



Universidad
Zaragoza

Trabajo Fin de Grado

Métodos de banda estrecha para reconstrucción sin
línea de visión eficiente

Narrow-band methods for efficient non-line-of-sight
reconstruction

Autor

Jorge Aranda Sanz

Director

Julio Marco Murria

Titulación

Grado en Ingeniería informática

ESCUELA DE INGENIERÍA Y ARQUITECTURA
2024

AGRADECIMIENTOS

Quiero agradecer a Julio Marco Murria por todo el apoyo y motivación durante estos meses. Porque no es solo lo que se enseña, sino con la motivación que se enseña. Gracias sobretodo por haberme abierto los ojos a este campo tan fascinante.

También, a esas amistades que haces un día cualquiera, en clase, sin saber que van a ser de por vida. Gracias por estos años.

Por último pero no menos importante, a mi familia, a mis amigos de siempre, a mi pajarillo, y a cualquiera que haya tenido que aguantarme hablar sobre ver conejos y sillas detrás de las esquinas estos últimos meses. Sin vosotros no habría sido posible.

“Rabia, rabia, contra la
luz que se esconde.”
— Dylan Thomas,
Do not go gentle into
that good night.

RESUMEN

La imagen computacional ha abierto nuevas posibilidades al permitir la formación de imágenes a partir de datos capturados por sensores digitales más allá de los sensores de las cámaras clásicas. Se han logrado avances notables en cámaras computacionales ultrarrápidas[1] capaces de registrar el transporte de luz en una escena a lo largo del tiempo, como si de una cámara lenta que pudiera captar la velocidad de la luz se tratara. Así como en métodos de *renderizado*² transitorio de escenas[2], los cuales facilitan la investigación sin necesidad de emplear costosos y complejos sistemas de medición.

Estos avances han propiciado el surgimiento de aplicaciones innovadoras, particularmente en el ámbito de la visualización de objetos ocultos en entornos sin línea de visión (NLOS³). Esta rama de investigación implica el análisis de la luz indirecta que se refleja en objetos visibles para reconstruir, mediante procesos computacionales, la escena oculta. Entre los métodos utilizados para lograr esta reconstrucción se destacan la proyección inversa (back projection) y los métodos basados en la propagación de ondas, como *phasor fields* y *F-K migration*. Estos enfoques permiten manipular las ondas de luz reflejada para reconstruir imágenes de objetos que no se encuentran directamente en la línea de visión del sensor. Este desarrollo tecnológico presenta prometedoras aplicaciones prácticas, desde la prevención de colisiones de vehículos hasta la visualización de áreas de difícil acceso.

Sin embargo, a pesar de los avances significativos en la reconstrucción de escenas sin línea de visión, persisten desafíos computacionales que limitan su aplicación práctica. Este trabajo se centra en abordar uno de estos desafíos: la eficiencia computacional en la reconstrucción tridimensional de escenas ocultas.

El objetivo de esta investigación radica en optimizar la velocidad de reconstrucción 3D mediante la implementación de técnicas de filtrado de bandas estrechas de frecuencias. Se han explorado diversas técnicas para la selección de frecuencias, priorizando las frecuencias que aportan la información más relevante para el proceso de reconstrucción. Esto permite un enfoque selectivo y eficiente al utilizar únicamente las frecuencias más útiles, agilizando así el proceso computacional, con mejoras de orden al orden de magnitud en tiempos de ejecución manteniendo una calidad aceptable. Funcionando más rápido aún usando *datasets* con 4 veces más resolución.

²Renderizado: proceso de crear una imagen o conjunto de imágenes digitalmente.

³NLOS: *Non-Line-of-Sight*. Sin línea de visión.

Índice

1. Introducción y objetivos	1
1.1. Objetivos	4
1.2. Metodología	5
2. Conocimiento previo	7
2.1. Datos de entrada y su captura	7
2.2. Herramientas matemáticas empleadas.	9
2.3. Soluciones al problema de reconstrucción de escenas.	10
2.3.1. Phasor Fields	10
2.3.2. F-K migration	11
2.3.3. Puntos de mejora	13
2.4. Filtrado	13
3. Optimización de los algoritmos <i>Phasor Fields</i> y <i>F-K migration</i>	15
3.1. Phasor Fields	16
3.1.1. Modificaciones	16
3.2. F-K migration	17
3.2.1. Modificaciones	17
4. Resultados	21
4.1. Métricas	21
4.2. Phasor Fields	22
4.2.1. Experimentación	22
4.2.2. Análisis de resultados	24
4.3. F-K migration	25
4.3.1. Experimentación	26
4.3.2. Análisis de resultados	26
5. Conclusiones	31
6. Bibliografía	33

Lista de Figuras	35
Anexos	37
A. Experimentación de parámetros de Narrow-Band F-K migration	39
B. No sobreyectividad de la función de interpolación	41

Capítulo 1

Introducción y objetivos

En el ámbito de la computación visual, la imagen computacional ha representado un hito significativo al permitir la captura y representación de información visual a través de medios distintos a los habituales. A diferencia de la toma de imágenes mediante cámaras convencionales, la imagen computacional no se limita simplemente a registrar escenas en tiempo real, sino que es capaz de reconstruir la geometría de entornos completos a partir de datos capturados y procesados por diferentes tipos de sensores. Esto es posible gracias a la explotación y procesamiento computacional de propiedades físicas del transporte de luz.

Un hito de la imagen computacional es lograr ver lo que a priori, mediante cámaras tradicionales, no es visible. Ejemplo de esto es poder ver el interior del cuerpo humano, agujeros negros en el centro de nuestra galaxia, estructuras subterráneas geológicas o arqueológicas... En esta disciplina, entra la reconstrucción de escenas fuera de la línea de visión de la cámara o sensor que utilizamos para capturar el entorno. A este tipo de técnicas se le conoce como imagen NLOS, por su denominación en inglés non-line-of-sight. Para ello, los algoritmos de NLOS *imaging*¹ se basan en la medición, análisis y procesamiento de la luz indirecta que la escena oculta refleja en objetos visibles por un sensor, utilizando algoritmos y procesos computacionales para reconstruir digitalmente la escena oculta. El estado del arte en reconstrucción de escenas ocultas lo proporcionan las técnicas de imagen NLOS basadas en el tiempo de vuelo de la luz (en inglés Time-of-Flight o ToF). Este tipo de técnicas habitualmente implementan el *setup* que se puede observar en la Figura 1.1, basado en la emisión y medición de la luz mediante un láser y un sensor ultrarrápidos con resolución temporal de picosegundos, capaces de distinguir fotones con caminos ópticos a escala de milímetros. El procedimiento para reconstruir escenas ocultas mediante este tipo de técnicas es habitualmente el siguiente: primero, se emite un pulso de luz a un punto

¹NLOS *imaging*: Conjunto de tecnologías que permiten reconstruir geometrías ocultas sin línea de visión.

conocido de una superficie visible que refleja la luz de manera uniforme (e.g. una pared de yeso, escayola, papel). Esta superficie es conocida como muro repetidor (*relay wall* en inglés). Este muro repetidor refleja la luz recibida contra la escena oculta, que a su vez refleja la luz de vuelta al muro repetidor, de donde es medida en un conjunto de puntos mediante el sensor ultrarrápido. Es entonces cuando se captura esa luz indirecta, con la información que esta porta (tiempo, intensidad, etc.). En concreto, el tiempo de vuelo porta información relevante ya que está relacionado con la distancia entre el relay wall y los distintos puntos de la escena oculta que reflejaron esa iluminación indirecta. Los datos resultantes de este proceso de captura sirven de entrada para multitud de métodos computacionales de imagen NLOS para generar reconstrucciones tridimensionales de geometría oculta.

Aunque los métodos pueden diferir en sus principios físicos y estrategias de procesamiento, todos ellos habitualmente requieren procesar una cantidad muy elevada de datos espacio-temporales de transporte de luz, resultando en algoritmos con un alto coste computacional y de memoria.

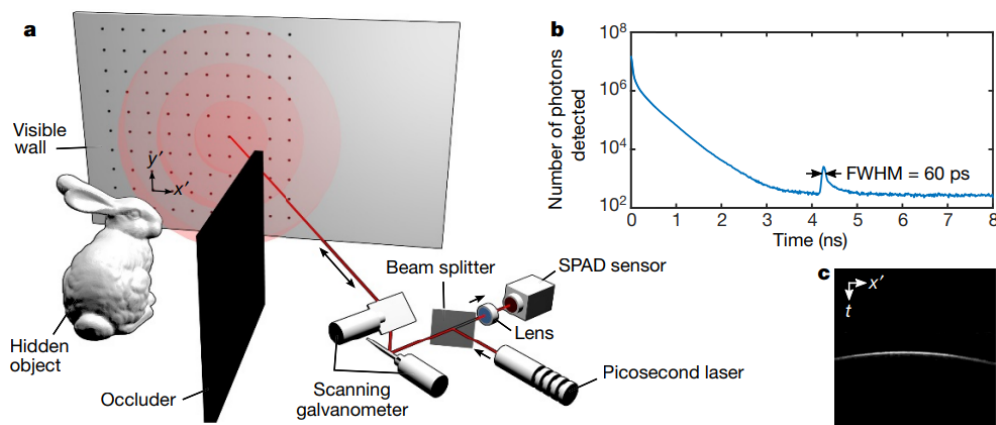


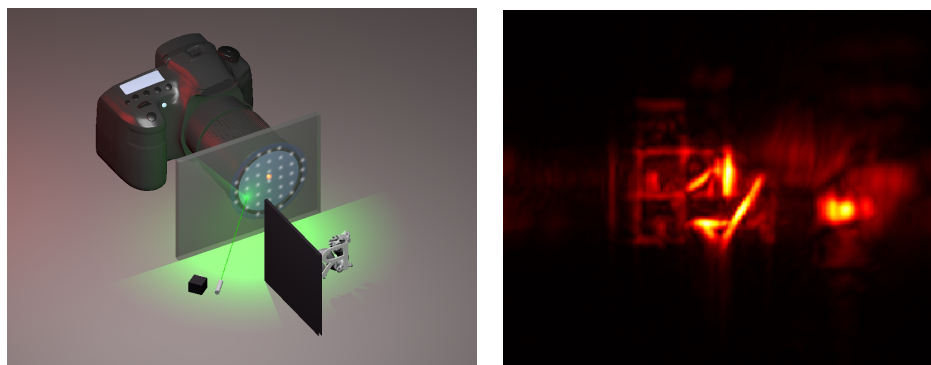
Figura 1.1: (a) *Setup* de NLOS. Al lado izquierdo del ocluser se encuentra la geometría oculta. En frente de esta, se encuentra la pared visible. Al otro lado del ocluser se encuentra el sistema de emisión y captación de luz, con un láser emitiendo el pulso de luz y un sensor SPAD captando el tiempo de vuelo de la luz. (b) Gráfica de la intensidad de la luz captada en función del tiempo transcurrido. (c) Imagen de un frente de onda avanzando (figura de [3]).

Una de las técnicas clásicas para la reconstrucción de escenas sin línea de visión a partir de datos de tiempo de vuelo, consiste en el uso del algoritmo de proyección inversa o *Back-Projection*[4]. Se basa en la triangulación de los posibles puntos de procedencia de la luz captados en la pared visible en base a los datos ToF. No obstante, en NLOS *imaging*, esta técnica genera reconstrucciones poco definidas y es computacionalmente exigente debido a su naturaleza iterativa.

En 2019 surgieron dos métodos basados en principios de propagación de ondas

mediante la transformación e interpretación de los datos como conjuntos de ondas medidas en una superficie visible. Uno de ellos es el llamado método de *Phasor Fields*[5]. Este método crea “lentes computacionales” en la superficie visible a partir de los datos de transporte de luz capturados, simulando en la práctica, crear una cámara en la pared observable como ilustra la Figura 1.2a. Estas “lentes” toman imágenes virtuales enfocadas a distintos planos de la escena, generando una reconstrucción tridimensional de la escena oculta. Esto se logra a través del uso de la integral de difracción de *Raileigh-Sommerfeld*. Esta integral se puede resolver rápidamente con una convolución, en este caso multiplicando en espacio de frecuencias. Esta es una de las claves del éxito de *Phasor Fields* ya que es una operación que abarata el coste computacional con respecto a los métodos iterativos previos, consiguiendo reconstrucciones 3D de escenas complejas en cuestión de segundos. Un ejemplo de reconstrucción se puede ver en la Figura 1.2b.

El otro método que siguió este enfoque basado en propiedades de las ondas es *F-K Migration* [6]. Este artículo propuso emplear *F-K migration*, un algoritmo con base en la sismología que se usa para reconstruir estructuras subterráneas, en la tarea de NLOS *imaging*. *F-K migration* se basa en invertir el transporte de las ondas medidas en el relay wall para inferir la geometría de la escena oculta que las produjo. Todo ello mediante una rápida interpolación en espacio de Fourier para resolver el problema. Permitiendo capturar los datos y reconstruir escenas 3D en segundos, conservando una alta calidad en las reconstrucciones. Sin embargo, es un método muy sensible a datos espurios o ruido, que acaba transformándose en artefactos en las reconstrucciones finales.



(a) Lente virtual.

(b) Reconstrucción de escena oculta.

Figura 1.2: Escena oculta reconstruida y método de lentes virtuales (Figura (a) de [5]).

Sin embargo, pese a todos los avances, uno de los problemas de la tecnología de NLOS *imaging* sigue siendo la complejidad computacional de estos algoritmos. Esto se debe en parte a la gran dimensionalidad de los datos. La gran cantidad de dimensiones se suma al volumen de los datos y a su baja densidad de información, siendo

relevantes solo unos pocos de estos. Es aquí donde entra mi contribución, filtrando los datos y disminuyendo su volumen obteniendo reconstrucciones menos exigentes computacionalmente y de mayor calidad. El filtrado se ha realizado en *Phasor Fields* y *F-K migration* ya que es un filtrado que atañe a datos en espacio de frecuencias, que son los datos con los que trabajan las partes críticas de estos algoritmos. El filtrado pretende reducir la carga computacional, disminuyendo la cantidad de datos a operar sin perder mucha calidad. Esto es posible gracias a que, a diferencia de los datos de tiempo de vuelo de la luz, donde cada elemento posee la misma importancia que el resto, cuando se trabaja con datos en una representación basada en frecuencias, sí existen frecuencias que aportan más al resultado que otras. Esto puede ser controlado mediante diferentes técnicas, como las propuestas en este trabajo.

1.1. Objetivos

El objetivo de este trabajo es modificar los algoritmos de *Phasor Fields* y *F-K migration*, para obtener resultados de calidad equiparable haciendo un uso más eficiente del espacio de frecuencias, y con ello, más eficiente tanto en memoria como en tiempo de ejecución. A partir de ello, comparar los diversos métodos de selección de frecuencias, tanto entre ellos, como con los algoritmos originales usando varios *datasets*.

Tras la modificación de los algoritmos, se espera obtener una reducción de un orden de magnitud en tiempo de ejecución. También se espera necesitar menos cantidad de memoria así como poder ejecutar *datasets* más pesados, de 4 veces más tamaño en el mismo tiempo, permitiendo reconstrucciones más rápidas y de mayor calidad.

Estas modificaciones constarán de varios métodos de selección de subconjuntos de frecuencias en *Phasor Fields*. En *F-K migration*, se aplicará el filtrado del algoritmo original de *Phasor Fields* para concentrar la información. Posteriormente, se usará esta información para reconstruir las escenas de forma más eficiente.

La memoria consta de una estructura dividida en 5 capítulos. El primer capítulo es esta introducción, donde se comenta el contexto en el que se encuentra el trabajo. El segundo capítulo indaga en las diferentes formas de reconstruir escenas ocultas. Explicando el funcionamiento de *Phasor Fields* y *F-K migration*. También se hablará sobre métodos de muestreo (*sampling*). En el tercer capítulo se abordará la metodología seguida, detallando los diferentes métodos de optimización de frecuencias usados. El cuarto capítulo recopila los resultados obtenidos de aplicar las diferentes optimizaciones en los algoritmos. Estas se comparan entre sí y con los algoritmos originales. A su vez se hace una breve descripción de la medida de error empleada. Por último, el quinto capítulo trata las conclusiones obtenidas a lo largo del trabajo.

1.2. Metodología

El proyecto ha comprendido varias fases. Una fase inicial de investigación sobre el estado del arte de técnicas de reconstrucción de escenas. En esta fase se ha implementado de *Back-Projection* para comprender la base del funcionamiento de los algoritmos de reconstrucción de geometría oculta. También se ha experimentado con distintos filtros en espacios de Fourier, investigando cómo aprovecharlos para reducir el espacio de frecuencias. También se ha estudiado el estado del arte de métodos de reconstrucción de escenas mediante espacios de frecuencias, en concreto *Phasor Fields* y *F-K migration*. En la fase de implementación de las mejoras, se ha modificado *Phasor Fields* para poder filtrar las frecuencias a usar. Esto se ha hecho mediante el uso de tres técnicas de selección de frecuencias: una que emplea rangos de frecuencias, otra que elige frecuencias aleatorias mediante muestreo por importancia y una última que modifica el muestreo de importancia para evitar artefactos. Una vez modificado *Phasor Fields* se ha estudiado el funcionamiento de la interpolación de *F-K migration* para su adaptación. Para ello, se ha aplicado filtro de *Phasor Fields* al algoritmo de *F-K migration*, permitiendo un uso más eficiente de las frecuencias. Como último paso de la modificación, se ha mejorado la forma en la que se interpola para hacer uso de todas las frecuencias de interés manteniendo coste en memoria bajo. Por último, se han realizado pruebas y estudio de resultados, empleando tres *datasets* que provienen de las fuentes originales de *Phasor Fields* y *F-K migration*. Todo ello mientras se elaboraba la memoria en LaTeX.

En cuanto a tecnologías, se ha trabajado en MATLAB para las modificaciones. También se ha usado Python y C++ en algunas fases de experimentación. Las pruebas se han llevado a cabo con los *datasets* originales publicados en los artículos, componiendo escenas diferentes.

Capítulo 2

Conocimiento previo

En este capítulo, se va a explicar el funcionamiento de los métodos de *Phasor Fields* y *F-K migration* y sus descripciones matemáticas. Estos métodos serán adaptados posteriormente para permitir un uso más eficiente de los datos. Previo a ello, se va a hacer una explicación de los datos que emplean estos algoritmos y sus métodos de captura, así como una breve introducción a las transformadas de Fourier y a por qué emplear datos en dominio de frecuencias abre puertas a varios métodos de optimización. Finalmente, se detallarán diferentes métodos de filtrado de datos, sus propiedades, limitaciones y qué oportunidades brindan.

2.1. Datos de entrada y su captura

Para reconstruir geometrías ocultas, se pueden emplear varios algoritmos. Los algoritmos de reconstrucción pueden tomar tipos de datos de transporte de luz en diferentes formas. Uno de los datos más comunes son datos de Tiempo de Vuelo de la luz (ToF), es decir, el tiempo que emplea la luz desde que se emite por un láser del *setup* hasta que alcanza el sensor que capta la información, rebotando entre medias por el volumen oculto. Este viaje de la luz se ve ilustrado en la Figura 2.1. El proceso es el siguiente: primero, se emite un pulso de luz hacia el muro repetidor, el cual es un muro difuso que refleja la luz en todas las direcciones. Después, viaja por el volumen oculto, rebotando en la geometría que queremos reconstruir hasta volver al muro repetidor de nuevo. Por último, llega al sensor de nuestro *setup* donde se almacena la información que la luz porta. Estos datos pueden ser tomados mediante femtofotografía [1] o bien simulados mediante aplicaciones como MITSUBA 2 [2]. Estos datos suelen constar de tres dimensiones principales, distribuidas en dos dimensiones espaciales x_c e y_c , que representan los puntos de captura en el muro repetidor y una temporal, t . El resultado podría asemejarse a crear un vídeo en cámara lenta de la luz indirecta que llega al muro repetidor.

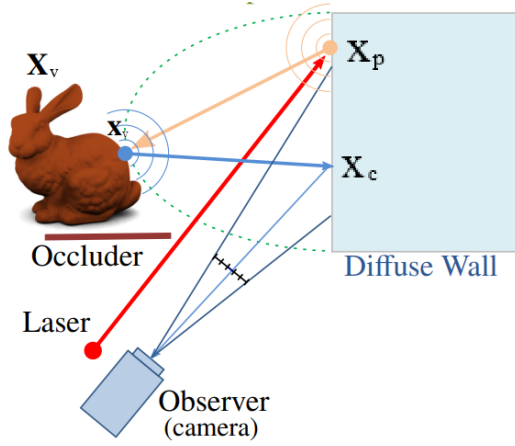


Figura 2.1: Un pulso de luz es emitido por el láser hacia un punto s visible. El pulso rebota en un punto desconocido x . Finalmente vuelve a un punto p de la pared visible, donde posteriormente es captado por el observador (figura de [7]).

Así pues, las dimensiones espaciales x_c e y_c representan los diferentes puntos de la pared visible donde se hacen las mediciones y, la dimensión t , a qué momento después de emitir la señal corresponde la medición. Adicionalmente, existe la opción de apuntar el láser hacia diferentes puntos \mathbf{x}_p de la superficie visible, aumentando los puntos de vista de las triangulaciones.

Partiendo de esta forma de captura de los datos, la ecuación para modelar los datos de tiempo de vuelo (ToF) puede tomar la forma de una integral sobre la intensidad de la luz en un instante t y la geometría de la escena oculta. La intensidad de la señal recibida $H(\mathbf{x}_p \rightarrow \mathbf{x}_c, t)$ puede ser expresada como:

$$H(\mathbf{x}_p \rightarrow \mathbf{x}_c, t) = \int_{\mathbf{x}_v} \phi(\mathbf{x}_v, t) \delta\left(t - \frac{\|\mathbf{x}_0 - \mathbf{x}_p\| + \|\mathbf{x}_p - \mathbf{x}_v\| + \|\mathbf{x}_v - \mathbf{x}_c\| + \|\mathbf{x}_c - \mathbf{x}_0\|}{c}\right) d\mathbf{x}_v \quad (2.1)$$

donde \mathbf{x}_0 , \mathbf{x}_p , \mathbf{x}_v y \mathbf{x}_c son, respectivamente: las coordenadas espaciales del sensor/láser, el *relay wall* en los puntos donde incide el láser, los distintos puntos de la geometría oculta y el *relay wall* en los puntos donde se captura la luz. $\phi(\mathbf{x}_v, t)$ es la intensidad de la luz en un instante t habiendo sufrido atenuación causada por rebotar en un punto de la geometría oculta \mathbf{x}_v . Por último c representa la velocidad de la luz, para convertir la distancia a tiempo. En este trabajo, solamente se va a emplear un punto \mathbf{x}_p por cada \mathbf{x}_c . Esta es una práctica común que disminuye en una dimensión los datos. Por lo que de ahora en adelante la variable H se expresará como $H(\mathbf{x}_c, t)$. A su vez, \mathbf{x}_p quedará definido como constante para cada \mathbf{x}_c .

Una vez capturados los datos de tiempo de vuelo H , los algoritmos trabajan empleando información relativa a qué intensidad llegó en qué momento de qué parte del *relay wall* o muro repetidor para recrear la escena oculta.

2.2. Herramientas matemáticas empleadas.

Los métodos que reconstruyen escenas mediante datos de tiempo de vuelo de la luz (ToF) destacan por ser computacionalmente costosos debido a la gran dimensionalidad de los datos, así como por la naturaleza iterativa que requieren. Es por ello, que métodos que emplean datos en espacio de frecuencias tienden a obtener mejores resultados en menos tiempo. Estos métodos componen una rama importante del campo de *non-line-of-sight imaging*. Estos son métodos de propagación de ondas como *Phasor Fields* o *F-K migration*. Estos métodos emplean los datos en forma de ondas de luz. Que puede ser representado de varias formas.

Para entender la transformada de Fourier se va a explicar el concepto de fasor. Un fasor es un número complejo oscilatorio usado en muchos campos para representar ondas (como la luz). Un fasor se puede expresar mediante la siguiente expresión:

$$e^{i2\pi\omega t} \quad (2.2)$$

Donde i es la unidad imaginaria, ω es la frecuencia de oscilación y t , el tiempo transcurrido. La ventaja de los fasores radica en que son capaces de describir la luz como un número complejo oscilante de forma simple y compacta.

En *Phasor Fields* y *F-K migration*, las dimensiones de los datos en las partes cruciales de cómputo no incluyen una dimensión temporal, como ocurre en *Back-Projection*, sino que esta dimensión es remplazada por una representación en espacio de frecuencias. Esta reinterpretación de los datos se lleva a cabo mediante transformadas de Fourier, que pueden representarse de la siguiente forma:

$$\chi(\omega) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi\omega t} dt \quad (2.3)$$

La transformada de Fourier es una herramienta que permite conocer cuánta energía porta una señal $x(t)$ dependiendo de una frecuencia de oscilación ω . Es decir, si se suman dos ondas con frecuencias $2Hz$ y $5Hz$ en una señal $x(t)$. La transformada resultante $\chi(\omega)$ tendría picos de valor para $\omega = 2$ y $\omega = 5$, dependiendo de la amplitud de las señales originales. Esta representación puede usarse para transformar la dimensión temporal de los datos ToF al espacio de frecuencias. Esto no solo suele reducir el tamaño en memoria de los datos, sino que abre las puertas a una nueva forma de cómputo de *Back-Projection: Wave-Projection*. Este puede implementarse con la solución convolucional de la RSD¹[8], o interpolando frecuencias[6]. Estas implementaciones reducen el tiempo de reconstrucción de la escena a segundos.

¹RSD: Rayleigh-Sommerfield Diffraction integral. Integral de difracción de Rayleigh-Sommerfield.

2.3. Soluciones al problema de reconstrucción de escenas.

En esta sección, se va a explicar en detalle el funcionamiento de dos métodos de reconstrucción que hacen uso de espacios de frecuencias: *Phasor Fields* y *F-K migration*. En el trabajo se ha modificado la versión disponible del código de los artículos *Phasor field diffraction based reconstruction for fast non-line-of-sight imaging systems* [8] y *Wave-based non-line-of-sight imaging using fast f-k migration* [6].

Para mejorar la precisión en la reconstrucción de las geometrías ocultas, se pueden emplear técnicas que utilicen la transformada de Fourier para procesar las señales de tiempo de vuelo. Esto implica convertir las señales de dominio temporal a dominio de frecuencia. Así, podemos definir la ecuación para la transformación en el dominio de frecuencia como:

$$\hat{H}(\mathbf{x}_c, \Omega) = \mathcal{F}\{H(\mathbf{x}_c, t)\} \quad (2.4)$$

donde $\hat{H}(\mathbf{x}_c, \Omega)$ representa la señal en el dominio de frecuencia con \mathcal{F} denotando la transformada de Fourier y Ω la dimensión frecuencial.

A partir de aquí, *Phasor Fields* y *F-K migration* toman caminos separados para invertir la ecuación 2.1 y obtener el volumen V mediante H .

2.3.1. Phasor Fields

Por un lado, *Phasor Fields*[5] es una formulación propuesta para implementar diferentes modalidades dentro del *NLOS imaging*. Principalmente se basa en interpretar la señal recibida H en espacio de frecuencias como un campo de fasores que representa uno o varios frentes de onda virtual. Estos frentes de onda virtuales se pueden utilizar para crear “cámaras virtuales” en el muro repetidor (figura 2.2). Computacionalmente, esto se consigue mediante una convolución en la dimensión temporal entre la señal recibida H y un parámetro fasorial² $\mathcal{P}(\mathbf{x}_p, t)$, que simula una difracción emergente del punto \mathbf{x}_p , es decir, simula una onda virtual con origen en el punto donde incide el láser:

$$H_{\mathcal{P}}(\mathbf{x}_c, t) = \mathcal{P}(\mathbf{x}_p, t) *_t H(\mathbf{x}_p \rightarrow \mathbf{x}_c, t) \quad (2.5)$$

Así pues, $H_{\mathcal{P}}(\mathbf{x}_c, t)$ representa la señal difractada recibida, de la cual se pueden extraer imágenes de la escena oculta en diferentes planos de profundidad mediante el operador de *wave propagation*: $\Phi(\cdot)$. La versión en dominio de Fourier $\Phi_{\mathcal{F}}(\cdot)$ de este operador es conocida como la Integral de Rayleigh-Sommerfeld (RSD)

²Fasorial: relativo o que emplea fasores.

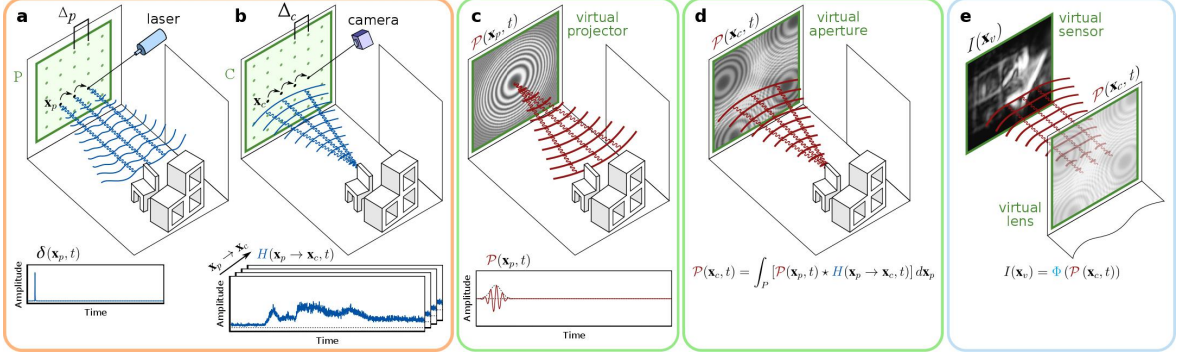


Figura 2.2: Pasos de Phasor Fields. (a) Se emite un pulso de luz en un punto de la superficie visible, posteriormente se puede hacer con más. (b) Se captura la intensidad de la luz en cada momento en la superficie visible. (c) Fuente de luz virtual. La onda virtual de campos de fasores se emite a la escena oculta. (d) La escena responde a la iluminación proyectando en la pared visible un campo de fasores. (e) La escena es finalmente reconstruida usando wave projection. (figura de [5]).

La integral de difracción de Rayleigh-Sommerfeld (RSD) es una ecuación matemática que describe la propagación de ondas en un medio homogéneo. Esta integral se puede emplear para, a partir de los campos de fasores que emplea *Phasor Fields*, calcular de donde procede la luz. La ventaja de este método es su posible resolución mediante transformada de Fourier rápida. Concretamente con el método de resolución propuesto en el artículo *Fast-Fourier-transform based numerical integration method for the Rayleigh-Sommerfeld diffraction formula* [9]. Este método logra reducir significativamente el tiempo y memoria de cómputo al calcularse solamente mediante una convolución. Gracias a esto, la ecuación de reconstrucción puede expresarse como:

$$P(\mathbf{x}_c, \Omega) = \mathcal{F}\{H_{\mathcal{P}}(\mathbf{x}_c, t)\} \quad (2.6)$$

$$I(\mathbf{x}_v, t) = \mathcal{F}^{-1}\{\Phi_{\mathcal{F}}(P(\mathbf{x}_c, \Omega))\} \quad (2.7)$$

donde $I(\mathbf{x}_v, t)$ es la imagen reconstruida sobre las coordenadas \mathbf{x}_v de la escena y $P(\mathbf{x}_c, \Omega)$ representa la transformada de Fourier de la señal en la superficie de *relay wall*. La imagen bidimensional posteriormente se computa para los diferentes planos de imagen obteniendo reconstrucciones por capas como se puede apreciar en la Figura 2.3.

2.3.2. F-K migration

F-K migration[6], inspirado en técnicas de sismología, adapta un método en el dominio de frecuencias para resolver el problema de invertir el volumen oculto. El algoritmo interpreta los datos confocales en dominio de frecuencias como un frente de ondas producido por la escena oculta en la superficie visible. El método de

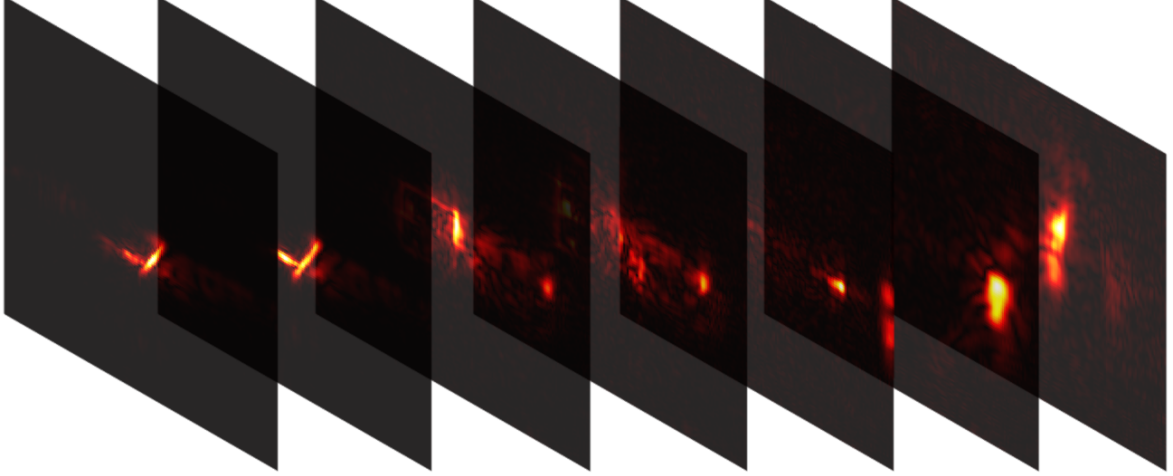


Figura 2.3: Capas de profundidad en una reconstrucción de escena NLOS.

reconstrucción de fk migration se basa en invertir ese transporte de ondas aprovechando propiedades específicas que proporciona la configuración confocal y la asunción de tercer rebote, es decir, que el láser apunte al mismo sitio que la cámara y que la luz captada es exclusivamente luz de tercer rebote (primer rebote en el muro repetidor, segundo en la escena oculta y tercero en el muro repetidor de nuevo). Estas propiedades, hacen que la ecuación de transporte de ondas que modela la H confocal, asumiendo iluminación de tercer rebote, sea invertible mediante una interpolación en el dominio de frecuencias de los datos capturados H . Esta interpolación se hace sobre todo el dominio de frecuencias de \hat{H} , pese a que no todas las frecuencias proporcionan información geométrica de la escena. La ventaja de este método radica su bajo coste computacional en comparación con la alta resolución de sus reconstrucciones.

Al realizar la interpolación, se están eligiendo ciertas coordenadas a lo largo de las dimensiones $\mathbf{x}_c = \{x_c, y_c\}$ y Ω . Mientras que las dos coordenadas espaciales se mantienen fijas, Ω sí cambia. La función de interpolación \mathcal{I} , que da como resultado los datos interpolados en dominio de Fourier \hat{H}' , se puede representar de la siguiente forma:

$$\hat{H}'(\mathbf{x}_c, \Omega) = \mathcal{I}(\hat{H}(\mathbf{x}_c, \Omega)) = \hat{H}(\mathbf{x}_c, \Omega') \quad (2.8)$$

$$\Omega'(\mathbf{x}_c) = f(\mathbf{x}_c, \Omega) = \sqrt{\rho * (x_c^2 + y_c^2) + \Omega^2} \quad (2.9)$$

Donde ρ es un factor de escala. Como se puede apreciar en esta implementación, los valores frecuenciales para cada Ω pasan a ser los que se encontraban en la posición $f(\mathbf{x}_c, \Omega) = \sqrt{\rho * (x_c^2 + y_c^2) + \Omega^2}$. Dependiendo ahora de las dimensiones espaciales. Mientras que las coordenadas espaciales se mantienen intactas.

Cabe mencionar que tanto la función de interpolación \mathcal{I} como la función de mapeado

de las frecuencias f , son no sobreyectivas. Esto se encuentra detallado en el Anexo B y es importante a tener en cuenta de cara a la hora de reducir frecuencias, como se pretende en los siguientes apartados, ya que hay frecuencias útiles que pueden no ser usadas.

Tras definirse la función de interpolación, la fórmula general de reconstrucción con *F-K Migration* se podría expresar entonces como:

$$I(\mathbf{x}_c, t) = \mathcal{F}^{-1}\{\mathcal{I}(\hat{H}(\mathbf{x}_c, \Omega))\} \quad (2.10)$$

2.3.3. Puntos de mejora

Como se ha explicado, estos métodos hacen uso del espacio de frecuencias. Para ello, aplican la transformada de Fourier a los datos de entrada. Después, estos datos en dominio de frecuencias se procesan para obtener las reconstrucciones de las geometrías ocultas. Sin embargo, no todos los datos en dominio de frecuencias son igual de útiles. De hecho, al igual que ocurre con la ecualización de sonido, pueden existir componentes de alta frecuencia que produzcan luego ruido en los resultados. Lo mismo ocurre con componentes de baja frecuencia. Al margen del ruido, no todas las frecuencias representan la misma cantidad de información útil sobre la geometría de la escena. Es por ello que un método inteligente de filtrado de frecuencias podría llevar a reconstrucciones más eficientes. Todo mediante seleccionar subconjuntos de datos representativos y deshaciéndose de los datos que aportan poca información, o ruido directamente.

2.4. Filtrado

En *NLOS imaging*, así como en muchos campos de la informática, a veces la solución a muchos problemas pasa por preprocesar los datos de entrada, aplicándoles algún tipo de filtro. En *Back projection* filtrado (algoritmo usado en tomografías), se suele emplear el filtro LoG (Laplaciano de una Gaussiana) para conseguir resultados de mayor calidad que sin aplicar el filtro. Para ello, simplemente se aplica una operación de convolución a los datos con el filtro LoG. Este filtro se puede observar en la Figura 2.4a junto con una ondícula (*wavelet*), que es el tipo de filtro que se emplea en *Phasor Fields*. Cabe destacar la diferencia entre este filtro y el algoritmo de *Phasor Fields* en sí. La ondícula, es solo un filtro que se emplea para preprocesar los datos en ese algoritmo. Es de ahí que toma el nombre del “filtro de *Phasor Fields*”.

Una de las principales diferencias entre estos dos filtros reside en cómo se comportan en dominio de frecuencias. Si realizamos la transformada de Fourier de estos filtros

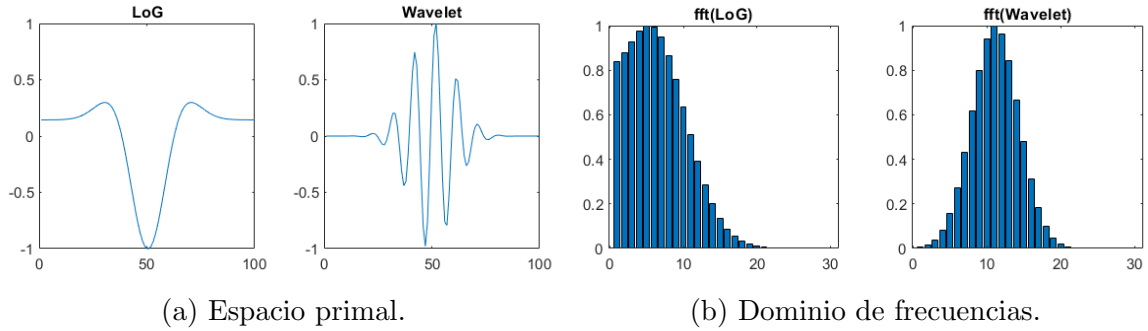


Figura 2.4: Filtros en espacio primal y de frecuencias.

obtendremos unas gráficas como las de la Figura 2.4b, donde el LoG siempre posee componentes de baja frecuencia mientras que la ondícula es una Gaussiana, cuyo centro depende de la frecuencia de la onda portadora, y cuya desviación es inversamente proporcional a la cantidad de ciclos que la onda completa.

Los componentes en dominio de frecuencias de los filtros no parece de gran relevancia hasta que se busca un tipo de resolución. Si el filtro posee componentes de baja frecuencia, los datos resultantes carecerán de resolución fina, obteniendo contornos difuminados. Por otro lado, si el filtro posee en gran medida componentes de alta frecuencia, los datos resultantes tendrán ruido fino, y las grandes estructuras perderán intensidad frente a los contornos, que serán resaltados. Otro punto importante es que, aunque se desee implementar un filtro LoG con componentes de alta frecuencia, **siempre** poseería bastante intensidad en los componentes de baja frecuencia. Esto genera resultados más difuminados y prohíbe prescindir de las frecuencias bajas dada su intensidad.

En resumen, en *NLOS imaging*, si queremos aplicar un filtro a los datos que nos permita seleccionar subconjuntos de frecuencias reducidos pero representativos, la mejor opción entre estos dos filtros son lás ondículas presentadas en *Phasor Fields*. A través del filtrado mediante ondículas, se puede eliminar ruido de baja frecuencia, pudiendo establecer una frecuencia central y una cantidad de frecuencias en torno a esta. Otorgando flexibilidad y delimitando rangos de frecuencias de forma clara, como por ejemplo, seleccionando las frecuencias comprendidas en una cantidad de σ entorno al centro.

Capítulo 3

Optimización de los algoritmos *Phasor Fields* y *F-K migration*

En este capítulo se describen las modificaciones implementadas en los algoritmos de *Phasor Fields* y *F-K migration*. Como se explicó en el capítulo anterior, estos métodos interpretan los datos como frentes de ondas medidos en la superficie visible o muro repetidor (*relay wall*). Este enfoque en el dominio de frecuencias permite reducir significativamente el coste computacional. Sin embargo, los algoritmos originalmente aplican operaciones a todas las frecuencias, independientemente de su relevancia en la representación de geometrías en el espacio primal¹. Aquí es donde entran en juego nuestras optimizaciones, seleccionando de forma eficiente las componentes frecuenciales más útiles.

En *Phasor Fields* se han implementado tres métodos de filtrado de frecuencias: uno por rangos, otro mediante muestreo por importancia, y un tercero con muestreo por importancia suavizado. En el caso del algoritmo *F-K migration*, se ha reducido el espacio frecuencial tras aplicar el filtro de *Phasor Fields*, eliminando frecuencias innecesarias. Para simplificar la notación, utilizaremos la variable $P(\mathbf{x}_c, \Omega)$, que se describe en la ecuación 2.6, y que representa la transformada de Fourier de los datos de tiempo de vuelo filtrados mediante el filtro de *Phasor Fields*.

La clave para optimizar ambos algoritmos es que, al aplicar los distintos filtrados propuestos a los datos de entrada, estos solo conservan valores diferentes de cero en un rango específico en el espacio de frecuencias. Esto permite que tanto las operaciones de propagación de *Phasor Fields* como las de interpolación de *F-K migration* se realicen únicamente sobre el subconjunto de frecuencias relevantes, permitiendo reducir así el coste computacional.

¹Espacio primal: el espacio resultante de la transformada inversa de Fourier en el dominio frecuencial.

3.1. Phasor Fields

Phasor Fields trabaja con datos en forma de frecuencias a la hora de realizar la RSD, el punto crítico del algoritmo. Esto puede servir para experimentar con la reducción de frecuencias y sus efectos. Este algoritmo ya aplica un filtro de ondícula a los datos, por lo que el objetivo de estas modificaciones es ver cuánto más se puede reducir el espacio de frecuencias, mediante otras técnicas. Todo manteniendo una calidad aceptable en los resultados.

3.1.1. Modificaciones

Para optimizar el algoritmo de *Phasor Fields*, hemos diseñado, implementado y analizado tres modificaciones que reducen el tiempo de cálculo procesando únicamente un subconjunto de las frecuencias distintas a cero que resultan tras aplicar el filtrado a los datos de entrada. A continuación, se detallan los tres métodos que se han implementado para la reducción de frecuencias.

Método de rangos

El método de rangos consiste en seleccionar intervalos específicos de frecuencias en el dominio de Fourier y operar únicamente con los datos correspondientes a esos rangos. Esto permite enfocar los cálculos en determinadas bandas de frecuencia, las cuales pueden ser de interés dependiendo de la aplicación. Por ejemplo, se puede optar por eliminar las frecuencias bajas para centrarse en las componentes de alta frecuencia de la señal, o viceversa. A parte de por la aplicación, la selección de los rangos viene dada por las propiedades del filtro original aplicado a los datos.

Muestreo por importancia

En este método, dada una distribución de frecuencias, se seleccionan aleatoriamente frecuencias dentro de esa distribución para ser utilizadas en el procesamiento de la señal. El muestreo por importancia se enfoca en aquellas frecuencias que tienen una mayor relevancia o que contribuyen de manera significativa a la representación de la señal original. Para ello, se selecciona un subconjunto de frecuencias siguiendo una distribución Gaussiana. Esta distribución depende de las propiedades del filtro de ondícula aplicado previamente. Finalmente, se usan las frecuencias seleccionadas aleatoriamente en el cálculo de la escena oculta.

Muestreo por importancia suavizado

El método de muestreo por importancia suavizado surge como una mejora al método previo. Después de implementar el muestreo por importancia, se observó que la selección aleatoria de frecuencias puede causar artefactos no deseados en la señal reconstruida, debido a la aparición de deltas en las frecuencias. Estos deltas se transforman en ondas sinusoidales al realizar la transformada inversa de Fourier. Para mitigar este problema, el enfoque se ajusta de manera que, además de seleccionar las frecuencias más relevantes, se incluyen también las frecuencias adyacentes. Estas frecuencias colindantes se eligen siguiendo una distribución Gaussiana, lo que suaviza la transición entre las frecuencias seleccionadas y reduce la aparición de artefactos.

En la figura 3.1 se pueden observar los diferentes efectos de estos métodos. Es de recalcar que las frecuencias cuyo valor es cero no se incluyen luego en los cálculos, permitiendo ahorrar tiempo y memoria.

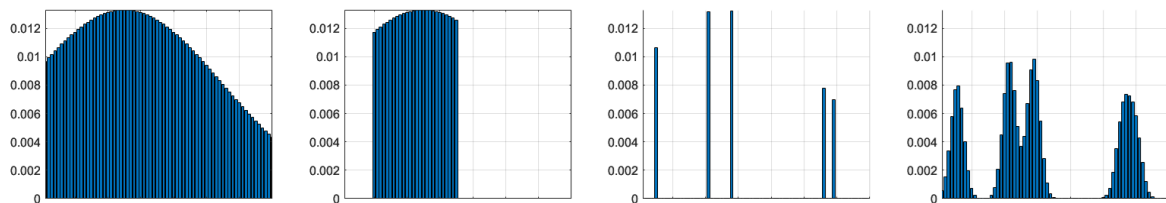


Figura 3.1: Diferentes formas de seleccionar frecuencias. De izquierda a derecha: potencia de las frecuencias originalmente (tiene forma de Gaussiana ya que se aplicó el filtro de *Phasor Fields* previamente); potencia de las frecuencias al aplicar método de rangos; frecuencias al aplicar el método de muestreo por importancia; las mismas frecuencias usadas como medias en curvas Gaussianas (muestreo por importancia suavizado).

3.2. F-K migration

F-K migration transforma los datos ToF a dominio de frecuencias. Luego realiza una interpolación sobre el dominio de frecuencias evaluándolas en un nuevo conjunto de coordenadas de frecuencias. La interpolación se hace siguiendo las ecuaciones 2.8 y 2.9. Es justo antes de la interpolación cuando se puede reducir la cantidad de frecuencias a operar, simplificando el cómputo.

3.2.1. Modificaciones

Para realizar la interpolación en el método original, se emplea todo el dominio de frecuencias, con miles de elementos. En la implementación ofrecida en el artículo

original[6], las dimensiones espaciales del *dataset* son 512×512 , sin embargo, el algoritmo no es capaz de albergar tal cantidad de datos en ordenadores convencionales, requiriendo más de 12GB de memoria de GPU. Además, la interpolación (la cual es el cuello de botella de este algoritmo) aumenta su tiempo de cómputo de forma cuadrática con la resolución. Produciendo reconstrucciones lentas para altas resoluciones. Las modificaciones se centran en la reducción del espacio de frecuencias de forma eficiente.

Phasor fields

Antes de reducir el espacio sobre el que vamos a interpolar, hay que seleccionar la información útil. Es por ello que se propone usar el filtro de *Phasor Fields* para esta tarea. Este filtro, al ser una ondícula, el comportamiento que posee en el espacio de frecuencias es la aplicación de una curva Gaussiana. Por ello, convolucionar este filtro con los datos mediante una multiplicación en espacio de frecuencias logrará que toda la información útil se disponga en forma de campana, con centro igual a la frecuencia *carrier* de la ondícula y desviación inversamente proporcional a la cantidad de oscilaciones de esta en el filtro. Es decir, mediante la aplicación de este filtro, ahora podemos elegir qué rangos de frecuencias queremos usar y cuales desechar.

En resumen, aplicar el filtro de *Phasor Fields* a los datos en *F-K migration* antes de la interpolación modifica la ecuación 2.10. Ahora, habiendo aplicado el filtro, la ecuación que define la imagen final quedaría así:

$$I(\mathbf{x}_c, t) = \mathcal{F}^{-1}\{\mathcal{I}(P(\mathbf{x}_c, \Omega))\} \quad (3.1)$$

Después de aplicar el filtro de *Phasor Fields*, el espacio donde los valores son distintos de 0 queda restringido a los rangos fijados. Podemos ahora utilizar esta información para computar la interpolación **única y exclusivamente** en ese espacio.

Rangos

A la hora de la interpolación surge un problema con la dimensión frecuencial Ω , no así con las espaciales x_c dado que estas se mantienen fijas. Si bien es fácil calcular los valores de Ω en los que $P(\mathbf{x}_c, \Omega) \neq 0$. Este subconjunto de Ω no sirve para realizar la interpolación **dada la naturaleza discreta de los datos**. Ni como coordenada origen ni como destino.

La deformación que sufre el espacio de frecuencias es bastante evidente en la Figura 3.2. Esta deformación se explica con la ecuación de interpolación 2.9.

Tras aplicar el filtro de *Phasor fields*, el espacio de frecuencias queda reducido a una banda estrecha. Esta banda está comprendida en un rango $[\Omega_{min}, \Omega_{max}]$, tomando una

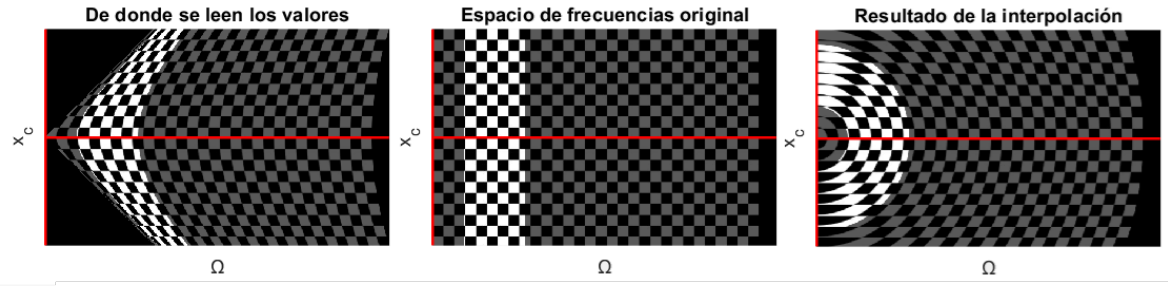


Figura 3.2: Efectos de la interpolación. En el centro está el espacio de frecuencias original representado como un ajedrezado para mejor visualización. En este ajedrezado, se muestran resaltados los rangos de frecuencias distintos de 0. Es decir, las frecuencias dentro del rango que crea el filtro de *Phasor Fields*. A la derecha, se puede ver el efecto de aplicar la interpolación al espacio de frecuencias. Por último, a la izquierda se ve de dónde ha leído el valor cada punto de la cuadrícula. Nótese que esto a su vez sería el efecto de aplicar la función inversa de interpolación.

distancia de 3σ a ambos lados desde el centro de la banda (ya que el filtro aplica una campana Gaussiana de media y desviación conocida). Sin embargo, estos rangos no nos sirven para seleccionar las frecuencias a interpolar, ya que habría valores útiles, es decir, valores distintos de 0, que no serían evaluados en la interpolación. A su vez, al buscar recortar los datos solo a esa franja, se intentaría leer en posiciones, que al estar fuera de la franja, no existen en memoria. Es decir, esos rangos delimitan el área de donde leer los datos, no donde almacenarlos. Si se trabajase en un espacio continuo el problema se tornaría tan sencillo como invertir la función de interpolación. Sin embargo, los datos poseen posiciones discretas de memoria por lo que esta tarea se ve dificultada. Para evitar este problema, el rango de frecuencias $\Omega'_{\alpha\beta}$ donde leer los datos debe acotarse como:

$$\lfloor \Omega_{min} \rfloor \leq \Omega'_{\alpha\beta} \leq \lceil f(\mathbf{X}_{max}, \Omega_{max}) \rceil \quad (3.2)$$

Donde f es la función de interpolación, mostrada en la ecuación 2.9. De forma similar, las coordenadas Ω donde almacenar, serán ahora $\Omega = f^{-1}(\Omega')$. Para ello hay que discretizar la ecuación 2.8, evitando el problema de la diferente dimensionalidad que aparece al no ser lineal la función de interpolación. Por ello, el rango de frecuencias $\Omega_{\alpha\beta}$ donde almacenar los datos estará delimitado en:

$$\lfloor f^{-1}(\mathbf{x}_{max}, \Omega_{min}) \rfloor \leq \Omega_{\alpha\beta} \leq \lceil \Omega_{max} \rceil \quad (3.3)$$

Una vez obtenidos unos rangos seguros que garantizan usar todos los datos útiles y evitar errores, esto se puede usar para mejorar la eficiencia. Para ello, basta con reducir los datos originales P al rango Ω' y almacenar los resultados en un espacio de memoria

que represente solamente el rango de frecuencias Ω .

Índices

En el método anterior, se escogen todos los valores de \mathbf{x}_c para cada frecuencia comprendida en los rangos. Esto nos asegura interpolar en todas las posiciones donde hay valor en P . Sin embargo, no nos garantiza que todas las posiciones sean distinto de 0. Esto se debe a la naturaleza curva de esta interpolación frente a los rangos planos. Sin embargo, es un buen punto de partida para hacer un método basado en índices.

En esta implementación se recortarán los datos de igual manera, siguiendo los rangos Ω y Ω' anteriormente definidos. Sin embargo, en lugar de evaluar la interpolación en Ω' , se va a interpolar solamente en los valores distintos de 0. De esta forma, los subconjuntos \mathcal{S} y \mathcal{S}' donde, respectivamente, almacenar y leer datos serían:

$$\mathcal{S}' = \text{find}(P \neq 0) \quad (3.4)$$

$$\mathcal{S} = f^{-1}(\mathcal{S}') \quad (3.5)$$

Es decir, evaluaremos los valores donde P' sea distinta de 0. De esta forma, ahorramos frecuencias que, podían poseer valores cercanos o iguales a 0 pese a estar en los rangos.

Capítulo 4

Resultados

En secciones anteriores se ha explicado el funcionamiento de los algoritmos de *Phasor Fields* y *F-K migration*. Estos métodos trabajan con datos en dominio de frecuencias en las partes más críticas del procesamiento de datos. Es ahí donde se han implementado técnicas de filtrado de datos para poder disminuir el coste computacional del procesamiento. Esta reducción en el coste computacional no solo se ve reflejada en una reducción del tiempo medio de ejecución, sino que permite emplear *datasets* con mayor resolución que antes no se podían procesar en la mayoría de ordenadores debido a requerimientos de memoria.

En esta sección, se van a explorar los resultados. Se va a experimentar con los parámetros, comparando resultados también entre los distintos métodos y el estado del arte actual. Primero, se van a exponer los resultados de las modificaciones en el código de *Phasor Fields*. Estas modificaciones comprenden un método de selección de rangos de frecuencia, otro de muestreo por importancia y un último método de muestreo por importancia, esta vez suavizando las selecciones con una distribución Gaussiana en frecuencias. Posteriormente, se mostrarán los resultados de aplicar el filtro de *Phasor Fields* a *F-K migration*. Este filtrado permite recortar la cantidad de frecuencias sustancialmente.

4.1. Métricas

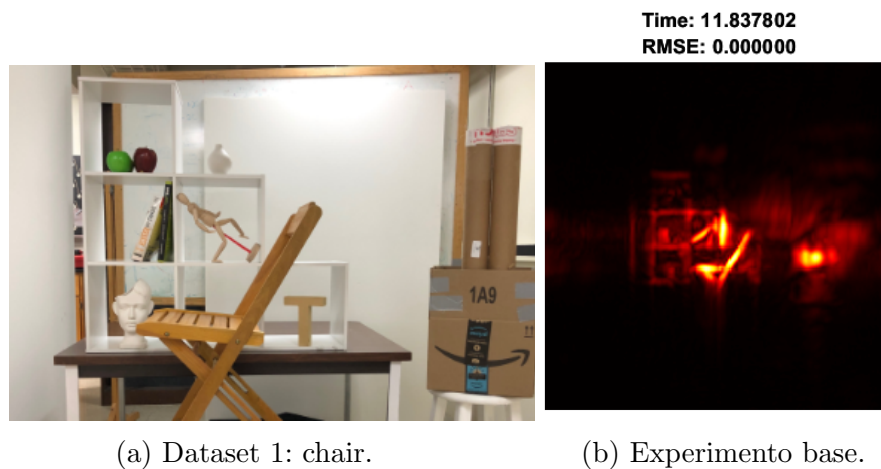
Para poder comparar entre resultados, es decir, las reconstrucciones de los diferentes métodos, se van a usar una serie de métricas tanto cuantitativas como cualitativas. Como métricas cuantitativas se van a emplear tanto el número de frecuencias a interpolar (solo en *F-K migration*) como el tiempo de cómputo de la interpolación, así como la resolución final de los datos. A su vez, se va a evaluar cualitativamente la calidad de los resultados, poniendo énfasis en la tolerancia al ruido tanto de alta frecuencia como de baja frecuencia. Sin embargo, en *Phasor Fields* se ha implementado

una métrica de calidad cuantitativa que mide el $RMSE^1$ con respecto a emplear todas las frecuencias, es decir, con respecto al método original. Esta métrica es una medida de cuánto empeoran las reconstrucciones conforme se emplean menos frecuencias que en el método original. Cabe resaltar que no es una métrica de error con respecto a la verdad base (*ground truth*), sino que trata al método original como el mejor resultado posible dadas las frecuencias que hay.

4.2. Phasor Fields

En esta sección, se van a mostrar los resultados de las modificaciones en el código original de *Phasor Fields*. Como comparación, se van a usar los resultados de emplear todas las frecuencias.

Los *datasets* empleados son escenas NLOS estándares. En concreto, para este algoritmo se ha empleado el *dataset* expuesto en la Figura 4.1a. Esta escena representa una habitación con una silla, una estantería y cajas. Cabe destacar que se trata de datos capturados, es decir, no sintéticos. Esto suele conllevar ruido asociado.



(a) Dataset 1: chair.

(b) Experimento base.

Figura 4.1: Dataset usado con Phasor Fields.

4.2.1. Experimentación

Se han llevado a cabo varios experimentos para evaluar la eficacia de las modificaciones en la reconstrucción de las escenas. Para ello, cabe mencionar que el uso de todas las frecuencias emplea 10,6s de media, obteniendo la reconstrucción de la figura 4.1b. Se han seguido unas pautas oportunas para que todos los experimentos cuenten con igualdad de condiciones con respecto a las capacidades de cómputo.

¹RMSE: del inglés *root mean square error*. Raíz del error cuadrático medio.

En esta sección se van a explicar las tres técnicas de reducción de frecuencias propuestas para *Phasor Fields*. Estas modificaciones presentan un compromiso entre tiempo de ejecución y calidad. Finalmente, se compararán los tres métodos propuestos para ver cuál reduce frecuencias de forma más eficiente.

Método de rangos

Como se explicó anteriormente, este método se basa en restringir las frecuencias a rangos estrechos. Esto permite que, al emplear menos frecuencias, el tiempo de cómputo se reduzca sustancialmente. De igual manera, la calidad disminuye ya que se pierde información. Para estos experimentos, se han elegido rangos en base al filtro original aplicado a los datos. Como el filtro produce una curva Gaussiana en espacio de frecuencias, se ha procedido a establecer los rangos en torno a la media de esta. En cuanto al grosor de los rangos, se ha experimentado siguiendo diferentes grosores y viendo cómo afectan a los resultados.

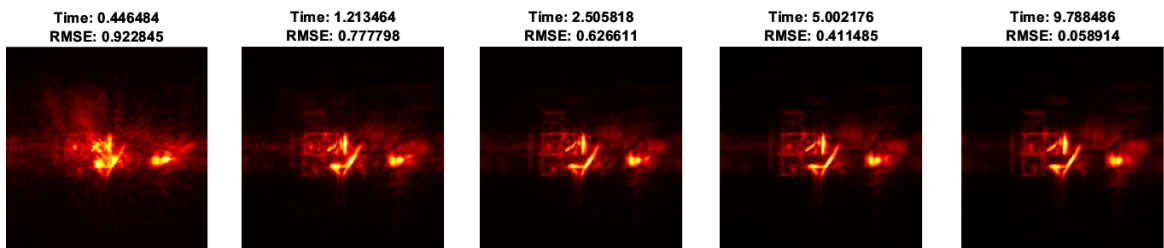


Figura 4.2: De izquierda a derecha se han utilizado: 5, 15, 30, 60 y 120 frecuencias.

En la Figura 4.2 se puede observar cómo, a medida que los rangos de frecuencias son más estrechos, disminuye la calidad de la reconstrucción pero también el tiempo de cómputo. Un resultado destacable es que con 15 y 30 frecuencias se obtienen resultados similares al experimento base empleando casi un orden de magnitud menos de tiempo.

Muestreo por importancia

Este método elige frecuencias al azar. Para ello, se genera un número n de frecuencias siguiendo una distribución Gaussiana modelada con conocimiento previo. Los resultados, visibles en la Figura 4.3 han disminuido su calidad con respecto al método previo. Esto se debe a la selección de deltas, que producen ondas sinusoidales en espacio primal, es decir, hay mayor propensión de artefactos aún usando las mismas frecuencias.

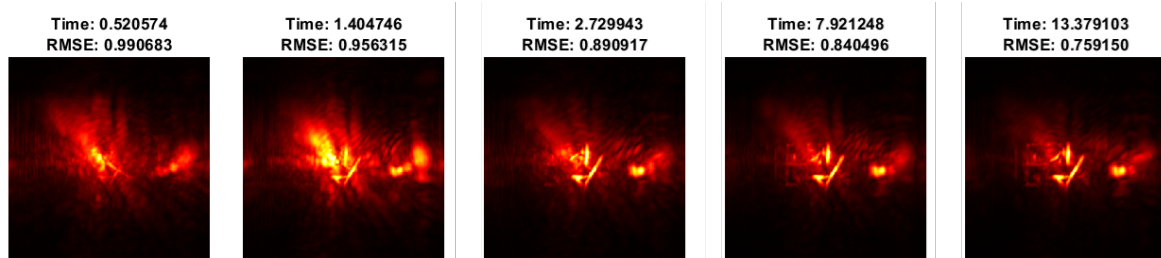


Figura 4.3: De izquierda a derecha se han utilizado: 5, 15, 30, 60 y 120 frecuencias.

Muestreo por importancia suavizado

Tras observar el efecto de las deltas, se probó a seleccionar frecuencias colindantes a las elegidas de forma aleatoria. Estas frecuencias colindantes se eligen siguiendo una distribución Gaussiana con desviación igual a 2. Esta vez, los experimentos no cuentan con una cantidad controlable de frecuencias. Si dos frecuencias elegidas se encuentran cerca, pueden compartir frecuencias colindantes, así como ocurrir más variabilidad de casuística. En la Figura 4.4 se puede apreciar como este método, pese a ser peor que el método de rangos, tiene mejores resultados que el método de muestreo por importancia previo sin suavizar.

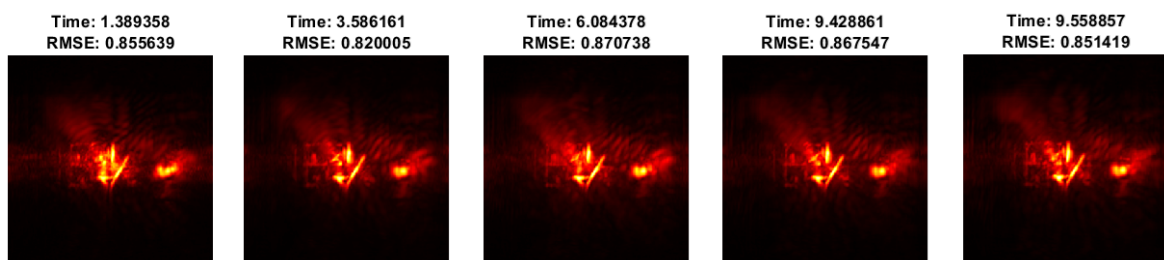


Figura 4.4: De izquierda a derecha se han utilizado: 15, 43, 73 y 92 frecuencias.

4.2.2. Análisis de resultados

Como comparativa final, se ha comparado el tiempo de ejecución con el error obtenido en las reconstrucciones. En la Figura 4.5 se puede comprobar cómo el método de rangos predomina sobre los demás. Los métodos de muestreo por importancia mejoran cuando no aparecen deltas. Sin embargo, en el método suavizado, el error crece rápidamente al incrementar frecuencias debido a sumas de Gaussianas generando picos aleatorios.

Estos resultados indican que, si bien existe un compromiso a la hora de reducir frecuencias, existen formas mejores y peores de hacerlo. De cara al siguiente apartado, se ha visto que métodos como muestreo por importancia no obtienen resultados mejores que un simple filtrado por rangos. También se ha cumplido la hipótesis de que

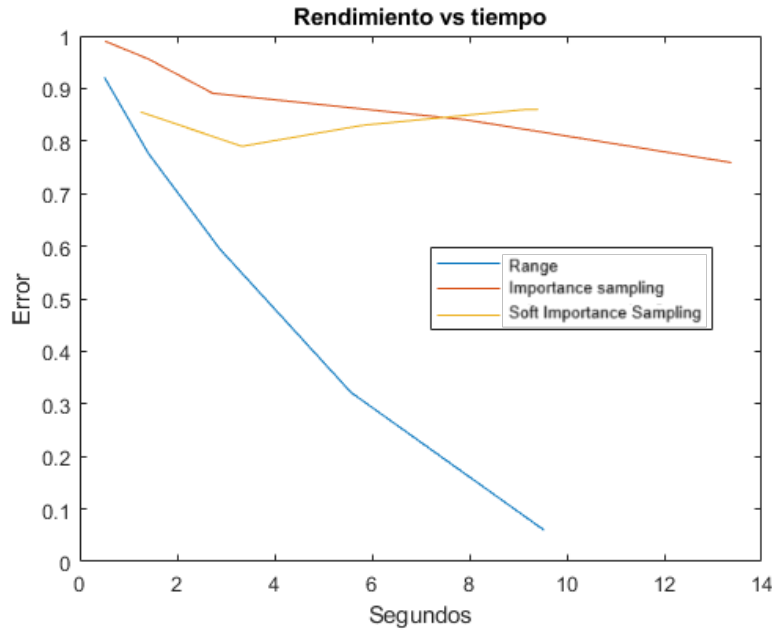


Figura 4.5: Métrica de error con respecto al tiempo de ejecución en los diferentes métodos.

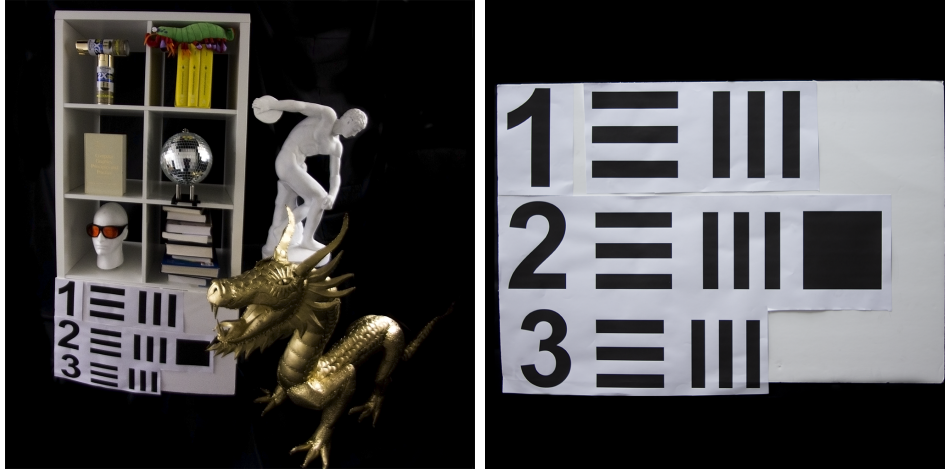
suavizar muestreos por importancia en espacios de frecuencia mejora los resultados. Una Gaussiana en espacio de frecuencias es un *wavelet* u ondícula en el espacio primal, mientras que una delta en frecuencias es simplemente una onda en espacio primal. Esto se ha visto que es importante de cara a reducir artefactos en las reconstrucciones.

4.3. F-K migration

Para evaluar las modificaciones en el algoritmo de *F-K migration*, se realizaron varios experimentos utilizando *datasets* estándares en NLOS imaging (Figura 4.6). El primero de ellos, es una escena con ciertas geometrías y superficies complejas, como la bola de discoteca, el dragón o la estatua del fondo. El segundo es simplemente una cuadrícula plana, con distintas figuras como números o líneas verticales y horizontales. Esta última se usará para probar la resolución del sistema.

Ambos *datasets* han sido obtenidos mediante captura real (no simulado) y cuentan con una resolución en los datos de entrada de $512 \times 512 \times 2048$. Esto quiere decir que las dimensiones espaciales del muro repetidor son de 512×512 mientras que la dimensión temporal contiene 2048 unidades de tamaño. Esta resolución de los datos de entrada marca la resolución de salida del algoritmo, en este caso, una reconstrucción de 512×512 . Esta resolución es demasiado alta para el algoritmo original, debido a su uso ineficiente de las frecuencias. Intentar correr el *dataset* completo en un ordenador convencional resultaría en un error por falta de memoria. Aún en el mejor de los casos,

el algoritmo conseguiría funcionar a costa de emplear un tiempo no razonable para este problema. Por lo que en el código original reducen el dataset a una resolución de $128 \times 128 \times 2048$. Es decir, 16 veces menos tamaño.



(a) Dataset 2: teaser.

(b) Dataset 3: resolution.

Figura 4.6: Datasets usados con F-K migration.

4.3.1. Experimentación

Las mejoras propuestas para *F-K migration* en este trabajo pasan primero por filtrar los datos. Para ello, se aplica el filtro de *Phasor Fields*, que cuenta con dos parámetros: frecuencia base y número de oscilaciones. Dados estos dos parámetros, se puede comprobar su comportamiento de forma sencilla mediante experimentación. Cómo se puede comprobar en el Anexo A, cambiar ligeramente los parámetros puede alterar el resultado tanto en calidad como en coste. En los siguientes apartados se detallará cómo afecta cada parámetro del filtro a las reconstrucciones.

En *F-K migration* se han realizado dos implementaciones. Como se ha explicado en el capítulo anterior, la primera implementación aplicaba la interpolación a unos rangos en espacio de frecuencias, cuidadosamente definidos, en función de las propiedades del filtro. La segunda implementación hacía uso de estos rangos para seleccionar los espacios de memoria deseados. Sin embargo, en la interpolación, se interpolaban solamente en los valores útiles. En este apartado, se considera al primer método un paso intermedio de la implementación del segundo. Por lo tanto, se van a mostrar resultados solamente del segundo método.

4.3.2. Análisis de resultados

A continuación, se van a exponer los resultados obtenidos mediante la implementación del algoritmo de *F-K migration* con las modificaciones propuestas.

Se han considerado diferentes configuraciones de los parámetros del filtro de *Phasor Fields*: la longitud de onda λ y la cantidad de ciclos ϕ . Para facilitar la comparación, se realizaron experimentos sobre los dos *datasets* distintos, descritos previamente.

El primer experimento se centra en la variación de la longitud de onda λ , mientras que el segundo analiza el impacto de modificar la cantidad de ciclos ϕ del filtro. Por último, se comparan los resultados del método mejorado con los métodos actuales, utilizando diferentes parámetros y resoluciones para evaluar su eficacia en la reconstrucción de imágenes complejas.

Variación de Lambda

Se ha probado variando primero a cambiar la longitud de onda λ para apreciar su efecto. λ es la longitud de la onda portadora (*carrier*) del filtro de *Phasor Fields*. En la Figura 4.7 se ve claramente el efecto que tiene cambiar la longitud de onda del filtro.

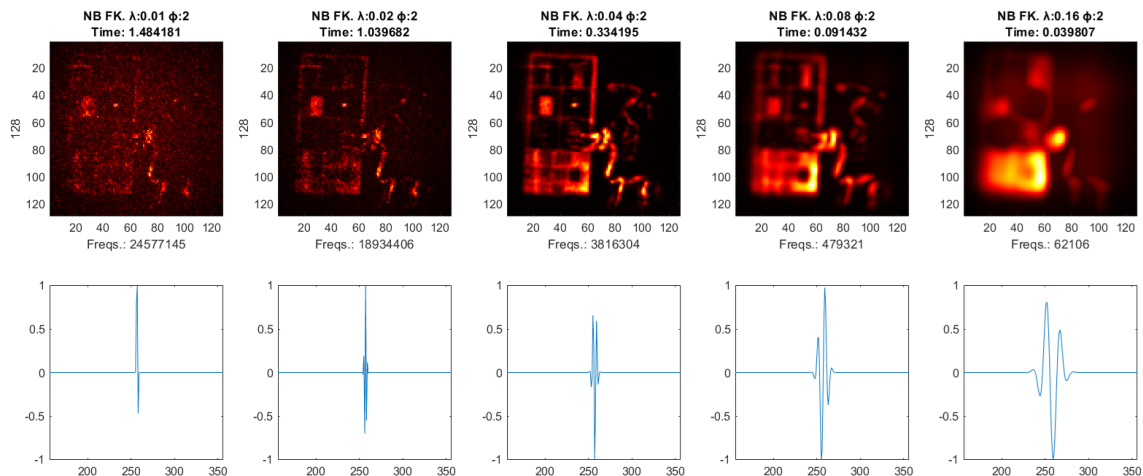


Figura 4.7: Arriba se encuentran los resultados obtenidos, mientras que abajo, los filtros aplicados. De izquierda a derecha se ha empleado $\lambda = (0.01, 0.02, 0.04, 0.08$ y $0.16)$.

En cuanto a los resultados, una menor longitud de onda proporciona mayor sensibilidad a los elementos de alta frecuencia, es decir, aquellos más pequeños, generando en el extremo ruido de alta frecuencia. Mientras que una alta longitud de onda resalta los cuerpos y estructuras de mayor volumen, perdiendo la nitidez que otorgan las altas frecuencias. Por otro lado, se puede percibir cómo el tiempo disminuye. Esto es debido a la no sobreyectividad² de la función de interpolación comentada en el Anexo B, la cual se localiza sobretodo en las bajas frecuencias. Esto hace que haya menos frecuencias sobre las que evaluar en la interpolación, y con ello, menos tiempo de cómputo.

²Sobreyectividad: propiedad de una función que indica si es exhaustiva, es decir, si cada elemento del dominio destino tiene al menos una preimagen en el dominio original.

Cantidad de ciclos

Por otro lado, está el parámetro ϕ . Este parámetro representa la cantidad de ciclos que realiza la ondícula. Esto significa, la cantidad de longitudes de onda comprendidas en el radio delimitado por 2σ al rededor de la Gaussiana. Si probamos a cambiar ϕ , veremos un efecto muy diferente al de cambiar λ . En la Figura 4.8 se puede ver el efecto de aumentar el número de ciclos del filtro aplicado así como una comparativa con el resultado original de *F-K migration*.

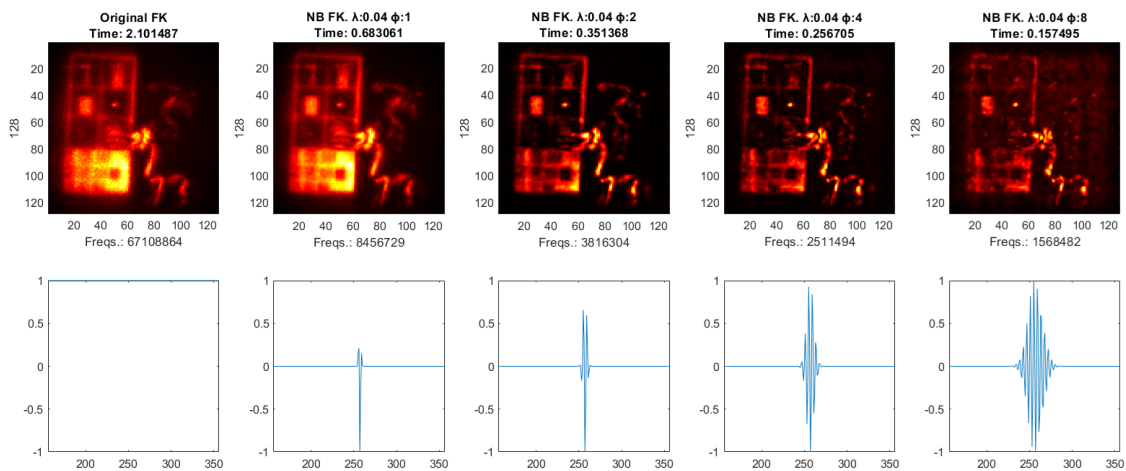


Figura 4.8: Arriba se encuentran los resultados obtenidos, mientras que abajo, los filtros aplicados. La primera columna representa el método original. El resto, de izquierda a derecha, han empleado $\phi = (1, 2, 4 \text{ y } 8)$.

Una de las cosas que definen a este parámetro es la relación cantidad de frecuencias versus tiempo. Una gaussiana en espacio primal sigue siendo una gaussiana en espacio de frecuencias, sin embargo, sus desviaciones son inversamente proporcionales. Es por esto que al incrementar el número de ciclos, y con ello la desviación de la Gaussiana aplicada a la sinusoidal, estamos decrementando la desviación de la curva Gaussiana en espacio de frecuencias. Esto nos deja con un rango de frecuencias distintas de 0 mucho más pequeño, con menos calidad en el resultado debido a la poca cantidad de datos, pero a la vez, con muchos menos elementos que computar.

A parte de la mejora en cuanto a tolerancia a ruido, es notable el tiempo que se ahorra en la interpolación usando este método, así como el ahorro en memoria. Este es tal, que ahora se pueden ejecutar *datasets* 4 veces más grandes, obteniendo mejores resultados en rangos de tiempos similares.

Comparación con métodos actuales

En este apartado, se van a comparar cuatro formas diferentes de reconstruir la escena para dos *datasets* diferentes. La primera forma es el algoritmo F-K original, sin ninguna modificación. Las siguientes emplean la técnica de reducción de frecuencias anteriormente expuesta, empleando diferentes configuraciones. La primera de ellas enfrenta el *dataset* con 4 veces más resolución. Las otras dos emplean el mismo *dataset* que el algoritmo original. Una de estas dos está más enfocada a obtener mayor calidad de imagen. La otra, busca reducir el tiempo.

En la Figura 4.9 podemos observar los diferentes métodos de reconstrucción, junto con el tiempo que tomó la interpolación, los parámetros usados en caso de Narrow Band (usando el filtrado propuesto) y la cantidad de frecuencias interpoladas.

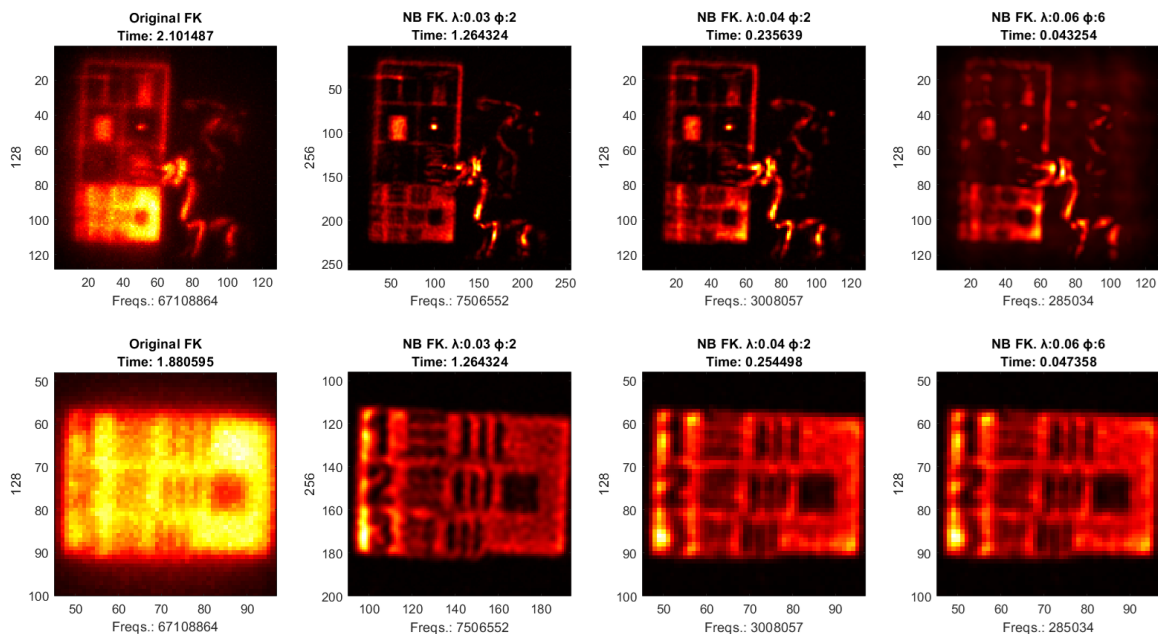


Figura 4.9: Comparativa entre modelos y parámetros en dos *datasets*. De izquierda a derecha: F-K original. Narrow Band (NB) F-K con 4 veces más resolución (y 4 veces más datos de entrada). NB F-K con parámetros ajustados a calidad. NB F-K con parámetros ajustados a menos tiempo.

Uno de los puntos más destacables de estos resultados es la ejecución sobre el *dataset* con 4 veces más datos (columna 2) que los que emplea el método original. Como era de esperar, al poseer 4 veces más resolución, se logra una mejor calidad en cuanto a la definición de objetos, números y formas. Lo que es sorprendente es el tiempo que emplea, pues el algoritmo modificado emplea menos tiempo y carga computacional que el método original, que tiene 4 veces menos datos de entrada. Los otros dos métodos son igualmente sorprendentes. Por un lado, el método enfocado en una mejor calidad (columna 3), aún usando el mismo *dataset*, logra una mayor definición en las formas

y una sustancial reducción del ruido en comparación con el método original. Además, reduce el tiempo de ejecución en cerca de un orden de magnitud, logrando un *speedup*³ del 8.2 para esta calidad de imagen. Al depender el filtro de *Phasor Fields*, se pueden ajustar los parámetros para forzar al algoritmo a ir más rápido, a costa de la calidad. Es por ello que los resultados donde se ha primado la velocidad (columna 4) ofrecen ruido de fondo o artefactos en la escena de la habitación. Por otro lado, en la escena de la cuadrícula, se puede apreciar cómo la calidad a penas ha disminuido. Esto es relevante ya que indica que para ciertas escenas o aplicaciones, se podrían hacer reconstrucciones de calidad aceptable en tiempo real, empleando menos de 0.05s, lo que es un *speedup* del 44.3. Esto, de contar con un *hardware* lo suficientemente potente como para capturar la información igual de rápido, podría llevar a un hito en el campo de NLOS *imaging*, al permitir ver detrás de obstáculos en tiempo real con una alta resolución. Si bien otras investigaciones ([6], [10], [11]) han logrado realizar NLOS *imaging* en tiempo real, las contribuciones aportadas en este trabajo permitirían aumentar tanto la resolución como la tasa de fotogramas por segundo sustancialmente.

Todos los beneficios del método propuesto proceden de la pequeña cantidad de frecuencias que emplea en la interpolación. Siendo llamativo el caso de que, aun empleando el *dataset* con 4 veces más tamaño, la cantidad de frecuencias es casi un orden de magnitud menor al método original. Esto es una gran ventaja de cara, tanto a los tiempos de ejecución anteriormente mencionados, como a requisitos de memoria y consumo energético.

En resumen, los resultados arrojan varias conclusiones: se ha conseguido reducir la carga computacional en cerca de una orden de magnitud y media, manteniendo una calidad similar en las reconstrucciones. A su vez, se ha logrado aumentar la resolución a 4 veces más, manteniendo tiempos razonables y pudiendo ejecutar el algoritmo en un ordenador convencional, sin que desborde la memoria debido al uso eficiente de las frecuencias. En cuanto a los parámetros del filtro, tanto los parámetros como sus efectos parecen ser extrapolables a otros *datasets*, consiguiendo resultados similares, lo cual generaliza el método. A su vez, se ha aumentado la tolerancia a ruido, tanto de alta como de baja frecuencia. Esto es debido al filtro de *Phasor Fields* y control sobre el espacio de frecuencias que posee, pudiendo seleccionar las frecuencias adecuadas para obtener resultados con suficiente contraste pero sin ruido ni difuminación.

³*speedup*: aceleración. En este caso, ratio entre el tiempo que empleaba el algoritmo original y las diferentes versiones de las modificaciones.

Capítulo 5

Conclusiones

En este trabajo se han explorado diferentes técnicas para optimizar el uso de frecuencias en los algoritmos de reconstrucción de escenas ocultas Phasor Fields y F-K migration.

En general, los métodos propuestos han demostrado su capacidad para mejorar la eficiencia de los algoritmos de reconstrucción de escenas ocultas, manteniendo una calidad de imagen buena. En algunos casos, se ha podido lograr una mejor calidad de imagen en menos tiempo, Esto es dado que las optimizaciones propuestas permiten abordar *datasets* con mayor resolución, manteniendo tiempos de ejecución razonables y uso de memoria normal.

Las modificaciones en el código de *Phasor Fields* indican que es posible reducir las frecuencias a calcular, todo a costa de la calidad de la reconstrucción. Es un compromiso que puede ser menos problemático si se abordan técnicas inteligentes, como evitar deltas en espacio de Fourier. Como conclusión, los métodos de selección de rangos de frecuencias han resultado ser los mejores, seguidos del muestreo por importancia suavizado y, por último, el muestreo por importancia sin suavizar. Se han conseguido unos *speedups* de cerca de un orden de magnitud para resultados de calidad aceptable.

La propuesta de combinar *F-K migration* con el filtrado de *Phasor Fields* otorga resultados prometedores, permitiendo reconstrucciones aceptables, de elevada resolución en tiempo real. En general esto se debe a la selección eficiente de frecuencias, que permite recortar el espacio de Fourier sustancialmente, abaratando la operación de interpolación. Finalmente, este método ha logrado que procesar *datasets* con el doble de resolución no sea una tarea tan computacionalmente demandante. Esto se ha logrado hasta el punto que una reconstrucción de estas características emplea menos tiempo y memoria que el método original. Cabe mencionar el *speedup* de 44.3 veces más rápido con respecto al método original logrado. Si bien para el dataset de la habitación genera ruido en el fondo, en el de la cuadrícula produce resultados similares al método original en cuanto a calidad se refiere.

Por último, es importante mencionar que todas las modificaciones poseen componentes parametrizables. Esto significa que es relativamente sencillo ajustar los diferentes parámetros de los filtros para lograr objetivos concretos en cuanto a la calidad de la reconstrucción, como a la reducción de ruido o como, por supuesto, el tiempo de ejecución. Todo ello prestando atención a los diferentes compromisos que poseen los métodos propuestos.

En resumen, el trabajo ha logrado optimizar la eficiencia computacional en la reconstrucción 3D de escenas ocultas sin línea de visión mediante la implementación de técnicas de filtrado de bandas estrechas de frecuencias. A pesar de los avances logrados, como la reducción significativa en el tiempo de ejecución y el uso de memoria, los métodos desarrollados aún presentan desafíos en cuanto a la precisión al reducir frecuencias a partir de cierto punto. Sin embargo, estas técnicas representan un avance importante hacia la aplicación práctica de la visualización de escenas ocultas en tiempo real, con potencial para mejorar en futuras investigaciones.

Capítulo 6

Bibliografía

- [1] Andreas Velten, Di Wu, Adrian Jarabo, Belen Masia, Christopher Barsi, Chinmaya Joshi, Everett Lawson, Mounqi Bawendi, Diego Gutierrez, and Ramesh Raskar. Femto-photography: Capturing and visualizing the propagation of light. *ACM Trans. Graph.*, 32(4), jul 2013.
- [2] Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. Mitsuba 2: A retargetable forward and inverse renderer. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 38(6), December 2019.
- [3] Matthew O’Toole, David B. Lindell, and Gordon Wetzstein. Confocal non-line-of-sight imaging based on the light-cone transform. *Nature*, 2018.
- [4] Victor Arellano, Diego Gutierrez, and Adrian Jarabo. Fast back-projection for non-line of sight reconstruction. *Optics Express*, 25(10), 2017.
- [5] Guillén I. La Manna M. et al. Liu, X. Non-line-of-sight imaging using phasor-field virtual wave optics. *Nature* 572, 620–623, 2019.
- [6] David B. Lindell, Gordon Wetzstein, and Matthew O’Toole. Wave-based non-line-of-sight imaging using fast f-k migration. *ACM Trans. Graph. (SIGGRAPH)*, 38(4):116, 2019.
- [7] Victor Arellano, Diego Gutierrez, and Adrian Jarabo. Fast back-projection for non-line of sight reconstruction. *Opt. Express*, 25(10):11574–11583, May 2017.
- [8] Xiaochun Liu, Sebastian Bauer, and Andreas Velten. Phasor field diffraction based reconstruction for fast non-line-of-sight imaging systems. *Nature Communications*, (11), 2020.

- [9] Fabin Shen and Anbo Wang. Fast-fourier-transform based numerical integration method for the rayleigh-sommerfeld diffraction formula. *Applied optics*, 45:1102–10, 03 2006.
- [10] Ji Hyun Nam, Eric Brandt, Sebastian Bauer, Xiaochun Liu, Marco Renna, Alberto Tosi, Eftychios Sifakis, and Andreas Velten. Low-latency time-of-flight non-line-of-sight imaging at 5 frames per second. *Nature Communications*, 12(1), November 2021.
- [11] Tomohiro Maeda, Yiqin Wang, Ramesh Raskar, and Achuta Kadambi. Thermal non-line-of-sight imaging. In *2019 IEEE International Conference on Computational Photography (ICCP)*, pages 1–11. IEEE, 2019.
- [12] Dylan Thomas. Do not go gentle into that good night, 1951. In *The Collected Poems of Dylan Thomas*, New Directions Publishing, 1952.

Lista de Figuras

1.1.	(a) <i>Setup</i> de NLOS. Al lado izquierdo del ocluser se encuentra la geometría oculta. En frente de esta, se encuentra la pared visible. Al otro lado del ocluser se encuentra el sistema de emisión y captación de luz, con un láser emitiendo el pulso de luz y un sensor SPAD captando el tiempo de vuelo de la luz. (b) Gráfica de la intensidad de la luz captada en función del tiempo transcurrido. (c) Imagen de un frente de onda avanzando (figura de [3]).	2
1.2.	Escena oculta reconstruida y método de lentes virtuales (Figura (a) de [5]).	3
2.1.	Un pulso de luz es emitido por el láser hacia un punto s visible. El pulso rebota en un punto desconocido x . Finalmente vuelve a un punto p de la pared visible, donde posteriormente es captado por el observador (figura de [7]).	8
2.2.	Pasos de Phasor Fields. (a) Se emite un pulso de luz en un punto de la superficie visible, posteriormente se puede hacer con más. (b) Se captura la intensidad de la luz en cada momento en la superficie visible. (c) Fuente de luz virtual. La onda virtual de campos de fasores se emite a la escena oculta. (d) La escena responde a la iluminación proyectando en la pared visible un campo de fasores. (e) La escena es finalmente reconstruida usando wave projection. (figura de [5]).	11
2.3.	Capas de profundidad en una reconstrucción de escena NLOS.	12
2.4.	Filtros en espacio primal y de frecuencias.	14
3.1.	Diferentes formas de seleccionar frecuencias. De izquierda a derecha: potencia de las frecuencias originalmente (tiene forma de Gaussiana ya que se aplicó el filtro de <i>Phasor Fields previamente</i>); potencia de las frecuencias al aplicar método de rangos; frecuencias al aplicar el método de muestreo por importancia; las mismas frecuencias usadas como medias en curvas Gaussianas (muestreo por importancia suavizado). 17	17

3.2.	Efectos de la interpolación. En el centro está el espacio de frecuencias original representado como un ajedrezado para mejor visualización. En este ajedrezado, se muestran resaltados los rangos de frecuencias distintos de 0. Es decir, las frecuencias dentro del rango que crea el filtro de <i>Phasor Fields</i> . A la derecha, se puede ver el efecto de aplicar la interpolación al espacio de frecuencias. Por último, a la izquierda se ve de dónde ha leído el valor cada punto de la cuadrícula. Nótese que esto a su vez sería el efecto de aplicar la función inversa de interpolación. . .	19
4.1.	Dataset usado con Phasor Fields.	22
4.2.	De izquierda a derecha se han utilizado: 5, 15, 30, 60 y 120 frecuencias.	23
4.3.	De izquierda a derecha se han utilizado: 5, 15, 30, 60 y 120 frecuencias.	24
4.4.	De izquierda a derecha se han utilizado: 15, 43, 73 y 92 frecuencias. . .	24
4.5.	Métrica de error con respecto al tiempo de ejecución en los diferentes métodos.	25
4.6.	Datasets usados con F-K migration.	26
4.7.	Arriba se encuentran los resultados obtenidos, mientras que abajo, los filtros aplicados. De izquierda a derecha se ha empleado $\lambda = (0.01, 0.02, 0.04, 0.08 \text{ y } 0.16)$	27
4.8.	Arriba se encuentran los resultados obtenidos, mientras que abajo, los filtros aplicados. La primera columna representa el método original. El resto, de izquierda a derecha, han empleado $\phi = (1, 2, 4 \text{ y } 8)$	28
4.9.	Comparativa entre modelos y parámetros en dos <i>datasets</i> . De izquierda a derecha: F-K original. Narrow Band (NB) F-K con 4 veces más resolución (y 4 veces más datos de entrada). NB F-K con parámetros ajustados a calidad. NB F-K con parámetros ajustados a menos tiempo.	29
A.1.	Cuadrícula donde se ha experimentado con las diferentes longitudes de onda y cantidad de ciclos.	39
B.1.	Efectos de la interpolación. En el centro está el espacio de frecuencias original representado como un ajedrezado para mejor visualización. A la derecha, se puede ver el efecto de aplicar la interpolación a la cuadrícula. Por último, a la izquierda se ve de dónde ha leído el valor cada punto de la cuadrícula, esto a su vez sería el efecto de aplicar la función inversa de interpolación.	42

Anexos

Anexos A

Experimentación de parámetros de Narrow-Band F-K migration

En la figura A.1 se pueden observar los resultados de combinar diferentes longitudes de onda y cantidad de ciclos. Conforme se aumenta λ , la longitud de onda del filtro, se obtienen resultados más difuminados. Lo contrario pasa al disminuir, obteniendo resultados más contrastados y con más ruido fino. Por otro lado, el parámetro ϕ , es decir, la cantidad de ciclos de la ondícula del filtro, regula cuánto de estrecha o no es la banda. Al aumentar ϕ , el rango de frecuencias que comprende el filtro es menor. Por ello, al trabajar con menos datos, las reconstrucciones son más rápidas a costa de la calidad de estas.

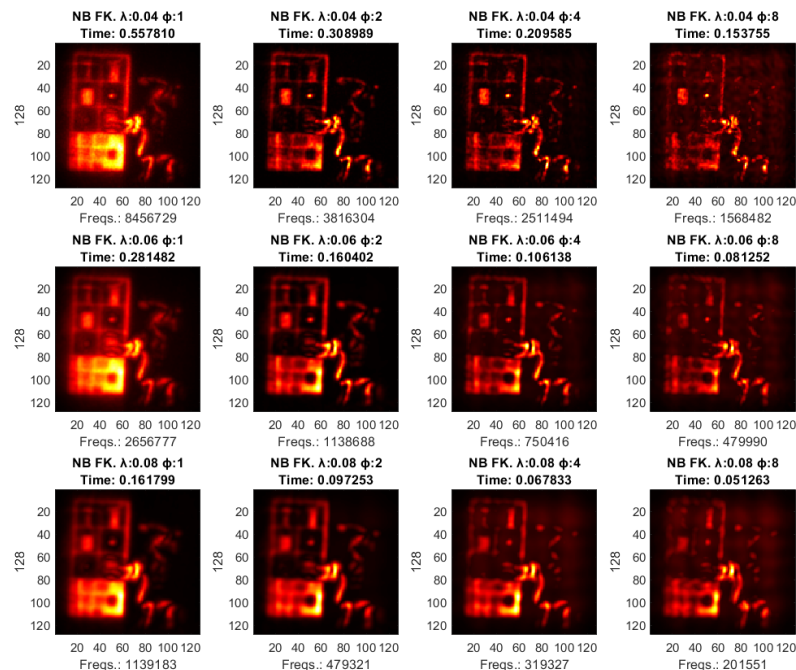


Figura A.1: Cuadrícula donde se ha experimentado con las diferentes longitudes de onda y cantidad de ciclos.

Anexos B

No sobreyectividad de la función de interpolación

A la hora de interpolar datos, como se hace en *F-K migration*[6], conviene tener en consideración ciertas propiedades de la función de interpolación a la hora de intentar sacarle el mayor rendimiento. Una de las propiedades de esta función es específico es que no es sobreyectiva. Es decir, existen valores en el dominio destino que, sin importar la entrada de la función, no van a poder alcanzarse. Esto en la práctica significa que la función de interpolación no podrá “leer” de ciertos puntos del dominio de frecuencias.

La no sobreyectividad de la función de interpolación $\Omega' = f(\mathbf{x}_c, \Omega) = \sqrt{\rho * (x_c^2 + y_c^2) + \Omega^2}$ puede verse representada en la Figura B.1, donde existen puntos del dominio (la cuadrícula) que no aparecen en el resultado. Esto se puede ver también en la representación de la función inversa donde existen puntos a la izquierda que no son consultados por ningún punto del dominio. Visto esto es fácilmente demostrable que:

$$\{\nexists \mathbf{x}_c \in \mathbb{R}^2 \text{ y } \Omega \in \mathbb{N} \mid \Omega' = \sqrt{\rho * (x_c^2 + y_c^2) + \Omega^2} \text{ y } \Omega' < \sqrt{\rho * (x_c^2 + y_c^2)}\} \quad (\text{B.1})$$

Esto significa que existen posiciones que no van a ser consultadas en la interpolación, y por lo cual, no tienen inversa. Siendo esta la demostración de que la función es no sobreyectiva.

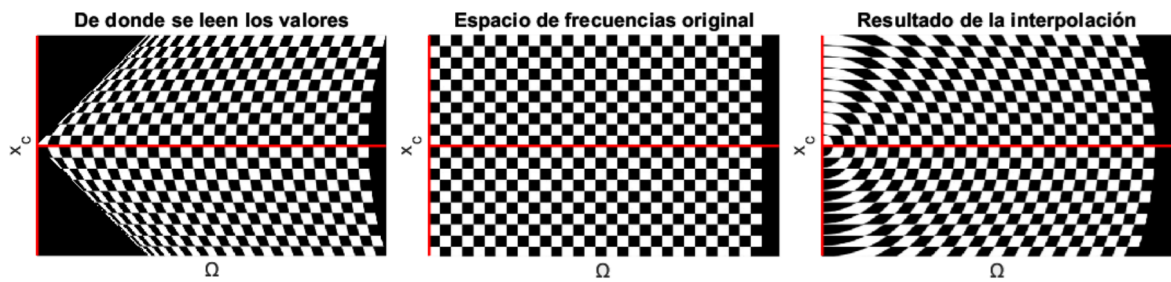


Figura B.1: Efectos de la interpolación. En el centro está el espacio de frecuencias original representado como un ajedrezado para mejor visualización. A la derecha, se puede ver el efecto de aplicar la interpolación a la cuadrícula. Por último, a la izquierda se ve de dónde ha leído el valor cada punto de la cuadrícula, esto a su vez sería el efecto de aplicar la función inversa de interpolación.