	4
		21	401	10
3	1	13	247	93
		21	1195	372
	2	13	200	7
		21	1077	20
4	1	13	85	34
		21	409	126
	2	13	69	3
		21	369	10
5	1	13	94	39
		21	484	169
	2	13	76	4
		21	436	11
6	1	13	89	36
		21	436	140
	2	13	72	4
		21	393	10

Tabla 4.5: Tabla de tiempos.

Volumen 1: El análisis arrojó 0 (cero) candidatos a pólipos por lo que posiblemente se trata de un colon sano.

Volumen 2: El análisis 2 mostró 2 candidatos a pólipos, haciendo un análisis visual de la forma ambos son técnicamente candidatos a pólipos, aunque pudiera tratarse de simple residuo (ver imágenes apéndice 2).

Volumen 3: En el estudio se detectaron 18 candidatos de los cuales solo 2 se asemejan a simples pliegues (ver imágenes apéndice 3).

Volumen 4: El análisis arrojó 2 candidatos a pólipos aunque uno de ellos es un pliegue (ver imágenes apéndice 4).

Volumen 5: El estudio mostró 5 candidatos, a simple vista 3 de ellos no parecen ser buenos candidatos (ver imágenes apéndice 5).

Volumen 6: El estudio detectó 3 candidatos a pólipos, solo uno de ellos parece ser potencialmente un buen candidato (ver imágenes apéndice 6).

De los resultados mostrados en las tablas anteriores se puede notar que los tiempos en GPU son mucho menores a los de CPU, esto debido a que los cálculos se hacen en forma paralela. Podemos observar además que mientras más grande es el kernel mas tiempo toma la ejecución. La arquitectura es también un factor importante debido a que influye la capacidad de cómputo del procesador y de la tarjeta gráfica. Analizando la Tabla 4.5 mostrada anteriormente, para la arquitectura 1 los tiempos en GPU mejoran entre un 62% y un 75% con respecto a los tiempos de corrida en CPU. Para la arquitectura 2 mejoran entre un 96% y 98%.

4.2.3. Sensibilidad y falsos positivos

Además de los tiempos de ejecución, es importante evaluar la sensibilidad y la cantidad de falsos positivos. La sensibilidad representa el porcentaje de pólipos detectados con respecto a los existentes. Los falsos positivos son aquellas protuberancias detectadas que no son pólipos reales. Para evaluar lo anterior, es necesario el análisis médico de los resultados obtenidos, recordemos que los pólipos resultantes de nuestro programa no son más que candidatos, solo un especialista puede determinar finalmente si son pólipos o no.

$$\text{Sensibilidad} = \frac{VP}{VP + FN}$$

El volumen 5 y el volumen 6 son tomografías de una misma persona, una tomada boca abajo (volumen 5) y otra boca arriba (volumen 6). Esto se realiza generalmente en personas cuya tomografía presenta mucho contraste o residuos, para así estar seguros de que todas las partes del colon han sido exploradas (ver Figura 4.1, Figura 4.2, Figura 4.3 y Figura 4.4).

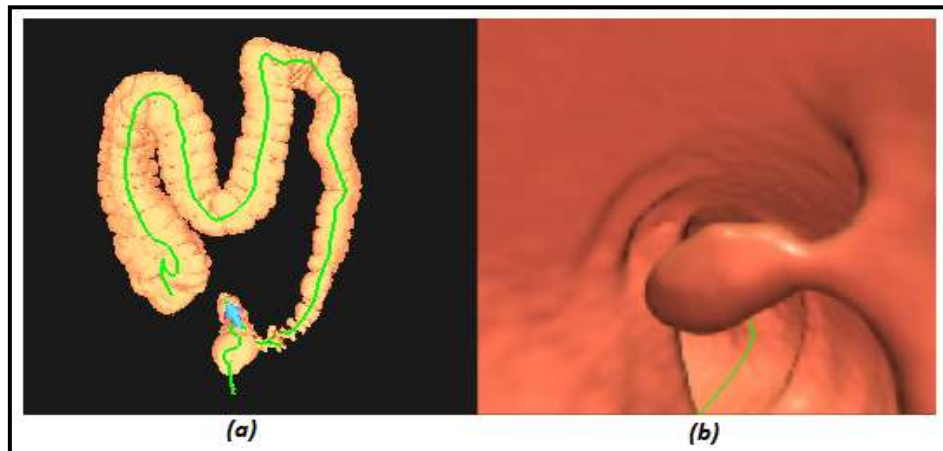


Figura 4.1: Resultados del análisis médico de la colonoscopia virtual prono (volumen 5). (a) la flecha azul indica la posición del pólipo dentro del colon. (b) el pólipo detectado.



Figura 4.2: Resultados obtenidos del programa. (a) el punto rojo indica la posición del pólipo dentro del colon. (b) pólipo encontrado.

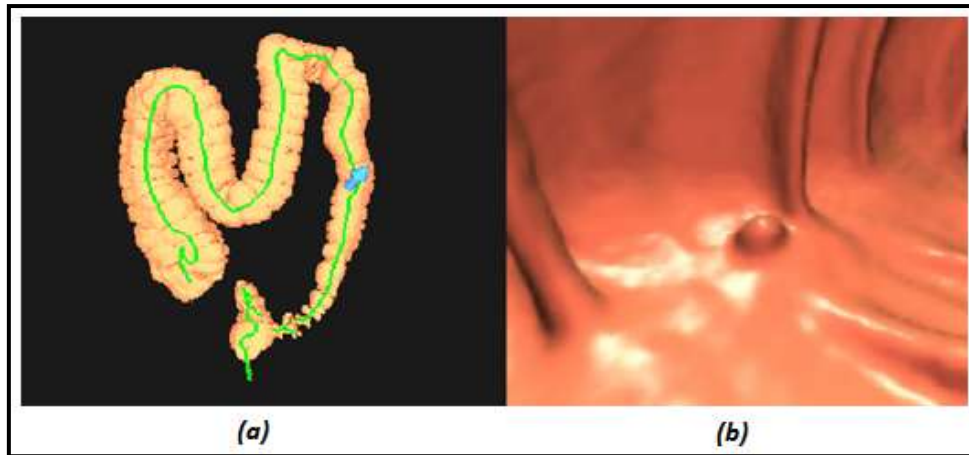


Figura 4.3: Resultados del análisis médico de la colonoscopia virtual supino (volumen 6). (a) la flecha azul indica la posición del pólipo dentro del colon. (b) el pólipo detectado.

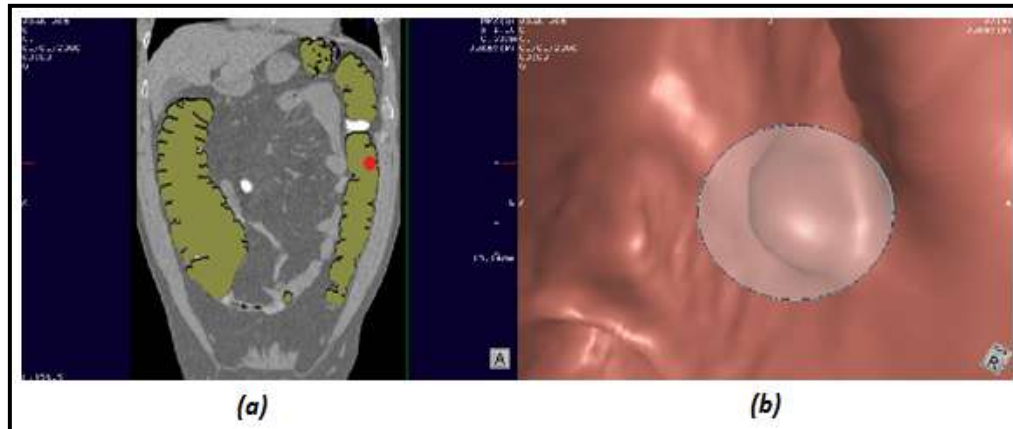


Figura 4.4: Resultados obtenidos del programa. (a) el punto rojo indica la posición del pólipo dentro del colon. (b) pólipo encontrado.

Las tablas a continuación muestran la sensibilidad y la cantidad de falsos positivos según los resultados médicos de los volúmenes 5 y 6 pertenecientes a un mismo colon.

Volumen 5

Sensibilidad	Falsos positivos
100%	80%

Tabla 4.6: Tabla de resultados para el volumen 5.

Volumen 6

Sensibilidad	Falsos positivos
100%	66,6%

Tabla 4.7: Tabla de resultados para el volumen 6.

A pesar de que en el colón se detectaron 2 pólipos, la sensibilidad en ambos casos es de 100%. Esto se debe a que solo un pólipo es visible en una tomografía (decúbito prono²⁰) y el otro es visible en la otra tomografía (decúbito supino²¹). Los falsos positivos detectados pudieran ser consecuencia de la segmentación utilizada.

²⁰ **Decúbito prono** es una posición anatómica del cuerpo humano que se caracteriza por una posición corporal tendido boca abajo, el cuello en posición neutra, miembros superiores extendidos pegados al tronco y extremidades inferiores también extendidas.

²¹ **Decúbito supino** es una posición anatómica del cuerpo humano que se caracteriza por una posición corporal acostado boca arriba, el cuello en posición neutra, miembros superiores extendidos pegados al tronco y extremidades inferiores también extendidas.

CAPÍTULO 5: Conclusiones y trabajos futuros

5.1. Conclusiones

En este trabajo se implementó un algoritmo de detección automática de pólipos, basado en análisis de curvatura, con aceleración en GPU en el cálculo aproximado de las derivadas usando convolución. Para tener tiempos y resultados visuales de referencia, se implementó además el algoritmo en CPU. Analizando ambos resultados se observa que la programación paralela mejora notablemente el tiempo de respuesta, entre 62% y 98% dependiendo de la arquitectura utilizada.

El algoritmo propuesto combina el criterio de Möller et al. [MOL98], el índice de forma propuesto por Yoshida y Nappi [YOS01], la extracción de un corte planar para cada grupo de candidatos y un análisis de la forma resultante del mismo. Los candidatos finales a pesar de ser técnicamente buenos candidatos, no aseguran un porcentaje de sensibilidad ni una tasa de falsos positivos del programa ya que no fueron analizados por médicos especialistas.

Los candidatos a pólipos arrojados por el programa parecen ser, en su mayoría, fuertes candidatos a pólipos (según su forma). Las protuberancias detectadas varían en dimensión; esto debido a que el tamaño del kernel utilizado influye directamente en el tamaño de las protuberancias detectadas; por consiguiente, elegimos un filtro relativamente pequeño y otro de mayor tamaño para detectar distintos tamaños de pólipos.

Finalmente, cabe señalar que los pólipos planos aparentemente pueden ser detectados, ya que entre las pruebas realizadas se detectaron protuberancias que se asemejan a este tipo de pólipos. Sin embargo es necesario realizar más pruebas con pólipos de este tipo.

5.2. Trabajos futuros

Como trabajo futuro se propone realizar más pruebas y analizar los resultados con médicos especialistas para así determinar un porcentaje concreto de sensibilidad y una tasa de falsos positivos. También sería interesante realizar pruebas con pólipos planos para determinar si el programa los detecta.

Por otro lado, se sugiere probar el algoritmo haciendo previamente una mejor segmentación de los datos. Además, analizando dicha segmentación sería ideal elegir un mejor umbral para el análisis de la forma. También se podría utilizar un rango de similitud en vez de un umbral, o aplicar un filtro paso bajo (Prewitt, gradiente direccional, etc.) antes de hacer el análisis de la forma y comparar los resultados.

Se puede aplicar el análisis de la forma del corte planar a otros métodos de detección (por ejemplo, el desarrollado por Kitasaka [KIT05]).

Además, se podría utilizar el corte planar para la segmentación de los pólipos. La idea sería mover el centro del corte en la dirección de la normal en ambos sentidos y obtener varios cortes mientras la condición del análisis selectivo se cumpla y seleccionar los vóxeles correspondientes a las muestras de la figura central del corte.

El criterio anterior se podría mejorar, si a partir del centro del último plano que cumplió con la condición del análisis discriminante, se obtienen diferentes planos variando la inclinación del plano original. Con esto se podrían obtener más vóxeles de pólipo para la segmentación.

Posterior a la segmentación de los pólipos, también podría realizarse un estudio para la clasificación y caracterización de los candidatos a pólipos utilizando descriptores (como por ejemplo la covarianza) que serían utilizados por una *Máquina de Soporte Vectorial (SVM – Support Vector Machine)* para una fase adicional de reducción de falsos positivos.

Referencias

- [BIT05] Bitter I., B. Aslam, A. Huang y R. Summers. (2005). *Candidate Determination for Computer Aided Detection of Colon Polyps*. En Amini, A. y Manduca A. (editores): *Medical Imaging 2005: Physiology, Function, and Structure from Medical Images*, pp 804-809.
- [COT03] Coto, Ernesto. (2003). *Métodos de segmentación de imágenes médicas*. Caracas, Venezuela: Universidad Central de Venezuela.
- [COT08] Coto, Ernesto. (2008). *Estrategias avanzadas de segmentación y visualización para colonoscopia virtual*. Caracas, Venezuela: Universidad Central de Venezuela.
- [GOK01] Gokturk, S. y C. Tomasi. (2001). *A New 3-D Pattern Recognition Technique With Application to Computer Aided Colonoscopy*. Kauai, Hawaii: En *Proceedings of the IEEE Computer Science Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 93-100.
- [HUA10] Huang, A., Jiang Li, Ronald M. Summers, Nicholas Petrick y Amy K. Hara. (2010). *Improving polyp detection algorithms for CT colonography: Pareto front approach*. En *Pattern Recognition Letters*, vol. 31, num. 11, pp. 1461-1469. Estados Unidos.
- [KAU01] Kaufman A., W. Li, y K. Kreeger. (2001). *Real-time volumen rendering for virtual colonoscopy*. En *Proceedings of Volume Graphics 01*, pp. 363-374.
- [KIT05] Kitasaka, T., Y. Hayashi, T. Kimura, K. Mori, Y. Suenada y J. Toriwaki. (2005). *Detection of colonic polyps from 3D abdominal CT images by surfacetting*. En *CARS 2005: Computer Assisted Radiology and Surgery*, pp 1151-1156. International Congress Series.

- [KON07] Konukoglu, E. y B. Acar. (2007). *HDF: Heat diffusion elds for polyp detection in CT colonography*. En *Signal Process*, vol. 87, num. 10, pp.2407-2416, ISSN 0165-1684.
- [LAK00] Lakare, S., Wan, M. , Sato M. y Kaufman, "3D digital cleansing using segmentation rays," . En *Proceedings of the conference on visualization 00*. Los Alamitos, CA. USA: IEEE Computer Society Press, 2000, pp. 37-44.
- [MOL98] Torsten Möller, Klaus Mueller, Yair Kurzion, Raghu Machiraju, Roni Yagel. (1998). *Design Of Accurate And Smooth Filters For Function And Derivative Reconstruction*. En *IEEE Symposium on Volume Visualization*, pp. 143-151. Estados Unidos.
- [MON92] Olivier Monga, Serge Benayoun. (1992). *Using partial derivatives of 3D images to extract typical surface features*. Francia.
- [NAV08] Navarro, Héctor. (2008). *Detección de pólipos en datos*. Caracas, Venezuela: Universidad Central de Venezuela.
- [PAG02] Page, D.L, Y. Sun, A. F. Koschan, J. Paik, y M. A. Abidi. (2002). *Normal vector voting: crease detection and curvature estimation on large, noisy meshes*. *Graphical Models*, vol 64, num. 3-4, pp. 199–229.
- [POR97] L. M. Portela. (1997) *Identification and Characterization of Vortices in the Turbulent Boundary Layer*. PhD thesis, Stanford University.
- [RAV09] V.F. van Ravesteijn, L. Zhao, C.P. Botha, F.H. Post, F.M. Vos, y L.J. van Vliet. (2009). *Combining Mesh, Volume, And Streamline Representations For Polyp Detection In CT Colonography* . Amsterdam, Holanda.
- [SAD99] A. Sadarjoen y F. H. Post. (May. 1999) . *Geometric methods for vortex extraction*. En *Data Visualization, proceedings of VisSym 99*, pp. 53–62.

- [URL01] Anuario de Mortalidad 2008. Tomado (2011, octubre 25) de http://www.bvs.org.ve/anuario/anuario_2008.pdf.
- [URL02] Equipo Editorial & Contribuyentes . (2008). *Enema de bario (rayos x de bario; series gi inferiores)*. Tomado de <http://healthlibrary.epnet.com/GetContent.aspx?token=70ff5260-81bd-4de1-9998-14fc98aa9133&chunkiid=103853>
- [URL03] Colonoscopia virtual. *Medline plus*. Tomado (2010, Enero 11) de <http://www.nlm.nih.gov/medlineplus/spanish/ency/article/007253.htm>
- [URL04] CUDA. *Wikipedia*. Tomado (2010, Mayo 28) de <http://es.wikipedia.org/wiki/CUDA>
- [URL05] CUBLAS Library. NVIDIA. Tomado (2010, mayo 25) de http://developer.download.nvidia.com/compute/cuda/1_0/CUBLAS_Library_1.0.pdf
- [URL06] CUFFT Library. NVIDIA. Tomado (2010, mayo 25) de http://developer.download.nvidia.com/compute/cuda/1_0/CUFFT_Library_1.0.pdf
- [URL07] CUDA GPUs. NVIDIA. Tomado (2010, mayo 25) de http://www.nvidia.com/object/cuda_gpus.html
- [URL08] GPU Computing. NVIDIA. Tomado (2011, Julio 4) de http://www.nvidia.es/page/gpu_computing.html
- [URL09] Cuda C Programming Guide. NVIDIA. Tomado (2011, Julio 6) de http://developer.download.nvidia.com/compute/cuda/3_2/toolkit/docs/CUDA_C_Programming_Guide.pdf.
- [VAN98] L. J. van Vliet, I. T. Young, y P.W. Verbeek. (Aug. 1998) *Recursive gaussian derivative filters*. En *Proceedings of the 14th International Conference of Pattern Recognition (ICPR'98)*, pp. 509–514.

- [WAN99] Wan, m., Q. Tang, A. Kaufman, Z. Liang y M. Wax. (1999). *Volume rendering based interactive navigation within the human colon*. En *Proceedings of the IEEE Visualization 99*. Washington, DC, USA: IEEE computer Society, pp.397-400.
- [WAN00] Wan, M., W. Li, K. Kreeger, I. Bitter, A. Kaufman, Z. Liang, D. Chen, y M. (Feb 2000) Wax, *3D virtual colonoscopy with real-time volume rendering*. En *Physiology and Function form Multidimensional Images, proceedings of SPIE*, C. Chen y A. Clough, Eds, vol 3978, pp. 165-171.
- [WAN02] M. Wan, Z. Liang, Q. Ke, L. Hong, I. Bitte, y A. Kaufman. (2002) *Automatic centerline extraction for virtual colonoscopy*. En *IEEE Transactions on Medical Imaging*, vol. 21, num. 12, pp. 1450–1460.
- [YOS01] Yoshida, H. y J. Nappi. (2001). *Three-Dimensional Computer-Aided Diagnosis Scheme for Detection of Colonic Polyps*. En *IEEE Transactions on Medical Imaging*, vol. 20, num. 12, pp. 1261-1274.
- [YOU97] You, S., L. hong, M. Wan, K. Junyaprasert, A. Kaufman, S. Muraki, Y. Zhou, M.Wax y Z. Liang. (Oct. 1997). *Interactive volume rendering for virtual colonoscopy*. En *Proceedings of the IEEE Visualization 97*. Los Alamitos, CA. USA: IEEE Computer Society Press, pp. 433-436.
- [ZHA06] Zhao, L., C. Botha, J. Bescos, R. Truyen, F. Vos y F. Post. (2006). *Lines of Curvature for Polyp Detection in Virtual Colonoscopy*. En *IEEE Transactions on Visualization and Computer Graphics*, vol.12, num 5, pp. 885-892.

CAPÍTULO 6: Apéndice

Apéndice 1: Cálculo del filtro de la segunda derivada según [MOL98].

Ecuaciones base:

$$f_r^w(t) = \sum_{n=0}^N a_n^w(\tau) f^{(n)}(t) + r_N^w, i(\tau)$$

con:

$$a_n^w(\tau) = \frac{T^n}{n!} \sum_{k=-\infty}^{\infty} (k - \tau)^n w(\tau - k)$$

Condición 1:

$$a_0(\tau) = 0$$

$$a_1(\tau) = 0$$

$$a_2(\tau) = 1$$

*Para el filtro de la
segunda derivada*

$$a_n^w = 0 \text{ para } n < 2$$

$$\text{y } a_2^w = 1$$

Es decir,

$$w_{-4} + w_{-3} + w_{-2} + w_{-1} + w_0 + w_1 + w_2 + w_3 = 0$$

$$(4-t)w_{-4} + (3-t)w_{-3} + (2-t)w_{-2} + (1-t)w_{-1} + (-t)w_0 + (-1-t)w_1 + (-2-t)w_2 + (-3-t)w_3 = 0$$

$$(4-t)^2 w_{-4} + (3-t)^2 w_{-3} + (2-t)^2 w_{-2} + (1-t)^2 w_{-1} + (-t)^2 w_0 + (-1-t)^2 w_1 + (-2-t)^2 w_2 + (-3-t)^2 w_3 = 0$$

Condición 2:

$$a_3(\tau) = 0$$

$$a_4(\tau) = 0$$

*Para un filtro con
error de orden 3*

$$a_n^w = 0 \text{ para}$$

$$2 < n \leq 2+3-1=4$$

Es decir,

$$(4-t)^3 w_{-4} + (3-t)^3 w_{-3} + (2-t)^3 w_{-2} + (1-t)^3 w_{-1} + (-t)^3 w_0 + (-1-t)^3 w_1 + (-2-t)^3 w_2 + (-3-t)^3 w_3 = 0$$

$$(4-t)^4 w_{-4} + (3-t)^4 w_{-3} + (2-t)^4 w_{-2} + (1-t)^4 w_{-1} + (-t)^4 w_0 + (-1-t)^4 w_1 + (-2-t)^4 w_2 + (-3-t)^4 w_3 = 0$$

Resolviendo el sistema de ecuaciones de las condiciones 1 y 2 obtenemos lo siguiente:

$$\begin{aligned}
 w_{-4} &= -\frac{3}{2}t + \frac{1}{2}t^2 - w_1 - 5w_2 - 15w_3 + \frac{11}{2} \\
 w_{-3} &= -\frac{14}{3} + 70w_3 + 7t - 2t^2 + 5w_1 + 24w_2 \\
 w_{-2} &= \frac{19}{2} - 126w_3 - 12t + 3t^2 - 10w_1 - 45w_2 \\
 w_{-1} &= 105w_3 - \frac{26}{3} + 9t - 2t^2 + 10w_1 + 40w_2 \\
 w_0 &= \frac{35}{12} - \frac{5}{2}t + \frac{1}{2}t^2 - 35w_3 - 5w_1 - 15w_2 \\
 w_1 &= w_1 \\
 w_2 &= w_2 \\
 w_3 &= w_3
 \end{aligned} \tag{1}$$

Condición 3:

$$w \in \mathcal{C}^2$$

$$w_k(\tau) \in \mathcal{C}^2 \text{ y } w_k^{(m)}(1) = w_{k+1}^{(m)}(0) \forall k, m < 2$$

$$\begin{array}{lll}
 w_{-4}(0) = 0 & w'_{-4}(0) = 0 & w''_{-4}(0) = 0 \\
 w_{-4}(1) = w_{-3}(0) & w'_{-4}(1) = w'_{-3}(0) & w''_{-4}(1) = w''_{-3}(0) \\
 w_{-3}(1) = w_{-2}(0) & w'_{-3}(1) = w'_{-2}(0) & w''_{-3}(1) = w''_{-2}(0) \\
 w_{-2}(1) = w_{-1}(0) & w'_{-2}(1) = w'_{-1}(0) & w''_{-2}(1) = w''_{-1}(0) \\
 w_{-1}(1) = w_0(0) & w'_{-1}(1) = w'_0(0) & w''_{-1}(1) = w''_0(0) \\
 w_0(1) = w_1(0) & w'_0(1) = w'_1(0) & w''_0(1) = w''_1(0) \\
 w_1(1) = w_2(0) & w'_1(1) = w'_2(0) & w''_1(1) = w''_2(0) \\
 w_2(1) = w_3(0) & w'_2(1) = w'_3(0) & w''_2(1) = w''_3(0) \\
 w_3(1) = 0 & w'_3(1) = 0 & w''_3(1) = 0
 \end{array} \tag{2}$$

Escogemos un polinomio de tercer grado debido a que uno de grado menor no nos arroja una solución sino infinitas.

$$w_k(t) = D_k t^3 + C_k t^2 + B_k t + A_k$$

Usando la notación $w_k = w_k(t)$, sustituimos la ecuación anterior en (1) y (2) y tenemos:

$$\begin{array}{lll}
 A_{-4} = 0 & B_{-4} = 0 & C_{-4} = 0 \\
 A_{-4} + B_{-4} + C_{-4} + D_{-4} = A_{-3} & B_{-4} + 2C_{-4} + 3D_{-4} = B_{-3} & C_{-4} + 3D_{-4} = C_{-3} \\
 A_{-3} + B_{-3} + C_{-3} + D_{-3} = A_{-2} & B_{-3} + 2C_{-3} + 3D_{-3} = B_{-2} & C_{-3} + 3D_{-3} = C_{-2} \\
 A_{-2} + B_{-2} + C_{-2} + D_{-2} = A_{-1} & B_{-2} + 2C_{-2} + 3D_{-2} = B_{-1} & C_{-2} + 3D_{-2} = C_{-1} \\
 A_{-1} + B_{-1} + C_{-1} + D_{-1} = A_0 & B_{-1} + 2C_{-1} + 3D_{-1} = B_0 & C_{-1} + 3D_{-1} = C_0 \\
 A_0 + B_0 + C_0 + D_0 = A_1 & B_0 + 2C_0 + 3D_0 = B_1 & C_0 + 3D_0 = C_1 \\
 A_1 + B_1 + C_1 + D_1 = A_2 & B_1 + 2C_1 + 3D_1 = B_2 & C_1 + 3D_1 = C_2 \\
 A_2 + B_2 + C_2 + D_2 = A_3 & B_2 + 2C_2 + 2D_2 = B_3 & C_2 + 3D_2 = C_3
 \end{array} \quad (3)$$

$$A_{-4} + tB_{-4} + t^2C_{-4} + t^3D_{-4} =$$

$$-\frac{3}{2}t + \frac{1}{2}t^2 - A_1 - tB_1 - t^2C_1 - t^3D_1 - 5A_2 - 5B_2t - 5t^2C_2 - 5t^3D_2 - 15A_3 - 15B_3t - 15t^2C_3 - 15t^3D_3 + \frac{11}{2}$$

$$A_{-3} + tB_{-3} + t^2C_{-3} + t^3D_{-3} =$$

$$-\frac{14}{30} + 70A_3 + 70B_3t + 70t^2C_3 + 70t^3D_3 + 7t - 2t^2 + 5A_1 + 5tB_1 + 5t^2C_1 + 5t^3D_1 + 24A_2 + 24B_2t + 24t^2C_2 + 24t^3D_2$$

$$A_{-2} + tB_{-2} + t^2C_{-2} + t^3D_{-2} =$$

$$\frac{19}{2} - 126A_3 - 126B_3t - 126t^2C_3 - 126t^3D_3 - 12t + 3t^2 - 10A_1 - 10tB_1 - 10t^2C_1 - 10t^3D_1 - 45A_2 - 45B_2t - 45t^2C_2 - 45t^3D_2$$

$$A_{-1} + tB_{-1} + t^2C_{-1} + t^3D_{-1} =$$

$$105A_3 + 105tB_3 + 105t^2C_3 + 105t^3D_3 - \frac{26}{3} + 9t - 2t^2 + 10A_1 + 10tB_1 + 10t^2C_1 + 10t^3D_1 + 40A_2 + 40tB_2 + 40t^2C_2 + 40t^3D_2$$

$$A_0 + tB_0 + t^2C_0 + t^3D_0 =$$

$$\frac{35}{12} - \frac{5}{2}t + \frac{1}{2}t^2 - 35A_3 - 35tB_3 - 35t^2C_3 - 35t^3D_3 - 5A_1 - 5tB_1 - 5t^2C_1 - 5t^3D_1 - 15A_2 - 15tB_2 - 15t^2C_2 - 15t^3D_2$$

Como el filtro corresponde a la derivada segunda debe ser simétrico. La condición de simetría se expresa de la siguiente forma:

$$\mathbf{w}_{k-1}(\tau) = \mathbf{w}_{-k}(\mathbf{1} - \tau)$$

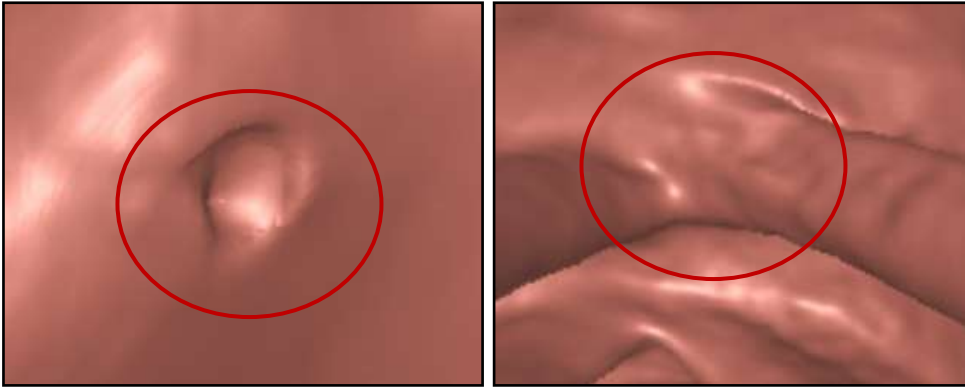
De lo cual se deducen las siguientes ecuaciones:

$$\begin{aligned}
 A_{-4} &= A_3 + B_3 + C_3 + D_3 & A_{-3} &= A_2 + B_2 + C_2 + D_2 \\
 B_{-4} &= -B_3 - 2C_3 - 3D_3 & B_{-3} &= -B_2 - 2C_2 - 3D_2 \\
 C_{-4} &= C_3 + 3D_3 & C_{-3} &= C_2 + 3D_2 \\
 D_{-4} &= D_3 & D_{-3} &= D_2 \\
 \\
 A_{-2} &= A_1 + B_1 + C_1 + D_1 & A_{-1} &= A_0 + B_0 + C_0 + D_0 \\
 B_{-2} &= -B_1 - 2C_1 - 3D_1 & B_{-1} &= -B_0 - 2C_0 - 3D_0 \\
 C_{-2} &= C_1 + 3D_1 & C_{-1} &= C_0 + 3D_0 \\
 D_{-2} &= D_1 & D_{-1} &= D_0
 \end{aligned} \tag{4}$$

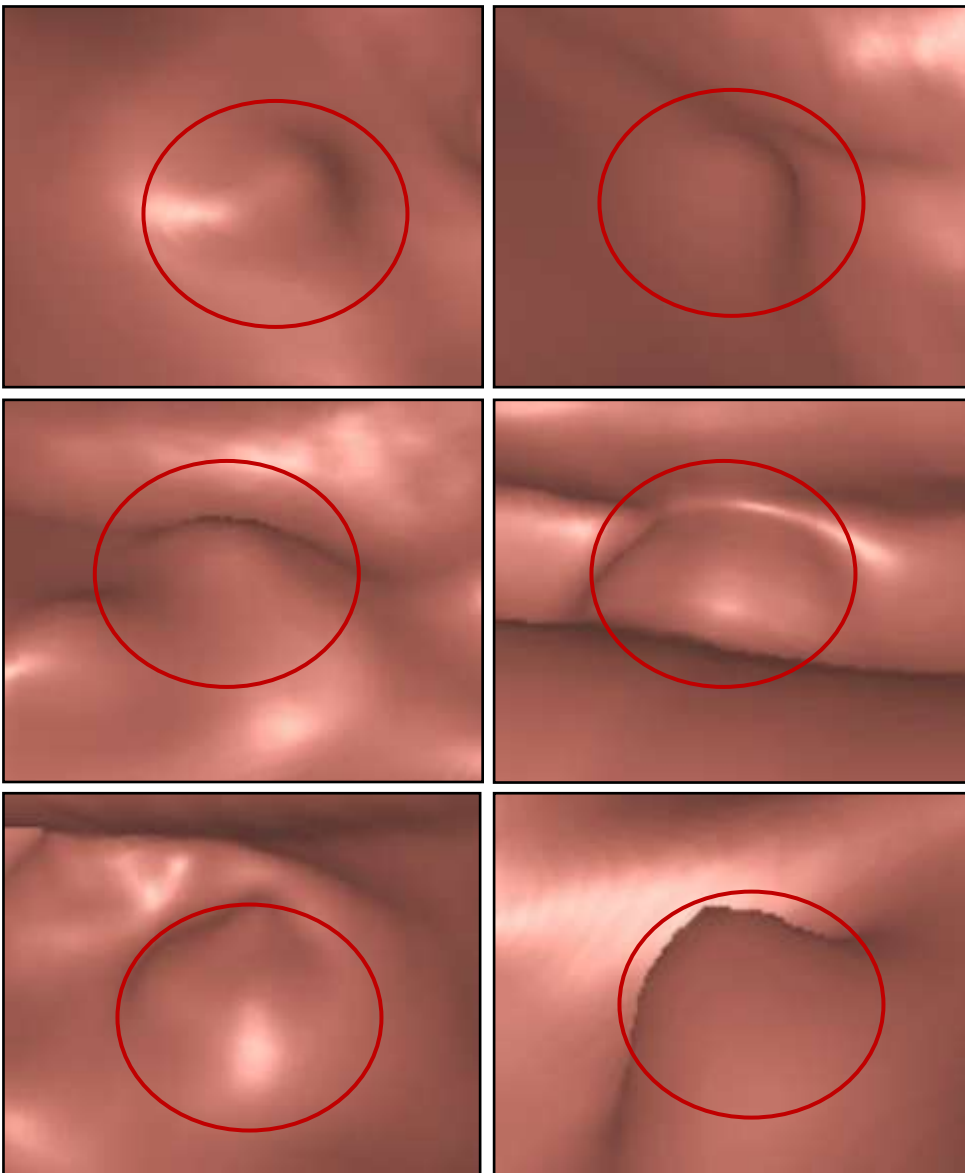
Resolviendo el sistema de ecuaciones resultante de (3) y (4) finalmente obtenemos:

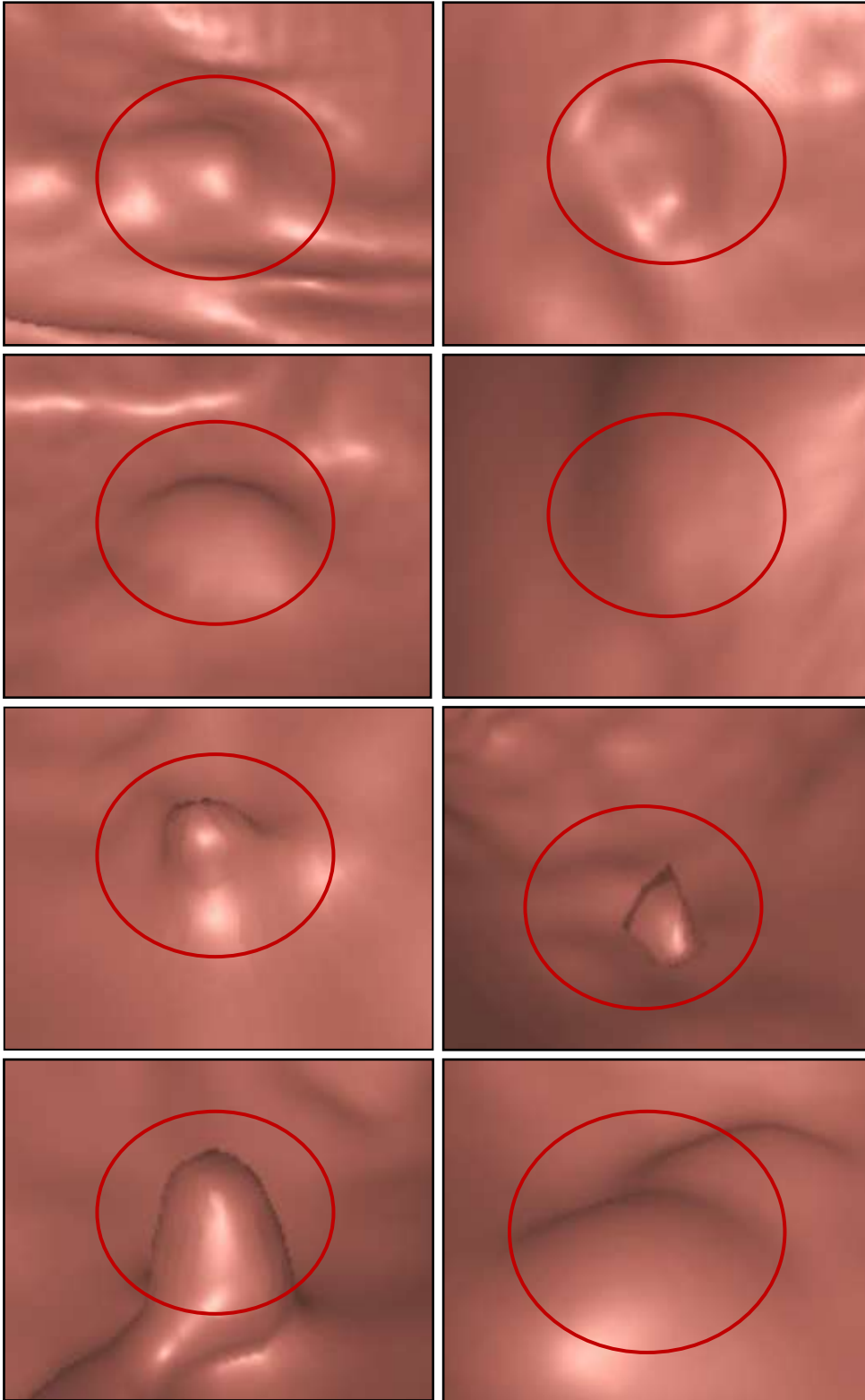
$$\begin{array}{llll}
 A_{-4} = 0 & B_{-4} = 0 & C_{-4} = 0 & D_{-4} = -\frac{1}{24} \\
 A_{-3} = -\frac{1}{24} & B_{-3} = -\frac{1}{8} & C_{-3} = -\frac{1}{8} & D_{-3} = \frac{11}{24} \\
 A_{-2} = \frac{1}{6} & B_{-2} = 1 & C_{-2} = \frac{5}{4} & D_{-2} = -\frac{41}{24} \\
 A_{-1} = \frac{17}{24} & B_{-1} = -\frac{13}{8} & C_{-1} = -\frac{31}{8} & D_{-1} = \frac{25}{8} \\
 A_0 = -\frac{5}{3} & B_0 = 0 & C_0 = \frac{11}{2} & D_0 = -\frac{25}{8} \\
 A_1 = \frac{17}{24} & B_1 = \frac{13}{8} & C_1 = -\frac{31}{8} & D_1 = \frac{41}{24} \\
 A_2 = \frac{1}{6} & B_2 = -1 & C_2 = \frac{5}{4} & D_2 = -\frac{11}{24} \\
 A_3 = -\frac{1}{24} & B_3 = \frac{1}{8} & C_3 = -\frac{1}{8} & D_3 = \frac{1}{24}
 \end{array}$$

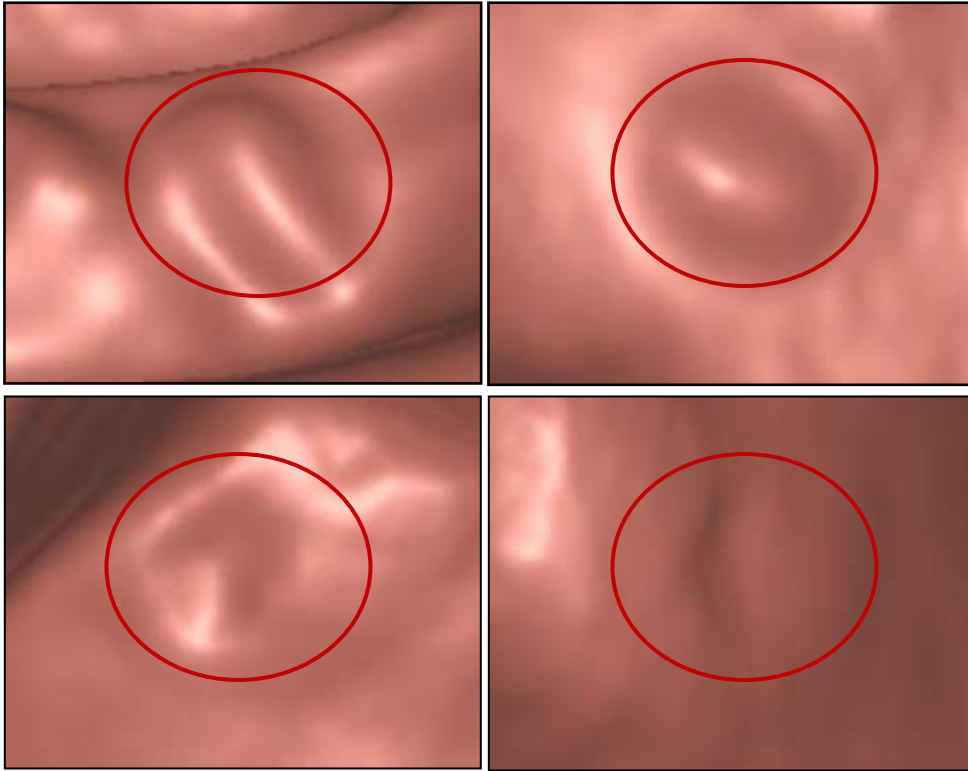
Apéndice 2: Imágenes de candidatos detectados en el volumen 2.



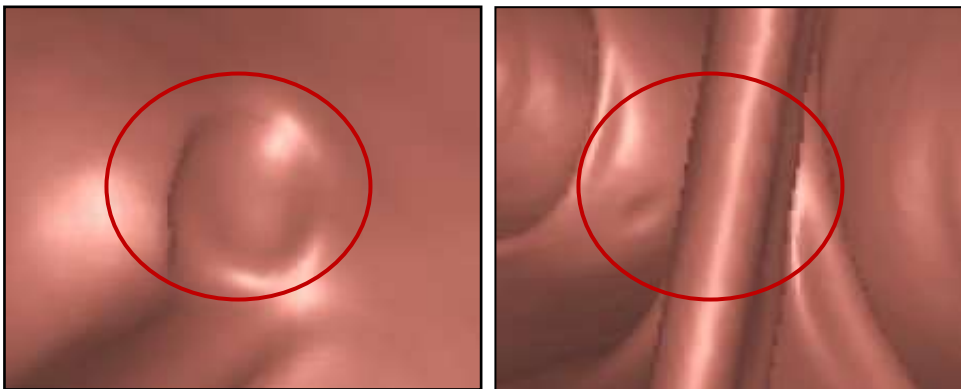
Apéndice 3: Imágenes de candidatos detectados en el volumen 3.



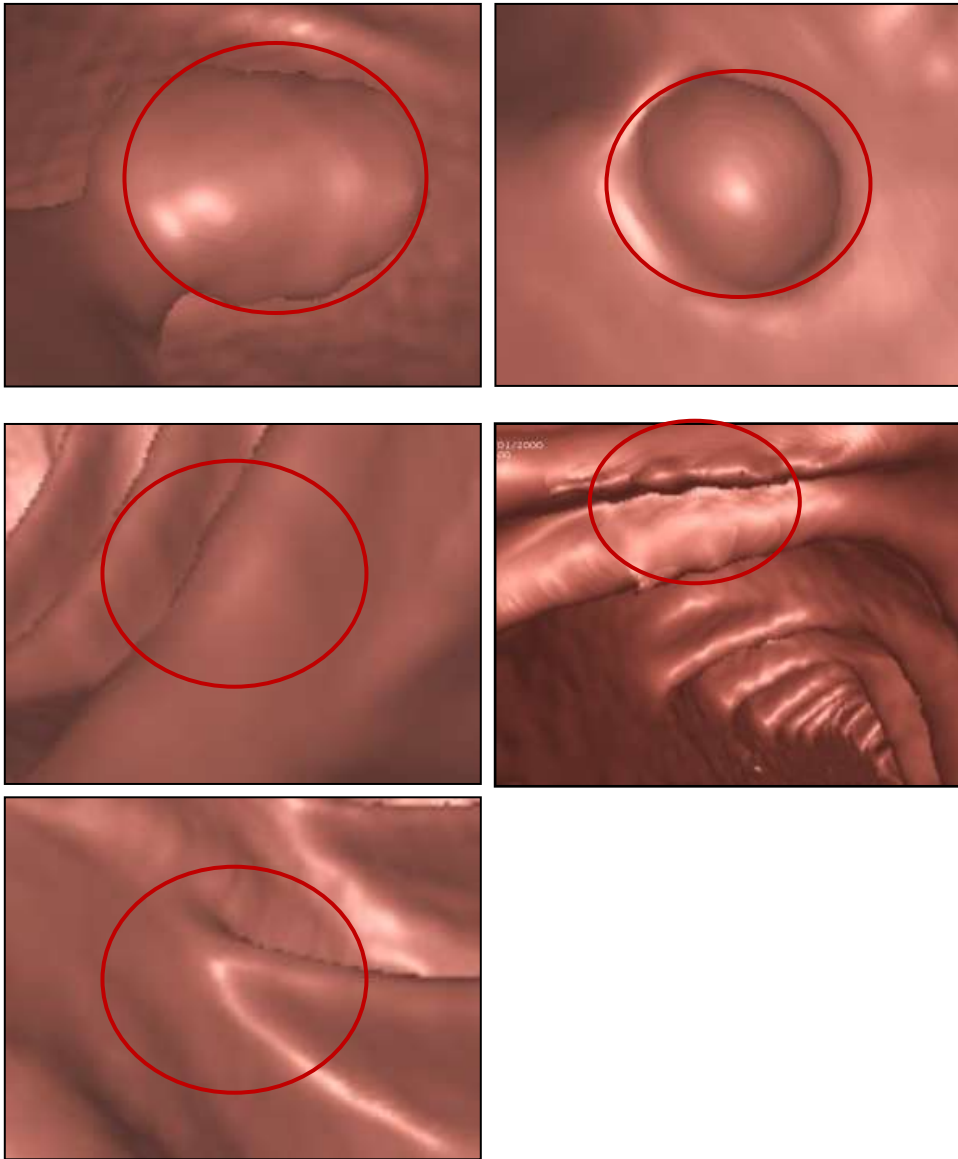




Apéndice 4: Imágenes de candidatos detectados en el volumen 4.

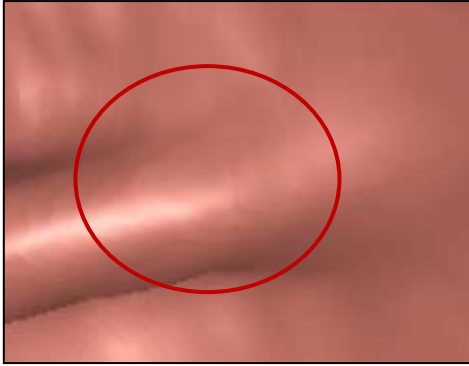


Apéndice 5: Imágenes de candidatos detectados en el volumen 5.

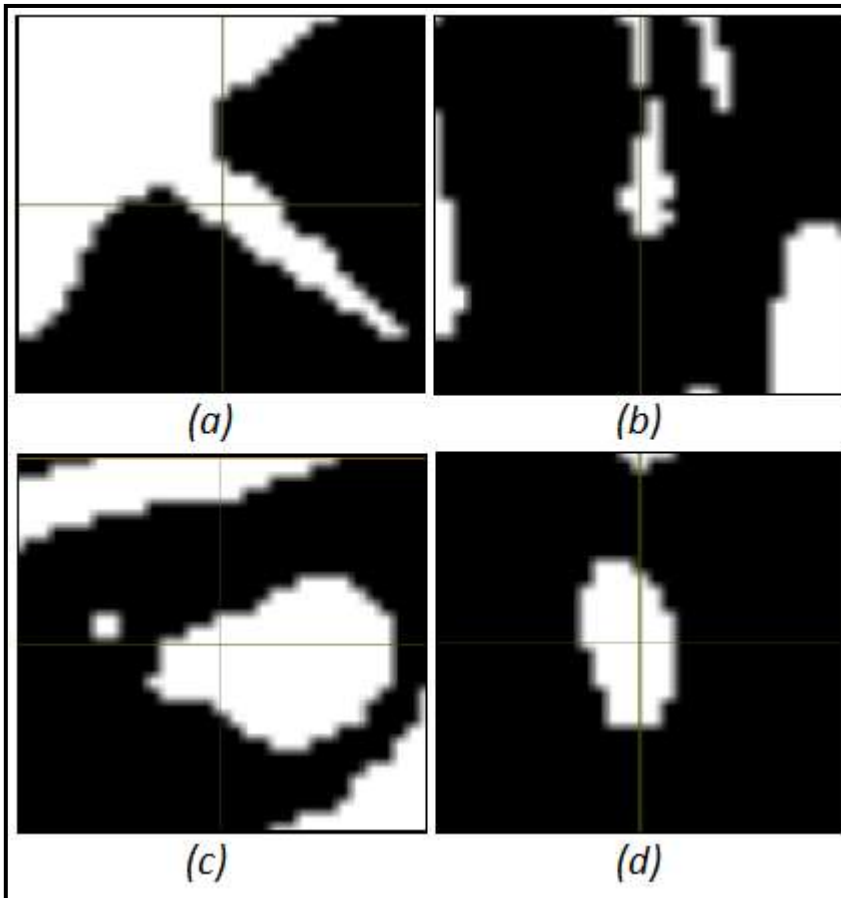


Apéndice 6: Imágenes de candidatos detectados en el volumen 6.





Apéndice 7: Imágenes de cortes planares.



- (a) Posible pliegue, llega al borde por lo que no es candidato a pólipo.
- (b) Forma alargada, a pesar de que no toca los bordes es descartado.
- (c) Fuerte candidato a pólipo.
- (d) Fuerte candidato a pólipo.