



Universidad de las Ciencias Informáticas
Centro de Tecnologías Interactivas, Facultad 4

Componente de filtrado de imágenes en el espacio de frecuencias

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autor: Danny Israel Guerra Enrique

Tutor: MSc. Rubén Alcolea Núñez

La Habana, 2020

El análisis matemático es tan extenso como la naturaleza misma; define todas las relaciones sensibles, mide el tiempo, los espacios, las fuerzas, las temperaturas: su atributo principal es la claridad; no tiene en absoluto signos para expresar nociones confusas. Relaciona los fenómenos más diversos y descubre las analogías secretas que los une.

Jean Baptiste Joseph Fourier (1768-1830)

Dedicatoria

A mi familia y amigos.

Agradecimientos

Agradezco al profesor Rubén y a todos aquellos que me acompañaron durante el proceso de desarrollo de esta tesis. A mi familia por apoyarme en todo momento y por último y en primer lugar a Dios.

Declaración de autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año_____.

Danny Israel Guerra Enrique
Autor

MSc. Rubén Alcolea Núñez
Tutor

Resumen

El filtrado de imágenes es una de las etapas fundamentales de la visualización de volumen. Este permite reducir el ruido presente en las imágenes y mejorar la calidad de modo que posibilite resaltar áreas o estructuras que no siempre apreciadas en las imágenes originales. Sin embargo, la aplicación de filtros en el dominio del espacio es un proceso costoso que depende del tamaño del volumen de datos procesado. Como consecuencia, para volúmenes de mediano y gran tamaño, la etapa de filtrado puede tener un impacto computacional considerable para la visualización. El presente trabajo propone la utilización de filtros en el dominio de las frecuencias empleando la Transformada de Fourier con el objetivo de agilizar la etapa de filtrado de imágenes digitales. La propuesta incluye filtros de paso alto para atenuar las bajas frecuencias y filtros de paso bajo para atenuar las altas frecuencias. Esta propuesta puede extenderse en el futuro a datos volumétricos.

Palabras clave: filtro, frecuencias, imágenes digitales, Transformada de Fourier.

Abstract

Image filtering is one of the most important stages of volume rendering. It allows to reduce noise present at the images and to improve the quality in order to highlight areas or structures that are not always measurable in the original images. However, filtering application at the space domain is an expensive process that depends of the volume size to be processed. As a consequence, for medium and high-volume sizes, filtering stage can have a remarkable computational cost for visualization performance. The present work proposes the use of filters at frequencies domain using the Fourier Transform in order to speed the filtering stage of digital images. The proposal includes high-pass filters to attenuate low frequencies and low-pass filters to attenuate high frequencies. This solution can be extended in the future to volumetric data.

Keywords: digital images, filter, Fourier Transform, frequencies.

Índice general

Introducción	2
Fundamentación Teórica.....	6
1.1. Transformada de Fourier y su Inversa.	6
1.2. Filtrado en el dominio de las frecuencias.....	8
1.3. Sistema Vismedic – Illustration.	9
1.4. Etapa de filtrado del sistema Vismedic – Illustration.	11
1.5. Herramientas y metodología de desarrollo.	15
Consideraciones parciales del capítulo	16
Características y diseño del sistema.....	17
2.1 Requisitos funcionales.	17
2.2 Requisitos no funcionales.	18
2.3 Historias de Usuarios.	19
2.4 Estimación de esfuerzo por Historias de Usuario.	20
2.5 Desarrollo del plan de iteraciones.	21
2.6 Tarjetas CRC.....	22
2.7 Tareas de ingeniería para la Iteración I.	22
2.8 Modelo de filtrado en el dominio de las frecuencias.	22
2.9 Filtros propuestos como solución.	24
2.9.1 Filtros de paso alto.	24
2.9.2 Filtros de paso bajo.	26
2.9.3 Filtros para la detección de bordes.....	27
2.10 Diagrama de clases.	29
2.11 Integración con el sistema Vismedic-Illustration.....	30
Consideraciones parciales del capítulo	30
Conclusiones	31
Recomendaciones	32
Bibliografía	33
Anexos	34

Índice de figuras

Figura. 1 Sistema Vismedic - Illustration para la generación de ilustraciones.	10
Figura. 2 Arquitectura del sistema Vismedic – Illustration.	10
Figura. 3 Filtro del promedio. (izquierda) Valores de intensidad de una imagen. (centro) Máscara definida para el promedio. (derecha) Valor resultante del filtro del promedio para cuatro píxeles de la imagen representada.	11
Figura. 4 Filtro de la mediana.	12
Figura. 5 Máscaras de convolución del filtro de Sobel.	13
Figura. 6 Ejemplo de aplicar el filtro Laplaciano. Imagen cortesía de [3].	13
Figura. 7 Modelo de filtrado en el dominio de las frecuencias.	23
Figura. 8 Ejemplos de tres imágenes con sus respectivas transformadas.	23
Figura. 9 Ejemplo de aplicación del filtro de la media.	25
Figura. 10 Aplicación del filtro media sobre una matriz numérica.	25
Figura. 11 Filtro Gaussiano.	26
Figura. 12 Filtro de sustracción de la media.	27
Figura. 13 Ejemplos de máscaras de Sobel: (izquierda) Máscara para la dirección "x". (Derecha) Máscara para la dirección "y"	27
Figura. 14 Ejemplo de aplicación del filtro Sobel.	28
Figura. 15 Diagrama de clases.	¡Error! Marcador no definido.
Figura. 16 Arquitectura del sistema Vismedic - Illustration.	¡Error! Marcador no definido.

Índice de tablas

Tabla 1 Requisitos funcionales.	17
Tabla 2 Requisitos no funcionales.	18
Tabla 3 Estimación de esfuerzo por Historia de Usuario.....	20
Tabla 4 Plan de duración de las iteraciones.	21
Tabla 5 Plan de entregas.	21
Tabla 6 Tarjeta CRC #1.	22

Introducción

El desarrollo actual de las tecnologías de adquisición y procesamiento de imágenes médicas ha contribuido a incrementar el desarrollo de aplicaciones para el tratamiento de imágenes digitales. Como resultado, el uso de software y herramientas informáticas para apoyar los procesos de las empresas, incrementa la eficiencia y disminuye los tiempos de desarrollo de tareas que anteriormente se hacían de forma manual o consumían un tiempo considerable. Estas herramientas permiten extraer información importante presente en las imágenes y visualizar con gran calidad la información relevante presente en el objeto estudiado.

Las imágenes médicas se generan empleando técnicas de adquisición como la Tomografía Computarizada (TC) y la Resonancia Magnética (RM). La serie de imágenes que se obtienen mediante los dispositivos de adquisición permiten generar una reconstrucción tridimensional del objeto analizado y apreciar las estructuras internas con mayor nivel de detalle [1]. Esta aplicación es muy útil en áreas como la medicina, donde se utiliza para asistir a los especialistas en el análisis de imágenes para el diagnóstico correcto de pacientes.

Los dispositivos de adquisición de imágenes como los resonadores magnéticos o los tomógrafos generan un gran volumen de información. Para poder utilizar de forma eficiente esta información, se utilizan diferentes técnicas avanzadas de procesamiento, análisis y visualización de imágenes. Una de las etapas fundamentales del procesamiento de imágenes es el preprocesamiento. Esta etapa es muy importante, pues en ella se realizan operaciones como reducir el ruido presente en las imágenes o extraer características de interés como los bordes de las estructuras presentes en las imágenes.

Las imágenes médicas correspondientes a un estudio se representan como un conjunto de imágenes individuales. Cada imagen representa un corte del objeto escaneado y se compone de píxeles. Al combinar las imágenes individuales en una red 3D se obtiene un volumen de datos [2]. Cada elemento individual de esta red se denomina vóxel y representa la unidad mínima de información de un volumen de datos. En adición, a las dimensiones horizontal "x" y vertical "y", se incrementa una dimensión que representa la profundidad "z". Como resultado, en la práctica, se trabaja con el volumen de datos obtenido a partir de las imágenes debido a que reduce la imprecisión de los resultados del trabajo con las imágenes individuales.

Para visualizar un volumen de datos, se utilizan algoritmos de visualización. La visualización de volumen tiene cinco etapas fundamentales: el muestreo, el filtrado, la clasificación, el sombreado y la composición [5].

En la etapa de muestreo se accede a los datos del volumen que se desea visualizar. En la etapa de filtrado, se reduce el ruido, se mejora la calidad del volumen y se resaltan características de interés como los bordes de las estructuras. En la etapa de clasificación, se realiza el mapeo de los valores del volumen a propiedades ópticas mediante el empleo de una o varias funciones de transferencia [2], lo que posibilita visualizar el volumen con un mayor nivel de realismo. La etapa de sombreado es opcional y solo se aplica si se tiene en cuenta la contribución de fuentes de luces externas para incorporar iluminación a la visualización. Finalmente, la etapa de composición calcula el color final para cada píxel en el *framebuffer* que posteriormente se visualiza en la pantalla. La presente investigación se centra en la etapa de filtrado.

Una de las etapas que mayor influencia tiene en la calidad de la visualización es el filtrado. En esta etapa se realizan operaciones para reducir el ruido presente en las imágenes y para mejorar la calidad de la imagen. Los filtros se clasifican de acuerdo al dominio en que se aplican a la imagen. Estos pueden ser en el dominio del espacio o en el dominio de las frecuencias [3]. Los filtros del dominio del espacio modifican directamente el valor de los píxeles de la imagen por el nuevo valor que se obtiene al aplicar el filtro. El nuevo valor se obtiene al aplicar una convolución en la vecindad definida para el píxel en cuestión. Por otra parte, los filtros del dominio de las frecuencias operan empleando la Transformada de Fourier de la imagen. Para ello se aplica la Transformada de Fourier a la imagen original, luego se multiplica por la función del filtro seleccionado y finalmente se realiza la transformación nuevamente al dominio del espacio empleando la Transformada Inversa de Fourier [3]. Una de las razones fundamentales para utilizar los filtros en el dominio de las frecuencias es generalmente debido a la eficiencia computacional. En la práctica es más eficiente realizar dos transformaciones 2D al espacio de Fourier y una multiplicación por el filtro que realizar una convolución en el dominio del espacio de una imagen [4]. Esto se evidencia en mayor medida al incrementar el tamaño del filtro.

El Centro de Tecnologías Interactivas (Vertex) en la Universidad de las Ciencias Informáticas, tiene dentro de sus líneas de investigación la visualización ilustrativa de volúmenes. Esta tiene como objetivo la generación de ilustraciones interactivas generadas a partir de volúmenes de datos. Uno de los resultados obtenidos en investigaciones anteriores es la aplicación *Vismedic – Illustration*, la cual posibilita generar ilustraciones interactivas a partir de datos volumétricos. Las ilustraciones pueden utilizarse como medios de enseñanza en áreas como la medicina o la química para representar de forma ilustrativa conceptos que se visualizan con un mayor nivel de detalle mediante estilos ilustrativos.

La etapa de filtrado actual de la aplicación Vismedic – Illustration incluye filtros de reducción de ruido como la mediana y el promedio, filtros para mejorar la calidad de la imagen como la expansión de contraste y filtros para calcular el gradiente y detectar bordes como el filtro de Sobel y el Laplaciano. La aplicación de estos filtros posibilita visualizar con un alto nivel de realismo los datos procesados. Como resultado, es posible resaltar áreas o estructuras que no siempre es posible apreciar en las imágenes originales [12].

Sin embargo, la etapa de filtrado actual de Vismedic – Illustration presenta algunas limitaciones. Los filtros actuales están implementados en el dominio del espacio. Esto implica que al aplicar un filtro se calcula el valor de cada vóxel del volumen mediante una función que analiza los vóxeles vecinos. En consecuencia, para volúmenes de mediano y gran tamaño (256^3 vóxeles), la aplicación de estos filtros tiene un costo computacional considerable, que influye en el rendimiento de la visualización. Por otra parte, la aplicación de filtros de reducción de ruido como el promedio y la mediana, en ocasiones eliminan detalles importantes del volumen como los bordes o las estructuras pequeñas, que pueden ser de interés para la visualización, provocando pérdida de información relevante para las ilustraciones generadas.

Teniendo en cuenta la situación problemática anterior, se define como **problema de investigación**: ¿Cómo incorporar filtros en el dominio de las frecuencias a la etapa de filtrado de volúmenes de datos? Se define como **objeto de estudio** el filtrado de imágenes digitales. El **objetivo general** del presente trabajo es diseñar una propuesta de filtros para reducir el costo computacional de la etapa de filtrado de volúmenes de datos, acotándose como campo de acción el filtrado de imágenes digitales en el dominio de las frecuencias.

Para dar cumplimiento al objetivo general se definieron las siguientes **tareas de investigación**:

- Elaboración del marco teórico a partir del estado del arte referente al filtrado de imágenes en el dominio de las frecuencias.
- Selección de las técnicas de filtrado para reducir el ruido y mejorar la calidad de una imagen.
- Selección de los algoritmos más eficaces para el filtrado de imágenes en el dominio de las frecuencias.
- Diseño de la solución propuesta.

En el desarrollo de la presente investigación, se utilizaron los siguientes métodos de la investigación científica:

Métodos teóricos:

Histórico-Lógico: Para analizar el origen, la evolución y las tendencias actuales del filtrado de imágenes o volúmenes de datos.

Analítico-Sintético: Para extraer y analizar la información sobre los algoritmos de filtrado de imágenes o volúmenes de datos.

Métodos empíricos:

Consulta de fuentes de información: Para consultar toda fuente de información para la actualización del conocimiento sobre filtrado de imágenes.

Observación: Para apreciar el impacto de los filtros propuestos en el rendimiento de la aplicación y en la calidad de la visualización.

Experimento: Para identificar las combinaciones más efectivas de parámetros para los filtros propuestos que ofrecen los mejores resultados para la etapa de filtrado.

El documento está estructurado en dos capítulos, que abordan los contenidos que se describen a continuación:

Capítulo 1: Fundamentación Teórica. En este capítulo se definen los términos y conceptos relacionados con el filtrado de imágenes en el dominio de las frecuencias. Además, se explican las herramientas que se utilizarán, así como la metodología que guiará el proceso de desarrollo.

Capítulo 2: Características y diseño del sistema. En este capítulo se definen los requisitos funcionales y no funcionales de la solución propuesta y las Historias de Usuario. Además, se explican los elementos esenciales de la solución propuesta.

CAPÍTULO 1

Fundamentación Teórica

Una de las etapas fundamentales del procesamiento de imágenes digitales es el filtrado. En esta etapa, se aplican filtros para reducir el ruido presente en las imágenes y para mejorar la calidad de la imagen. Estos filtros se pueden clasificar de acuerdo al dominio donde se aplican en la imagen.

Los filtros se pueden aplicar en el dominio del espacio o en el dominio de las frecuencias. Los que se aplican en el dominio del espacio, implican la transformación directa de los píxeles de la imagen por el nuevo valor que se obtiene al aplicar el filtro. El nuevo valor del píxel se calcula a partir de aplicar una convolución en la vecindad definida para el píxel en cuestión.

Por otra parte, los filtros que se aplican en el dominio de las frecuencias procesan la imagen trabajando sobre el dominio de las frecuencias en la Transformada de Fourier de la imagen. Para ello se aplica la Transformada de Fourier a la imagen original, luego se multiplica por la función del filtro seleccionado y finalmente se realiza la transformación nuevamente al dominio del espacio empleando la Transformada Inversa de Fourier [3].

Los filtros del dominio de las frecuencias posibilitan la asociación del concepto de frecuencia con algunas características de la imagen como los cambios de tonalidad. Además, proporcionan flexibilidad en el diseño de filtros y mayor rapidez al utilizar el Teorema de Convolución [4].

El filtrado de imágenes digitales puede abordarse de manera más intuitiva cuando se realiza en el espacio de las frecuencias, una vez aplicada la Transformada de Fourier a la imagen. La generación de filtros paso bajo, paso alto y paso banda es directa, aunque su mayor potencial reside en la creación de filtros para la eliminación del ruido periódico [2].

1.1. Transformada de Fourier y su Inversa.

La Transformada de Fourier es una herramienta muy poderosa que tiene una gran variedad de aplicaciones, en diversas ramas de la matemática y la física matemática, desde la teoría de números y geometría hasta mecánica cuántica, así como en otras áreas de la ciencia e ingeniería tales como procesamiento de señales (electrónica), comunicaciones, óptica, procesamiento de imágenes y medicina [8].

Históricamente, las series de Fourier aparecen de forma natural en Astronomía, de hecho, Neugebauer (1952) descubrió que los Babilonios utilizaron una forma primitiva de ellas en la predicción de ciertos eventos celestiales.

En 1747, la historia de las series de Fourier inicia de manera formal, con D'Alembert y su tratado de las oscilaciones de las cuerdas del violín [10]. Años más tarde, fue Fourier quien perfeccionó las ideas de Bernoulli y las aplicó en 1807, en el estudio del problema de la conducción de calor. Esto quedó plasmado por escrito en el libro clásico "Theorie analytique de la Chaleur" publicado en 1822 [10]. Los razonamientos realizados por Fourier en dicho libro, plantearon de manera inmediata numerosas controversias y cuestiones que han tenido una influencia significativa en la historia de las Matemáticas [8].

La Transformada de Fourier y su inversa tienen gran importancia en el procesamiento de imágenes digitales. Este se aplica en numerosas áreas de la sociedad como las telecomunicaciones, la medicina, la química, la astronomía y la industria. Las aplicaciones en el procesamiento de imágenes digitales se basan en extraer información contenida en la imagen. La Transformada de Fourier permite obtener la representación de información en el espacio de frecuencias y aplicando un operador en este dominio, se puede operar sobre la imagen para detectar y realzar bordes y eliminar ruido.

Una imagen se puede representar como la suma de dos componentes con diferentes escalas espaciales: la versión de la imagen con las bajas frecuencias y la de la imagen con las altas frecuencias [3]. La Transformada de Fourier supone la extensión de esta idea a muchas escalas. Una señal unidimensional se puede descomponer en un conjunto infinito de señales de tipo seno con diferentes frecuencias, amplitudes y fases [3]. El primer componente representa la amplitud media de la señal y tiene frecuencia cero. El siguiente componente, o fundamental, presenta la misma frecuencia que la señal original. Los demás componentes van adquiriendo frecuencias sucesivamente superiores y se denominan armónicos.

A medida que se van añadiendo los sucesivos componentes, la suma se va aproximando cada vez más a la señal original, aunque se necesita un número infinito de aquellos para obtenerla de forma exacta. A la suma total de componentes se le denomina serie de Fourier de la función original. Una imagen digital puede entenderse como una señal discreta bidimensional, de forma que su serie de Fourier es una suma finita de senos y cosenos.

Para obtener la imagen original será preciso sumar todos los términos de la serie. La suma parcial de los primeros componentes, con frecuencias más bajas, producirá una versión de las bajas frecuencias de la imagen, similar al resultado de aplicar un filtro o máscara de paso bajo a la imagen original. Análogamente, la suma de los últimos componentes de la serie, con frecuencias superiores, producirá una versión de la imagen de altas frecuencias, similar a la obtenida al aplicar un filtro paso alto. La frecuencia espacial de una imagen representa la distribución espacial de las intensidades o niveles de gris y sigue una ley sinusoidal, caracterizada por su amplitud, fase, frecuencia y orientación [3].

1.2. Filtrado en el dominio de las frecuencias

La importancia práctica de la Transformada de Fourier radica en que cualquier convolución de una imagen puede obtenerse como el producto de dos imágenes en el dominio de la frecuencia, lo que aporta como ventajas:

- El diseño de un filtro en el dominio de Fourier es directo, en función de la respuesta en frecuencias que se espera conseguir.
- A partir de la respuesta en frecuencias se puede calcular la máscara de convolución equivalente en el dominio espacial mediante la transformada inversa de la matriz de filtrado [4].
- Mientras que la convolución en el dominio espacial tan solo es viable para máscaras pequeñas, debido al notable incremento en el tiempo de procesamiento, la convolución en el dominio de las frecuencias es independiente de este parámetro, siendo así ideal cuando se deseen aplicar convoluciones de gran tamaño.

Una convolución puede aplicarse transformando ambos, imagen y filtro, al dominio de las frecuencias, multiplicando ambas transformadas entre sí y calculando la transformada inversa del resultado [6].

Dado que el tamaño del filtro es menor que el de la imagen, este debe completarse con ceros hasta alcanzar el mismo tamaño que la imagen antes de calcular su transformada. Los filtros se definen directamente en el espacio de Fourier sobre el propio espectro de la imagen original. Los valores de los píxeles en la imagen de la Transformada de Fourier $F(u, v)$, representan los componentes de frecuencia espacial, verticales y horizontales de la imagen original, de forma análoga a la frecuencia de una señal en el tiempo. Así, una señal sinusoidal con alta frecuencia, por ejemplo, varía muy rápidamente, mientras que otra con una frecuencia baja varía lentamente en el tiempo. De la misma forma, una imagen con una elevada frecuencia espacial en la dirección horizontal, contiene cambios frecuentes de intensidad en la dirección horizontal.

La Transformada de Fourier puede emplearse para diseñar operaciones de filtrado en imágenes en el dominio de las frecuencias. Ello presenta algunas ventajas con respecto a las convoluciones realizadas mediante máscaras en el dominio espacial. Por un lado, es muy eficiente cuando el tamaño del filtro es muy grande. Además, el diseño de los filtros es más intuitivo, ya que el ruido está asociado a determinadas frecuencias, fácilmente identificables en el espectro de Fourier.

Su aplicación principal en este campo es el diseño de filtros para la eliminación del ruido periódico. Para cumplir este objetivo, es necesario interpretar las imágenes transformadas e identificar los patrones debidos al ruido. Como resultado, la eficiencia de los filtros en el dominio de las frecuencias es muy superior y en ocasiones, la única alternativa viable de filtrado [2].

Filtros de paso alto:

El filtro paso alto se utiliza para eliminar o atenuar las bajas frecuencias de la imagen. Este puede diseñarse de forma sencilla mediante una máscara binaria circular con valor cero en el interior del círculo y valor uno en el exterior. Mientras mayor sea el radio del círculo, mayor será el intervalo de bajas frecuencias que se elimina y más severo será el efecto del filtrado [7].

Filtros de paso bajo:

El filtro paso bajo consiste en la eliminación o atenuación de las altas frecuencias de la imagen. Para ello, bastará definir una máscara binaria de forma que al aplicarla sobre el espectro de frecuencias mantenga los valores bajos, situados en el centro de la imagen y elimine los altos. Esto se consigue mediante círculos binarios de valor uno en el interior y cero en el exterior. Cuanto menor sea el radio del círculo, mayor será el efecto del filtro, ya que el intervalo de altas frecuencias que se elimina en la multiplicación $F(u, v) \cdot H(u, v)$ será mayor. Al contrario, cuanto mayor sea el radio del círculo, menor es el efecto del filtro [7].

1.3. Sistema Vismedic – Illustration.

Vismedic-Illustration es un software diseñado para la generación de ilustraciones interactivas a partir de datos volumétricos. El mismo está implementado en C++ utilizando el framework Qt. El sistema tiene una arquitectura modular basada en plugins que favorece la integración de nuevos componentes de una manera simple y eficiente. La arquitectura tiene interfaces claramente definidas que permiten la incorporación de nuevos algoritmos (plugins) con un mínimo esfuerzo. El sistema está compuesto por varios módulos que soportan todas las funcionalidades que posibilitan la generación de ilustraciones. En la **¡Error! La autoreferencia al marcador no es válida.** se observa una imagen del sistema Vismedic – Illustration en la cual se observa una ilustración de un cráneo humano generada con el sistema.

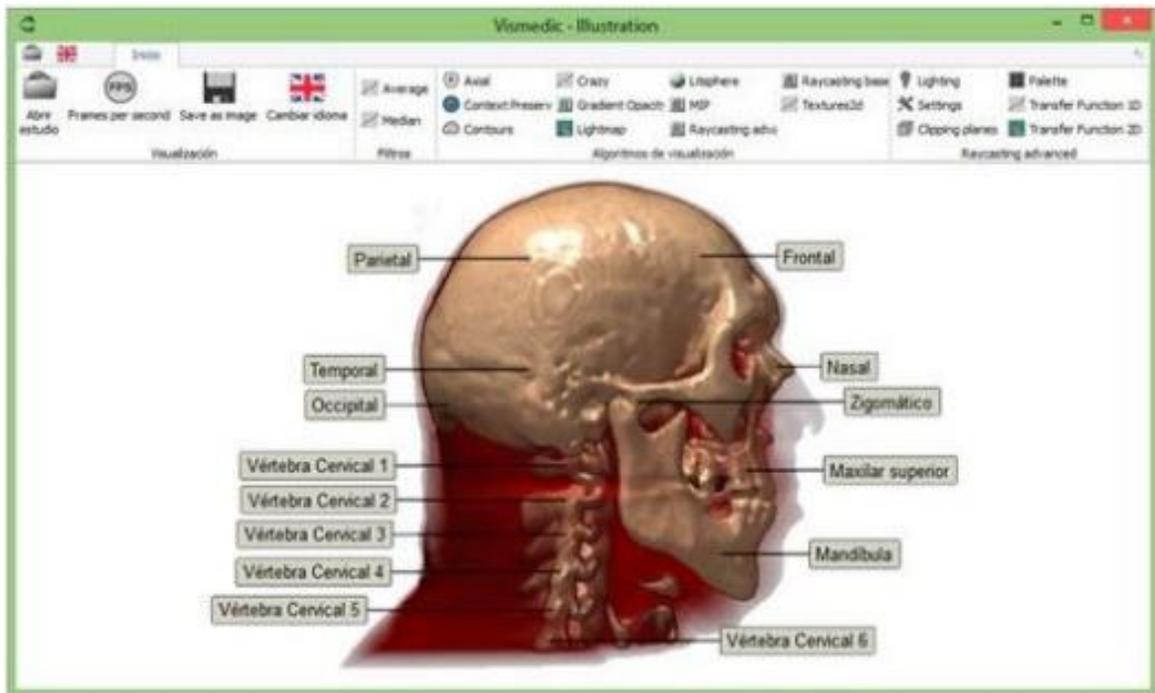


Figura. 1 Sistema Vismedic - Illustration para la generación de ilustraciones.

El sistema Vismedic – Illustration incluye diferentes módulos que permiten la generación de las ilustraciones. En la Figura. 2 se puede apreciar la arquitectura y los módulos del sistema Vismedic – Illustration. Los módulos principales del sistema están implementados en forma de *plugins*. Esto posibilita la extensión de funcionalidades de forma natural, debido a que cada módulo define las interfaces concretas que debe implementar cada nuevo algoritmo que se incorpore al sistema. Entre los módulos fundamentales del sistema se encuentran el de **Entrada – Salida**, el de **Visualización**, el de **Filtrado** y el de **Segmentación**. La presente investigación se centra en la etapa de **Filtrado**.

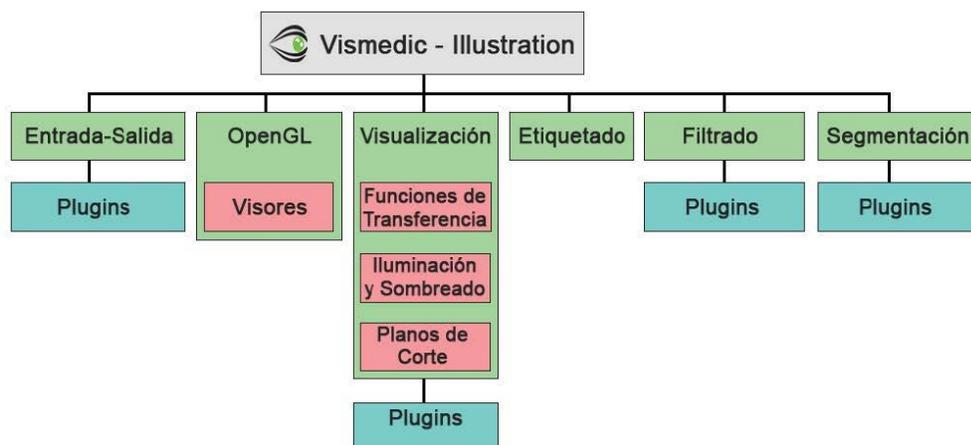


Figura. 2 Arquitectura del sistema Vismedic – Illustration.

1.4. Etapa de filtrado del sistema Vismedic – Illustration.

Uno de los módulos fundamentales del sistema Vismedic – Illustration es el de Filtrado. Este contiene filtros implementados para reducir el ruido presente en el volumen de datos, filtros para calcular el gradiente y detectar los bordes y filtros para mejorar el contraste. Estos filtros en su conjunto contribuyen a mejorar la calidad de la visualización y a resaltar detalles que en ocasiones no son apreciables en el volumen de datos original. A continuación, se explicarán con mayor profundidad los filtros implementados en el sistema Vismedic – Illustration.

Filtro del promedio

El filtro de promedio se utiliza para reducir el ruido presente en el volumen de datos que se visualiza. Este se basa en sustituir el valor de cada vóxel por la media aritmética de los valores de intensidad de los vóxeles en la vecindad definida por la máscara de convolución. Este filtro es simple, intuitivo y fácil de implementar. El caso más común se basa en una máscara de convolución como el que se muestra en la Figura. 3 La imagen de la izquierda representa los valores de intensidad de los píxeles en una imagen 2D, la imagen del centro representa la máscara para realizar la convolución y la imagen de la derecha representa los nuevos valores que se obtienen para la imagen original después de aplicar el filtro del promedio. La máscara que se muestra en la Figura. 3 (centro) contiene nueve coeficientes, debido a que se aplica sobre una imagen 2D. Sin embargo, este filtro en el sistema Vismedic – Illustration utiliza 27 coeficientes, debido a que se utiliza una matriz de 3x3x3 valores, todos con valor uno, debido a que además se utiliza la dimensión de profundidad (z) al trabajar con volúmenes de datos.

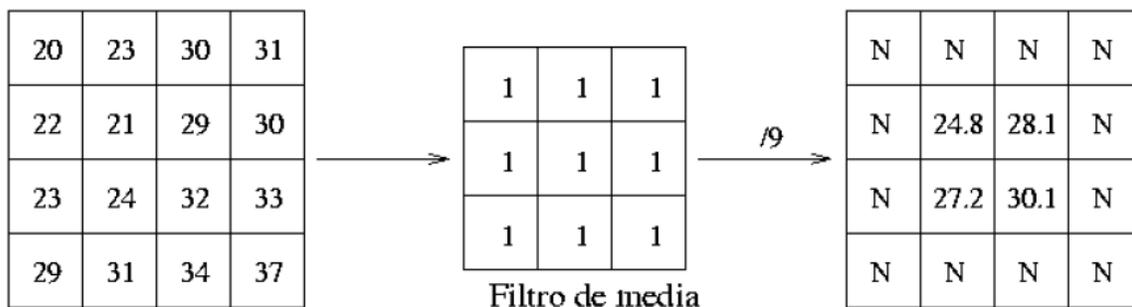


Figura. 3 Filtro del promedio. (izquierda) Valores de intensidad de una imagen. (centro) Máscara definida para el promedio. (derecha) Valor resultante del filtro del promedio para cuatro píxeles de la imagen representada.

El filtro del promedio es el más simple, intuitivo y fácil de implementar para suavizar imágenes. Este reduce la cantidad de variaciones de intensidad entre vóxeles vecinos. Sin embargo, el filtro del promedio es bastante sensible a cambios locales de intensidad y puede crear nuevos valores de intensidad que no aparecían en el volumen original.

Filtro de la mediana

Otro de los filtros de reducción de ruido implementado en Vismedic – Illustration es el filtro de la mediana. Este filtro se aplica para cada vóxel del volumen y se calcula sustituyendo el valor de intensidad de cada vóxel por la mediana de los valores de intensidad de los vóxeles vecinos definidos por la máscara de convolución. La mediana se calcula ordenando los valores de intensidad de los vóxeles vecinos de menor a mayor y seleccionando el que queda en medio. El filtro de la mediana ofrece muy buenos resultados para reducir ruido denominado sal y pimienta. En la Figura. 4 se puede apreciar como se calcula el valor de la mediana para dos píxeles de una imagen. Para ello se ordenan los valores de intensidad de los vóxeles vecinos definido por la máscara de convolución (9 en el ejemplo) y se toma el valor que ocupa el medio del arreglo de valores después del ordenamiento. El filtro de la mediana implementado en el sistema Vismedic – Illustration utiliza 27 vóxeles vecinos para el ordenamiento (3x3x3), debido a que se utiliza la dimensión de profundidad (z) al trabajar con volúmenes de datos.

$$\text{MED}(x) = y \quad X = \begin{bmatrix} 10 & 10 & 20 & 20 \\ 20 & 10 & 20 & 20 \\ 10 & 10 & 10 & 20 \\ 10 & 10 & 20 & 20 \end{bmatrix} ; \quad Y = \begin{bmatrix} 10 & 10 & 20 & 20 \\ 10 & 10 & 20 & 20 \\ 10 & 10 & 20 & 20 \\ 10 & 10 & 20 & 20 \end{bmatrix}$$

$$\begin{aligned}
 m1[3 \times 3] &= [10, 10, 20, 20, 10, 20, 10, 10, 10] \\
 \text{MED}[m1] &= [10, 10, 10, 10, 10, 10, 20, 20, 20] = 10 \\
 \\
 m2[3 \times 3] &= [10, 20, 20, 10, 10, 20, 10, 20, 20] \\
 \text{MED}[m2] &= [10, 10, 10, 10, 20, 20, 20, 20, 20] = 20
 \end{aligned}$$

Figura. 4 Filtro de la mediana.

Filtros para definir bordes y gradientes

Filtro de Sobel

El operador Sobel es utilizado en procesamiento de imágenes, especialmente en algoritmos de detección de bordes. Técnicamente es un operador diferencial discreto que calcula una aproximación al gradiente de la función de intensidad de una imagen. Para cada punto de la imagen a procesar, el resultado del operador Sobel es tanto el vector gradiente correspondiente como la norma de este vector.

El operador Sobel calcula el gradiente de la intensidad de una imagen en cada punto. Así, para cada punto, este operador muestra la magnitud del mayor cambio posible, la dirección de este y el sentido desde oscuro a claro.

El resultado muestra la medida en que cambia la intensidad en cada vóxel del volumen y, en consecuencia, ofrece para cada vóxel la probabilidad de representar un borde, así como su orientación. En la práctica, el cálculo de la magnitud es más fiable y sencillo de interpretar que el cálculo de la dirección y sentido.

Matemáticamente, el gradiente de una función de dos variables (en este caso, la función de intensidad de la imagen) para cada punto es un vector bidimensional cuyos componentes están dados por las primeras derivadas de las direcciones vertical y horizontal. Para cada punto de la imagen, el vector gradiente apunta en dirección del incremento máximo posible de la intensidad, y la magnitud del vector gradiente corresponde a la cantidad de cambio de la intensidad en esa dirección. Esto implica que el resultado de aplicar el operador Sobel sobre una región con intensidad de imagen constante es un vector de magnitud cero, y el resultado de aplicarlo en un punto sobre un borde es un vector que cruza el borde (perpendicular) cuyo sentido es de los puntos más oscuros a los más claros. El filtro de Sobel implementado utiliza una máscara de 6 valores para la dimensión “x” y 6 valores para la dimensión “y”. Por tanto, para cada vóxel analizado, se consultan 12 valores para calcular el nuevo valor resultante. En la Figura. 5 se muestran las máscaras de convolución correspondientes al filtro de Sobel en 2D. este se puede modificar para volúmenes de datos incorporando la dimensión “z” y su correspondiente máscara de convolución.

$$\begin{aligned}
 G_{jk} &= |G_x| + |G_y| \\
 G_x &= F_{j+1,k+1} + 2F_{j+1,k} + F_{j+1,k-1} - (F_{j-1,k+1} + 2F_{j-1,k} + F_{j-1,k-1}) \\
 G_y &= F_{j-1,k-1} + 2F_{j,k-1} + F_{j+1,k-1} - (F_{j-1,k+1} + 2F_{j,k+1} + F_{j+1,k+1})
 \end{aligned}$$

$$\begin{aligned}
 X \text{ mask} &= \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \\
 Y \text{ mask} &= \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}
 \end{aligned}
 = \begin{bmatrix} -2 & -2 & 0 \\ -2 & 0 & 2 \\ 0 & 2 & 2 \end{bmatrix}$$

Figura. 5 Máscaras de convolución del filtro de Sobel.

Filtro Laplaciano

El filtro Laplaciano es muy útil para realzar los bordes en todas direcciones. Este filtro trabaja con la segunda derivada, lo que permite obtener mejores resultados, a pesar del aumento del ruido que se produce en la imagen [7]. El uso de este filtro supone algunas ventajas como pueden ser:

- Aumenta la nitidez, destacando los detalles de la imagen, al afinar las líneas de transición de niveles de gris.
- El valor central de la máscara de los filtros laplacianos es negativo entre 4 a 8 veces menor que los parámetros del entorno. Esto crea el problema que la imagen queda muy oscura después de aplicar el filtro.

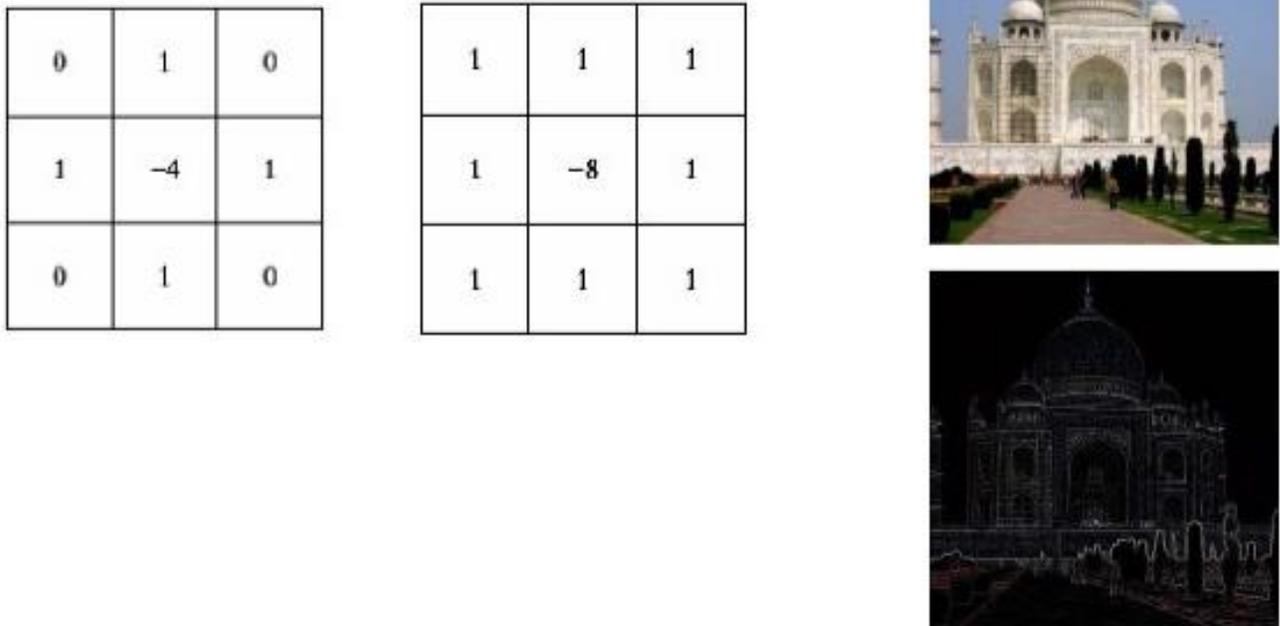


Figura. 6 Ejemplo de aplicar el filtro Laplaciano. Imagen cortesía de [3].

El filtro Laplaciano en el sistema Vismedic resalta los bordes y mejora el nivel de detalle de las estructuras presentes en los volúmenes de datos que se visualizan.

Filtro transformación logarítmica

El filtro de transformación logarítmica es uno de los filtros implementados en Vismedic – Illustration para mejorar el contraste de los niveles de grises de los volúmenes de datos. Este filtro se calcula mediante la ecuación siguiente:

$$s = c \log(1 + r)$$

Donde c es una constante y $r \geq 0$.

Este filtro se utiliza para expandir los valores de vóxeles oscuros a vóxeles más claros en un volumen, mientras se comprimen los valores de alto nivel. Se usa para visualizar bajos niveles de intensidad con mayor margen dinámico y para realzar el detalle en los niveles de gris [3].

1.5. Herramientas y metodología de desarrollo.

El sistema Vismedic – Illustration desde sus inicios, definió un entorno de desarrollo, así como las herramientas necesarias para que futuros desarrolladores contribuyeran a su crecimiento con nuevas funcionalidades. Esta propuesta incluye el framework Qt basado en el lenguaje C++ para el desarrollo del sistema, el Qt Creator como IDE y la biblioteca OpenGL para la visualización, la cual ya viene integrada en el framework Qt. Para esta investigación, se decidió utilizar las herramientas definidas anteriormente para el sistema Vismedic – Illustration.

Como elementos adicionales, se definieron la herramienta y el lenguaje de modelado. Como herramienta de modelado se definió Visual Paradigm. Este se basa en UML y soporta el ciclo de vida completo del desarrollo de software. Es un producto de probada calidad que soporta varios idiomas y con licencia gratuita y comercial. La herramienta está diseñada para una amplia gama de usuarios interesados en construir sistemas de software fiables con el uso del paradigma orientado a objetos.

Como lenguaje de modelado se definió UML. Este es un lenguaje de modelado visual que se usa para especificar, construir, documentar y visualizar artefactos de un sistema de software. El mismo está compuesto por diversos elementos gráficos que se combinan para conformar diagramas. Su objetivo es visualizar, especificar, construir y documentar los artefactos que se crean durante el proceso de desarrollo. Este brinda además la posibilidad de modelar sistemas utilizando técnicas orientadas a objetos.

Metodología de desarrollo

Para la selección de la metodología se tuvieron en cuenta las características del proyecto y el tiempo restringido para el desarrollo de la investigación. Estos elementos influyeron decisivamente en la selección de una metodología ágil para guiar el proceso de desarrollo [11]. El desarrollo de nuevas funcionalidades para el sistema Vismedic – Illustration se adapta a las características para las que se recomienda una metodología ágil de desarrollo, pues el tiempo de desarrollo es corto y los requisitos pueden llegar a cambiar en dependencia de las necesidades del cliente. Por esta razón, se seleccionó la metodología XP como guía para el desarrollo de la solución propuesta.

La metodología XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo. Algunas de las herramientas principales de XP son las Historias de Usuarios, los Casos de Prueba de Aceptación, las Tarea de Ingeniería y las Tarjetas CRC [9].

Consideraciones parciales del capítulo

Una vez concluido el capítulo y realizado el estudio de estado del arte con respecto al filtrado de imágenes en el dominio de las frecuencias es posible arribar a las siguientes conclusiones:

- Los filtros que operan en el dominio de las frecuencias permiten una reducción del costo computacional comparado con los filtros que operan en el dominio del espacio. Esta se hace más evidente a medida que aumenta el tamaño de la máscara de convolución de los filtros en el dominio del espacio.
- La solución propuesta incluye el diseño de filtros de reducción de ruido, filtros para mejorar la calidad de la imagen y filtros para el cálculo del gradiente. Estos se agruparán en tres categorías: los filtros de paso alto, los filtros de paso bajo y los filtros para el cálculo de gradiente.

CAPÍTULO 2

Características y diseño del sistema

En este capítulo se definen los requisitos funcionales y no funcionales. Se plantean las Historias de Usuario (HU), se realiza la estimación de esfuerzo por HU, el plan de iteraciones, las tarjetas CRC y las tareas de ingeniería por iteración. Finalmente se explican los filtros implementados en el dominio de las frecuencias y se profundiza en las cuestiones más relevantes de la solución propuesta.

2.1 Requisitos funcionales.

Los requisitos funcionales representan las funcionalidades del sistema 1 Estos se transforman en casos de uso y se modelan posteriormente. Los siguientes requisitos responden a las funcionalidades que el sistema debe tener una vez concluida su implementación [13].

Tabla 1 Requisitos funcionales.

No	Nombre	Complejidad	Prioridad
RF1	Cargar imagen en QT.	Baja	Alta
RF2	Calcular Transformada de Fourier de la imagen.	Alta	Alta
RF2.1	Convertir imagen QT a matriz numérica.	Media	Alta
RF2.2	Aplicar transformada de Fourier a la matriz numérica.	Alta	Alta
RF2.3	Convertir matriz numérica a imagen QT.	Alta	Alta
RF3	Calcular Transformada de Fourier de la máscara del filtro.	Alta	Alta
RF3.1	Multiplicar la imagen transformada y la máscara de filtrado haciendo uso de las propiedades de la Transformada de Fourier.	Alta	Alta
RF3.2	Calcular la transformada inversa de la imagen filtrada	Media	Alta
RF4	Mostrar imagen.	Baja	Baja

2.2 Requisitos no funcionales.

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener y representan las características que hacen al producto atractivo, usable, rápido y confiable. Seguidamente se enumeran los requisitos no funcionales del módulo a desarrollar [13].

Tabla 2 Requisitos no funcionales.

No	Tipo	Nombre	Complejidad	Prioridad
RnF1	Software	El modelo de microprocesador será Intel Core i3 a 3.40 GHz o superior. La memoria RAM será de 4GB. Espacio libre en disco duro de 512 Mb	Baja	Baja
RnF2	Hardware	El modelo de microprocesador será Intel Core i3 a 3.40 GHz o superior. La memoria RAM será de 4GB. Espacio libre en disco duro de 512 Mb	Baja	Baja
RnF3	Interfaz externa	La interfaz de usuario debe ser sencilla y amigable, lo que le permitirá al usuario interactuar de una forma rápida y cómoda con las funcionalidades del módulo.	Baja	Baja
RnF4	Restricciones en el diseño e implementación	Se empleará como lenguaje de programación C++ bajo el paradigma de Programación Orientada a Objetos y el Framework Qt para el diseño de las interfaces gráficas y programación.	Alta	Alta

:

2.3 Historias de Usuarios.

Historia de usuario	
Número: 1	Nombre: Implementación de un conjunto de plugins para incorporar filtros en el dominio de las frecuencias.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Danny Guerra Enrique	
Descripción: El usuario deberá poder hacer uso de filtros en el dominio de la frecuencia a través del sistema Vismedic mediante el panel de plugins.	
Observaciones: La cantidad de filtros a implementar son 6 Los filtros serán los mismos ya implementados en el dominio del espacio. Se deben realizar pruebas para garantizar la efectividad de los filtros.	
Programador responsable: Danny Guerra Enrique	

2.4 Estimación de esfuerzo por Historias de Usuario.

Se realiza la estimación de esfuerzo que arroja para cada HU, con el objetivo de obtener un correcto desarrollo del sistema. A continuación, se muestra la estimación de esfuerzo por Historias de Usuario.

Tabla 3 Estimación de esfuerzo por Historia de Usuario.

Iteración		Historias de usuario	Puntos estimados (semanas)
1	1	Cargar imagen en QT.	1
	2	Calcular Transformada de Fourier de la imagen	2
2	3	Convertir imagen QT a matriz numérica.	3
	4	Aplicar transformada de Fourier a la matriz numérica.	3
	5	Calcular Transformada de Fourier de la máscara del filtro.	3
3	6	Obtener la imagen resultante filtrada mediante el uso de la Transformada inversa de Fourier	3
	7	Convertir matriz numérica a imagen QT.	3
	8	Multiplicar la imagen transformada y la máscara de filtrado haciendo uso de las propiedades de la Transformada de Fourier.	3
	9	Calcular la transformada inversa de la imagen filtrada	3
	10	Mostrar Imagen.	3
Total			27

2.5 Desarrollo del plan de iteraciones.

Una vez definidas las HU y realizada una previa estimación de esfuerzos, se procede a la planificación de la etapa de implementación del sistema. En este espacio, se crea el plan de iteraciones, donde se especifica la prioridad con que se implementarán las HU organizadas por iteraciones.

Plan de duración de las iteraciones

A continuación, se presenta el plan de duración de las iteraciones. Este plan, tiene como finalidad, mostrar la duración de cada iteración, así como el orden en que serán implementadas las HU en cada iteración, como se muestra en la tabla siguiente:

Tabla 4 Plan de duración de las iteraciones.

Iteración		Historias de usuario	Duración (semanas)
1	1	Convertir imagen recibida a imagen QT	3.0
	2	Convertir imagen QT a matriz numérica	
2	3	Hacer uso de la Transformada de Fourier para obtener la imagen en el dominio de la frecuencia	6.0
	4	Hacer uso de la Transformada de Fourier para obtener la máscara de filtrado en el dominio de las frecuencias.	

Plan de entregas

En el plan de entregas que se muestra a continuación se especifica la fecha de inicio y de fin de cada iteración que se realizará. Por cada iteración se debe realizar una versión del sistema.

Tabla 5 Plan de entregas.

Iteraciones	Fecha de inicio	Fecha de fin
Iteración 1	1ro de enero de 2020	23 de enero de 2020
Iteración 2	23 de enero de 2020	5 de marzo de 2020
Iteración 3	5 de marzo de 2020	16 de abril de 2020

2.6 Tarjetas CRC.

Las tarjetas CRC permiten romper con el modo de procedimiento y de pensamiento para apreciar mejor la tecnología de objetos. También permiten que todo el equipo pueda contribuir al diseño del proyecto y aumentar el número de ideas incorporadas.

Tabla 6 Tarjeta CRC #1.

Tarjeta CRC	
Clase: FourierFilter::FFT	
Responsabilidad	Colaboración
applyFourierFilter()	FourierFilter() applyFilter() setPixmap() show()

2.7 Tareas de ingeniería para la Iteración I.

Las tareas de Ingeniería permiten organizar mejor el trabajo ingenieril ya que separan las Historias de Usuario y les asignan fechas específicas de cumplimiento. Además, presentan la información con claridad para el desarrollador permitiendo una mejor comprensión de la tarea y facilitando su cumplimiento.

Tabla 7 Tarea de ingeniería # 1.

Tarea	
Número de tarea: 1	Número de Historia de usuario: 1
Nombre de la tarea: Convertir imagen recibida a imagen QT	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 1 de enero de 2020	Fecha de fin: 8 de enero de 2020
Programador responsable: Danny Guerra Enrique	
Descripción: procesar imágenes al formato de visualización del framework QT	

2.8 Modelo de filtrado en el dominio de las frecuencias.

Los filtros frecuenciales procesan la imagen en el dominio de la frecuencia. Primero se debe obtener la Transformada de Fourier de la imagen luego se multiplica por la función del filtro y por último se aplica la transformada inversa para obtener la imagen filtrada en el dominio espacial.

Todos los filtros frecuenciales se pueden implementar en el dominio espacial siempre y cuando exista un kernel de convolución simple [12]. Siempre es conveniente trabajar en el dominio espacial pues es computacionalmente mejor. Una convolución puede aplicarse transformando ambos, imagen y filtro, al dominio de la frecuencia, multiplicando ambas transformadas entre sí y calculando la Transformada inversa del resultado, tal y como se indica gráficamente en el esquema de la Figura. 7.

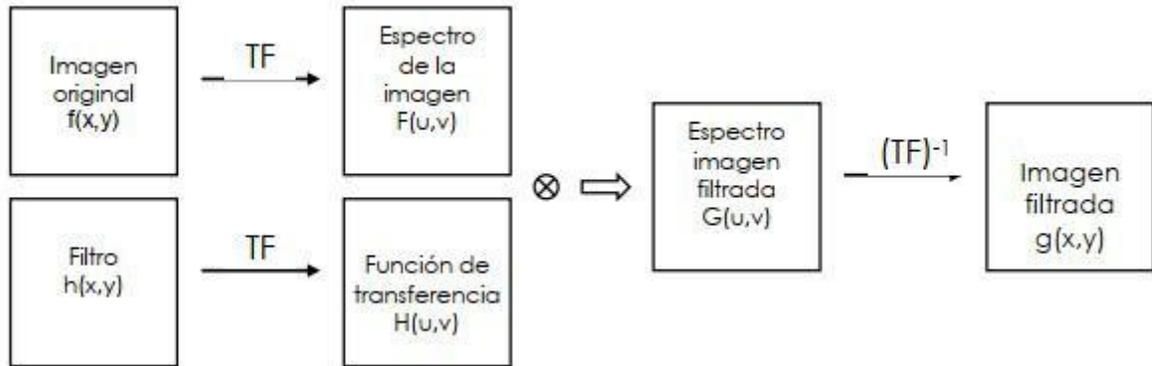


Figura. 7 Modelo de filtrado en el dominio de las frecuencias.

La función de transferencia se refiere a la respuesta en frecuencias de un filtro. Dado que el tamaño del filtro es menor que el de la imagen, deberá completarse con ceros hasta alcanzar el mismo tamaño que la imagen antes de calcular su transformada. Los filtros se definen directamente en el espacio de Fourier sobre el propio espectro de la imagen original. Así, una señal sinusoidal con alta frecuencia, por ejemplo, varía muy rápidamente, mientras que otra con una frecuencia baja varía lentamente en el tiempo. De la misma forma, una imagen con una elevada frecuencia espacial en la dirección horizontal, contiene cambios frecuentes de intensidad en la dirección horizontal.

En la Figura. 8 se muestran ejemplos de tres imágenes con sus respectivas transformadas. En ellas se pueden ver tres distribuciones espaciales distintas y el efecto que tienen sobre la representación en forma de imagen de sus transformadas, incluyendo el claro componente direccional de las frecuencias.

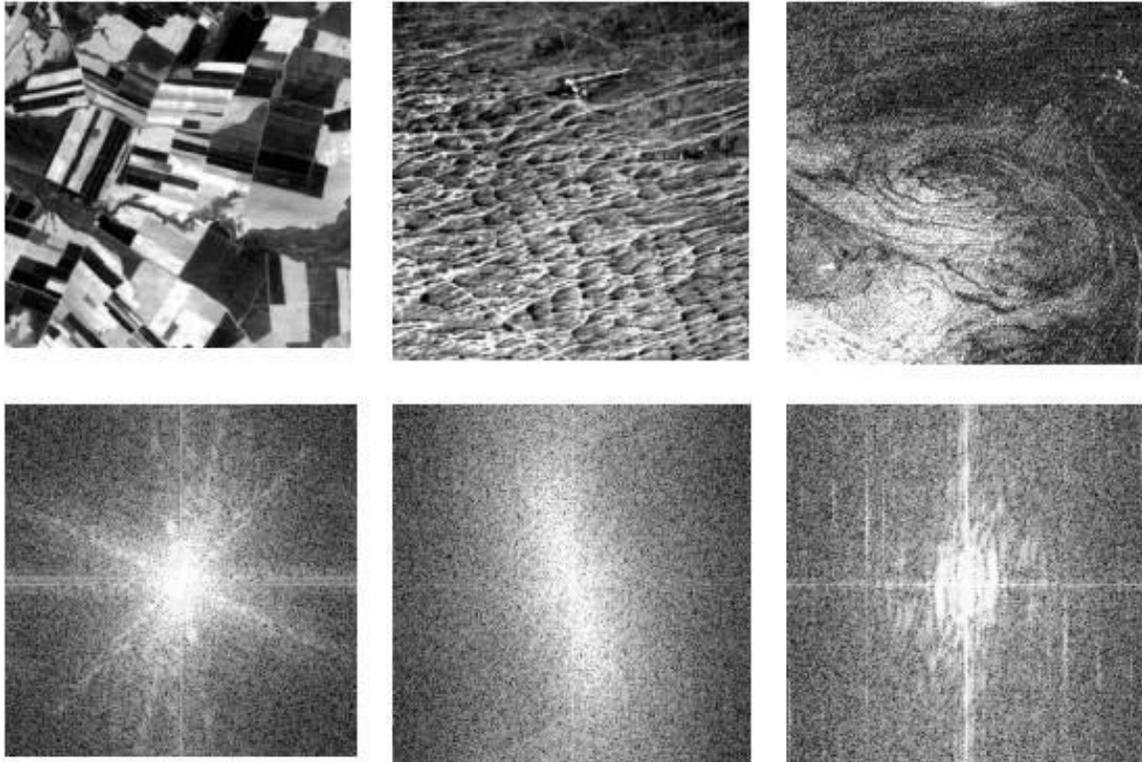


Figura. 8 Ejemplos de tres imágenes con sus respectivas transformadas.

2.9 Filtros propuestos como solución.

En esta sección se explicarán los filtros que se proponen como solución. Los filtros propuestos se pueden clasificar en los grupos de filtros de paso alto, filtros de paso bajo y filtros para la detección de bordes. En las siguientes secciones se profundizará un poco más en los elementos específicos para cada tipo de filtro propuesto.

2.9.1 Filtros de paso alto.

Su objetivo es suavizar la imagen, son útiles cuando se supone que la imagen tiene gran cantidad de ruido y se quiere eliminar. También pueden utilizarse para resaltar la información correspondiente a una determinada escala (tamaño de la matriz de filtrado); por ejemplo, en el caso de que se quiera eliminar la variabilidad asociada a los tipos de cubierta presentes en la imagen uniformizando de esta manera su respuesta. Existen varias posibilidades:

- **Filtro de la media**, asigna al píxel central la media de todos los píxeles incluidos en la ventana. La matriz de filtrado estaría compuesta por unos y el divisor sería el número total de elementos en la matriz.

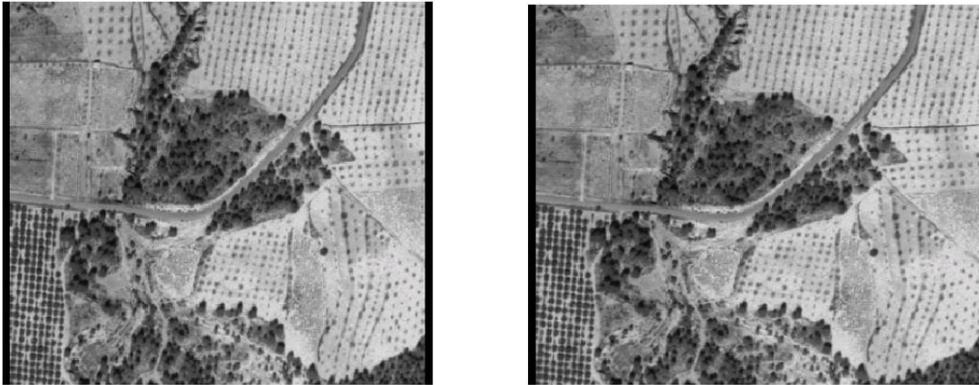


Figura. 9 Ejemplo de aplicación del filtro de la media.

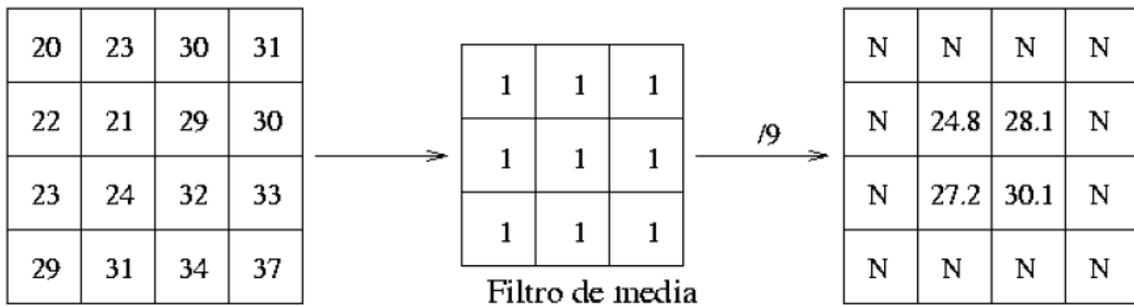


Figura. 10 Aplicación del filtro media sobre una matriz numérica.

- **Filtro de media ponderada**, los elementos de la matriz de filtrado no son todos 1 sino que se pondera más a uno de ellos (generalmente el píxel central) para obtener un resultado más parecido a la imagen original y evitar que aparezca borrosa.
- **Filtro de la mediana**, tiene la ventaja de que el valor final del píxel es un valor real presente en la imagen y no un promedio, de este modo se reduce el efecto borroso que tienen las imágenes que han sufrido un filtro de media. Además, el filtro de la mediana es menos sensible a valores extremos. El inconveniente es que resulta más complejo de calcular ya que hay que ordenar los diferentes valores que aparecen en los píxeles incluidos en la ventana y determinar cuál es el valor central.
- **Filtros gaussianos**. Simulan una distribución gaussiana binaria. El valor máximo aparece en el píxel central y disminuye hacia los extremos tanto más rápido cuanto menor sea el parámetro de desviación típicas. El resultado será un conjunto de valores entre 0 y 1. Para transformar la matriz a una matriz de números enteros se divide toda la matriz por el menor de los valores obtenidos.

1	4	7	4	1
4	20	33	20	4
7	33	55	33	7
4	20	33	20	4
1	4	7	4	1

Figura. 11 Filtro Gaussiano.

Un filtro paso alto se diseña mediante una máscara binaria circular, tal que dentro del círculo tenga como valor cero, y fuera del mismo la unidad. Cuanto mayor sea el radio del círculo, mayor será el intervalo de bajas frecuencias que se elimina y, por lo tanto, más severo será el efecto del filtrado, al contrario de lo que ocurre en el diseño de filtros paso bajo.

2.9.2 Filtros de paso bajo.

Su objetivo es resaltar las zonas de mayor variabilidad eliminando lo que sería la componente media, precisamente la que detectan los filtros de paso bajo. Por otra parte, la respuesta de cada pixel está contaminada por la de los pixeles vecinos ya que la radiación reflejada por un pixel se reparte hacia los pixeles vecinos. Los filtros de paso alto consiguen también eliminar en parte esta contaminación. Existen diversos métodos:

- **Sustracción de la media.** Si se considera que un filtro de paso bajo sirve para resaltar componentes a gran escala eliminando la variabilidad local, si a la imagen original se le resta el resultado de pasarle un filtro de paso bajo se consigue resaltar esa variabilidad local. La matriz de filtrado de este filtro menos media puede calcularse directamente restando a la matriz de filtrado identidad la matriz de filtrado de la media.

-1	-1	-1	DIV 9
-1	8	-1	
-1	-1	-1	

Figura. 12 Filtro de sustracción de la media.

Filtros basados en las derivadas. La derivada de una función $y = f(x)$ es el incremento de y para cada incremento infinitesimal de x . En el caso de Modelo Digital de Elevaciones (MDE) la derivada es la pendiente. La segunda derivada es la derivada de la derivada, en el caso de un MDE brinda información acerca de la forma (ladera recta, cóncava o convexa, valle, cresta o cima) del terreno. En el caso de una imagen de satélite ofrece una medida sobre la magnitud de la variación entre píxeles contiguos.

2.9.3 Filtros para la detección de bordes.

Uno de los filtros más utilizados es el detector de bordes de Sobel, el cual realiza la variación entre filas y columnas. A continuación, se muestran ejemplos de máscaras que se utilizan al aplicar el filtro de Sobel. La Figura. 13 en la imagen izquierda muestra la máscara correspondiente a la dirección "x", mientras que la imagen derecha muestra la máscara correspondiente a la dirección "y".

-1.0	0	1.0	-1.0	-2.0	-1.0
-2.0	0	2.0	0	0	0
-1.0	0	1.0	1	2	1

Figura. 13 Ejemplos de máscaras de Sobel: (izquierda) Máscara para la dirección "x". (Derecha) Máscara para la dirección "y".

La Figura. 14 muestra un ejemplo de aplicación del filtro de Sobel. En esta imagen se puede apreciar cómo se resaltan los bordes al aplicar este filtro sobre una imagen.



Figura. 14 Ejemplo de aplicación del filtro Sobel.

2.10 Diagrama de clases.

La Figura. 15 Diagrama de clases. representa una propuesta de diagrama de clases que puede servir como guía para implementar los filtros propuestos. En este diagrama el método más importante es el método `applyFilter(const QImage img) : QImage`, el cual recibe la imagen que se desea filtrar y devuelve la imagen obtenida como resultado de aplicar el filtro.

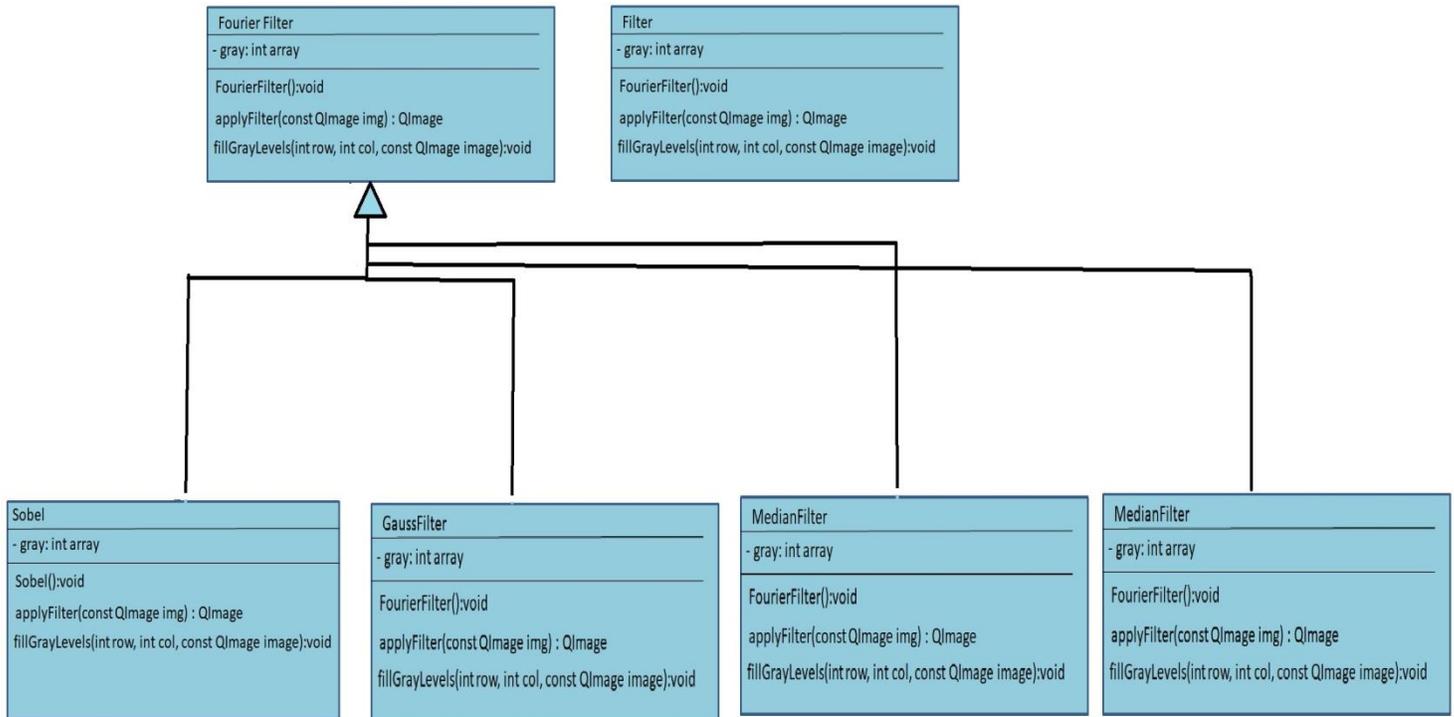


Figura. 15 Diagrama de clases.

2.11 Integración con el sistema Vismedic-Illustration.

Para integrar los filtros propuestos con el sistema Vismedic – Illustration es necesario adaptar los filtros propuestos para volúmenes de datos, porque al trabajar con volúmenes se adiciona la dimensión “z” de profundidad. Además, es necesario que cada filtro propuesto se adicione como un plugin al módulo de **Filtrado** del sistema Vismedic – Illustration. Este módulo tiene interfaces claramente definidas para incorporar nuevos algoritmos de filtrado. Por lo tanto, para integrar los filtros propuestos basta con adicionar un plugin por cada filtro e implementar la interfaz concreta de los algoritmos de filtrado adaptado para volúmenes de datos.

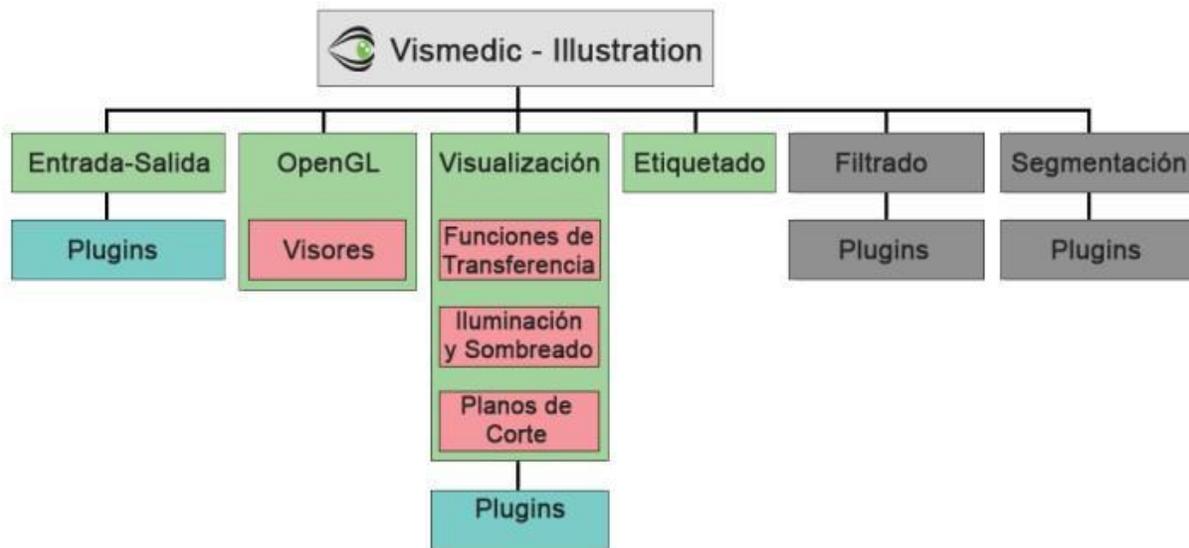


Figura. 16 Arquitectura del sistema Vismedic - Illustration.

Consideraciones parciales del capítulo

En este capítulo se analizaron los principales elementos de la solución propuesta para el filtrado de imágenes mediante el empleo de la Transformada de Fourier. Una vez concluido el desarrollo del capítulo, es posible arribar a las siguientes conclusiones:

- Los filtros propuestos se pueden implementar en el dominio de las frecuencias y se clasifican en filtros de paso alto, filtros de paso bajo y filtros para la detección de bordes.
- La propuesta de filtros debe adaptarse para su incorporación a volúmenes de datos y su correcta utilización en el sistema Vismedic – Illustration.

Conclusiones

Una vez concluido el presente trabajo, es posible arribar a las siguientes conclusiones:

- Se definió una propuesta de filtros en el dominio de las frecuencias con el objetivo de agilizar la etapa de filtrado de imágenes digitales.
- El dominio de las frecuencias constituye una alternativa más eficiente en cuanto al costo computacional que los algoritmos de filtrado que trabajan en el dominio del espacio.

Recomendaciones

Como trabajo futuro se proponen las siguientes líneas de trabajo:

- Extender la propuesta de filtros a volúmenes de datos para su posterior integración al sistema Vismedic – Illustration.
- Implementar la Transformada Rápida de Fourier para mejorar el rendimiento de la aplicación de los filtros propuestos en el dominio de las frecuencias.

Bibliografía

- [1] Mariana del Fresno. Modelos y algoritmos aplicados a la gestión y análisis de imágenes médicas. Ed. por Universidad Nacional del Centro de la Provincia de Buenos Aires. 25 de mayo de 2014.
url: <http://www.unicen.edu.ar/content/modelos-y-algoritmos-aplicados-la-gesti%C3%B3n-y-an%C3%A1lisis-de-im%C3%A1genes-m%C3%A9dicas>
- [2] Bernhard Preim y Dirk Bartz. Visualization in medicine: theory, algorithms, and applications. Morgan Kaufmann, 2007
- [3] Rafael C Gonzalez y Richard E Woods. Digital Image Processing. 2002
- [4] Paul Bourke. Image Filtering in the Frequency Domain. 20 de febrero de 2020.
url: <http://paulbourke.net/miscellaneous/dft/>
- [5] Klaus Engel y col. Real-Time Volume Graphics. CRC Press, 2006
- [6] Remote Sensing. Models and Methods for Image Processing. Robert A. Schowengerdt. Academic Press, 1997.
- [7] Image processing: principles and applications. John Wiley & Sons. Alegre, E., Sánchez, L., Fernández, R. Á., Mostaza, J. C. (2003).
- [8] Fast Fourier Transform. Kobenhavn, 1983.
- [9] Eing, Orlando Pamplona., and Rogerio Cid Bastos. Adaptações Na Metodologia Ágil De Desenvolvimento De Software XP (programação Extrema). Florianópolis, SC, 2003.
- [10] Fourier, Joseph. Oeuvres de Fourier. Paris : Gauthier-Villars, 1888.
- [11] Marin, Leonardo Florez, and Felipe Grisales Tobon. Formulación de criterios para la selección de metodologías de desarrollo de software. Colombia: Universidad Tecnológica de Pereira. Facultad de Ingenierías Eléctrica, Electrónica, Física y Ciencias de la Computación. Ingeniería de Sistemas y Computación, 2014.
- [12] María, Elvira Rodríguez José, and Cesáreo Lorenzo González. La Transformada de Laplace y el Teorema de Convolución, un método útil para la resolución de ecuaciones diferenciales: aplicación a algunos problemas interesantes. S.l.: S.n., 1998.
- [13] Pantaleo, Guillermo, Ludmila Rinaudo, and Asenjo Germán Asenjo. Ingeniería de Software. México: Alfaomega, 2015.

Anexos

Tabla 8 Historia de Usuario #2.

Historia de usuario	
Número: 2	Nombre: Cargar imagen en QT.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Descripción: Transformar imágenes de formato JPEG o PNG a QIMAGES	
Observaciones: Mantener tamaño de la imagen	

Tabla 9 Historia de Usuario #3.

Historia de usuario	
Número: 3	Nombre: Convertir imagen QT a matriz numérica.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar imagen QT en matriz numérica para poder aplicarle la transformada de Fourier.	
Observaciones: La matriz debe permitir datos complejos	

Tabla 10 Historia de Usuario #4.

Historia de usuario	
Número: 4	Nombre: Calcular Transformada de Fourier de la imagen.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar imagen del dominio del espacio al dominio de la frecuencia.	
Observaciones:	

Tabla 11 Historia de Usuario #5.

Historia de usuario	
Número: 5	Nombre: Aplicar transformada de Fourier a la matriz numérica.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar matriz numérica del dominio del espacio al dominio de la frecuencia.	
Observaciones: La matriz resultante contiene números complejos.	

Tabla 12 Historia de Usuario #6.

Historia de usuario	
Número: 6	Nombre: Calcular Transformada de Fourier de la máscara del filtro.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar la máscara de filtrado al dominio de la frecuencia.	
Observaciones: rellenar con ceros hasta alcanzar el tamaño de la imagen a filtrar	

Tabla 13 Historia de Usuario #7.

Historia de usuario	
Número: 7	Nombre: Multiplicar la imagen transformada y la máscara de filtrado haciendo uso de las propiedades de la Transformada de Fourier.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Danny Guerra Enrique	
Descripción:	

Tabla 14 Historia de Usuario #8.

Historia de usuario	
Número: 8	Nombre: Calcular la transformada inversa de la imagen filtrada.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Danny Guerra Enrique	
Descripción: Aplicar la transformada inversa de Fourier a la matriz numérica de frecuencias para obtener la matriz en el dominio del espacio y luego convertir a Qimage.	

Tabla 15 Historia de Usuario #9.

Historia de usuario	
Número: 9	Nombre: Calcular la transformada inversa de la imagen filtrada.
Usuario: Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 3
Programador responsable: Danny Guerra Enrique	
Descripción: Aplicar la transformada inversa de Fourier a la matriz numérica de frecuencias para obtener la matriz en el dominio del espacio y luego convertir a imagen QT.	

Tareas de ingeniería para la Iteración I.

Tabla 16 Tarea de ingeniería #2.

Tarea	
Número de tarea: 2	Número de Historia de usuario: 2
Nombre de la tarea: Cargar imagen en QT.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha de inicio: 8 de enero de 2020	Fecha de fin: 22 de enero de 2020
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar imágenes de formato JPEG o PNG a QIMAGES	

Tabla 17 Tarea de ingeniería #3.

Tarea	
Número de tarea: 3	Número de Historia de usuario: 3
Nombre de la tarea: Hacer uso de la Transformada de Fourier para obtener la imagen en el dominio de la frecuencia	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Fecha de inicio: 23 de enero de 2020	Fecha de fin: 13 de febrero de 2020
Programador responsable: Danny Guerra Enrique	
Descripción: Obtener imágenes en el dominio de la frecuencia haciendo uso de la Transformada de Fourier	

Tabla 18 Tarea de ingeniería #4.

Tarea	
Número de tarea: 4	Número de Historia de usuario: 4
Nombre de la tarea: Convertir imagen QT a matriz numérica.	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Fecha de inicio: 13 de febrero de 2020	Fecha de fin: 5 de marzo de 2020
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar imagen QT en matriz numérica para poder aplicarle la transformada de Fourier.	

Tareas de ingeniería para la Iteración III.

Tabla 19 Tarea de ingeniería #5.

Tarea	
Número de tarea: 5	Número de Historia de usuario: 5
Nombre de la tarea: Calcular Transformada de Fourier de la imagen.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 5 de marzo de 2020	Fecha de fin: 26 de marzo de 2020
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar imagen del dominio del espacio al dominio de la frecuencia.	

Tabla 20 Tarea de ingeniería #6.

Tarea	
Número de tarea: 6	Número de Historia de usuario: 6
Nombre de la tarea: Aplicar transformada de Fourier a la matriz numérica	
	Puntos estimados: 1
Fecha de inicio: 5 de marzo de 2020	Fecha de fin: 26 de marzo de 2020
Programador responsable: Danny Guerra Enrique	
Descripción: Transformar matriz numérica del dominio del espacio al dominio de la frecuencia.	