

Trabajo fin de grado  
Grado en Física

**Desarrollo de un algoritmo para la calibración de la distorsión óptica en  
adquisición de imágenes satelitales y comprobación experimental**  
Lanaren azpitituloa / Subtítulo del trabajo

Egilea/ Autor/a:  
Xabier Joel Yoldi García  
Zuzendariak/Directores/as:  
Josu Mirena Igartua Aldamiz  
Iñigo Irizar Loibide

© 2021, se puede proteger poniendo nombre y apellido/izen abizenak  
ezarriz babes zaitezke edo lizentzia CC batekin babestu / o con una  
licencia CC  
<http://es.creativecommons.org/blog/licencias/>

# Índice

<b>1</b>	<b>Introducción</b>	<b>2</b>
<b>2</b>	<b>Sistemas Electro-Ópticos</b>	<b>3</b>
2.1	Sistemas Ópticos . . . . .	3
2.1.1	Respuesta al impulso . . . . .	3
2.1.2	Dominio de frecuencias . . . . .	4
2.2	Fuentes de degradación en la calidad de la imagen . . . . .	7
2.2.1	Relación señal a ruido (SNR) . . . . .	7
2.2.2	Aberraciones ópticas . . . . .	8
2.2.3	Distorsión Óptica . . . . .	9
2.2.4	Modelado Matemático . . . . .	9
2.3	Métodos de corrección de distorsión . . . . .	10
2.3.1	Métodos clásicos . . . . .	10
2.3.2	Métodos de aprendizaje basados en regresión . . . . .	10
2.3.3	Métodos de aprendizaje basados en reconstrucción . . . . .	11
2.4	Métricas . . . . .	11
<b>3</b>	<b>Método propuesto</b>	<b>12</b>
3.1	Primer paso: detección de los centroides de los pinholes . . . . .	12
3.2	Segundo paso: caracterización de la placa de pinholes, y modelización de las pinholes ideales . . . . .	15
3.3	Tercer paso: cálculo de la constante de distorsión. . . . .	16
<b>4</b>	<b>Etapa de simulación</b>	<b>17</b>
4.1	Generación de la imagen ideal . . . . .	18
4.2	Introducción de la distorsión radial en la imagen ideal . . . . .	19
4.3	Aplicación del método propuesto para la detección de la distorsión . .	20
4.4	Evaluación de los resultados obtenidos. . . . .	21
4.4.1	Simulación 1 . . . . .	21
4.4.2	Simulación 2 . . . . .	21
4.4.3	Simulación 3 . . . . .	22
4.4.4	Simulación 4 . . . . .	24
4.4.5	Simulación 5 . . . . .	24
4.4.6	Simulación 6 . . . . .	25
4.4.7	Simulacion 7 . . . . .	26
4.4.8	Simulacion 8 . . . . .	27
<b>5</b>	<b>Aplicación en sistemas ópticos de Satlantis</b>	<b>27</b>
<b>6</b>	<b>Conclusiones</b>	<b>31</b>

## 1 Introducción

La industria aeroespacial y de observación terrestre ha crecido en los últimos años ampliamente y cada vez demanda sistemas más precisos debido a la carrera tecnológica que existe. En este contexto, la empresa Satlantis Microsats se ha especializado en el desarrollo de cámaras de alta resolución para pequeños satélites (microsatélites). Para garantizar la calidad y la fiabilidad de las imágenes capturadas, es esencial el desarrollo de sistemas con una calidad excelente y una mecánica cuidada al detalle. Sin embargo, es fundamental procesar las imágenes para caracterizar y corregir las posibles aberraciones ópticas que pudieran tener los sistemas de captura. Este campo, denominado comúnmente como *Image Processing*, es el encargado de realizar dichas correcciones, así como de aumentar la calidad de las imágenes o caracterizar patrones para futuras aplicaciones.

En la actualidad, una aberración presente en todos los sistemas ópticos es la distorsión óptica. Esta aberración es producida por la forma esférica característica de muchas lentes y provoca un ligero desplazamiento en las posiciones de la imagen. Principalmente, existen dos formas en las que se distorsionan las capturas, denominadas tipo **barrel** o **pincushion**. Debido a la gran sensibilidad que existe para caracterizar esta aberración, la forma más sencilla y utilizada para abordar este problema es corregirla en base a medidas experimentales de cada sistema. Es por ello que en este Trabajo de Fin de Grado se desarrolla una metodología para la medida y análisis de la distorsión óptica con el objetivo final de aplicarlas en las cámaras de Satlantis.

Para ello, el algoritmo desarrollado se ha planteado en el contexto de un banco de ensayo experimental que consiste en una placa de pinholes (orificios del orden de los micrómetros), que, al alumbrarlos, actúan como fuentes puntuales de luz. Las fotografías tomadas de esta placa permiten, mediante técnicas desarrolladas en este Trabajo de Fin de Grado, medir los desplazamientos relativos de cada punto y así calcular el perfil de distorsión introducido por la lente.

Como fase preliminar y para realizar una validación de la metodología, se ha elaborado una etapa de simulaciones. En esta etapa se han generado imágenes replicando las imágenes proporcionadas por Satlantis, y se les ha aplicado una distorsión artificial. Esto ha permitido desarrollar y ajustar los algoritmos de detección y medición del perfil de distorsión.

La memoria se estructura en los siguientes apartados: en primer lugar, se explica el marco teórico, el fenómeno de distorsión óptica y como modelarlo matemáticamente. A continuación, se desarrolla la metodología propuesta para la detección de distorsión y se valida con diferentes simulaciones. Finalmente, se presentan los resultados obtenidos con las imágenes de Satlantis y se concluye con una valoración general del método propuesto.

## 2 Sistemas Electro-Ópticos

Los sistemas ópticos conforman un campo muy interesante en el contexto científico debido a, principalmente, a su variedad en este campo. Desde un ojo humano hasta una cámara o, incluso, un telescopio son parte de este área. Sin embargo, cuando se desea caracterizar un sistema óptico en concreto, una medida comúnmente utilizada es la Función de Dispersión de Punto (PSF), que es una función en el dominio espacial. En concreto, se utiliza su función análoga en el espacio de frecuencias, la Función de Transferencia Óptica (OTF), que no es más que su transformada de Fourier bidimensional. Esta función da cuenta de la respuesta a una señal sinusoidal periódica a través de los diferentes componentes del sistema óptico, en función de su frecuencia espacial [1].

$$OTF = F[PSF] \quad (2.1)$$

La principal ventaja de trabajar con la OTF radica en la facilidad que muestra para estudiar el comportamiento de un sistema donde varios subsistemas están combinados. Esto es porque la función de transferencia de todo el sistema óptico puede expresarse como multiplicación de la función de transferencia de cada subsistema individual. Además, es de gran utilidad para detectar e implementar mejoras en cada subsistema.

No obstante, la calidad de la OTF y, en consecuencia, del sistema óptico, también depende de los componentes electrónicos que lo conforman. Los principales elementos que afectan a la OTF son los píxeles que conforman el sensor óptico y el ruido electrónico (que afecta a todos los aparatos electrónicos). Por lo tanto, es necesario saber cómo el tamaño del píxel afecta a la OTF de todo el sistema electro-óptico.

### 2.1 Sistemas Ópticos

#### 2.1.1 Respuesta al impulso

La respuesta al impulso  $h(x, y)$  o PSF, describe la respuesta de un sistema de imágenes a una fuente puntual o un objeto puntual. Esta función viene descrita por la anchura de la respuesta al impulso y, en esencia, es el resultado de la combinación de aberraciones y efectos de difracción, inherentes a los sistemas ópticos.

Comúnmente, la PSF es interpretada como la distribución del brillo en función de la posición. En sistemas ópticos incoherentes, si se capturan dos objetos diferentes al mismo tiempo, la imagen de cada objeto no se ve afectada por el otro, debido a la propiedad no interactuante de los fotones. Esto significa que la imagen resultante  $g(x, y)$  se puede expresar como la imagen ideal  $f(x, y)$  convolucionada (\*) con la respuesta al impulso  $h(x, y)$ .

$$g(x, y) = f(x, y) * h(x, y) \quad (2.2)$$

**Prueba:**

Considerando una situación unidimensional, la señal de entrada  $f(x)$  es analizada por el sistema óptico (lineal e invariante) aplicando la transformación  $T$  a la señal de entrada, y este sistema devuelve la señal de salida  $g(x)$ .

$$g(x) = T[f(x)]$$

Esto se puede escribir de la siguiente manera usando la función Delta:

$$g(x) = T \left[ \int_{-\infty}^{\infty} f(\tau) \delta(x - \tau) d\tau \right]$$

Dado que el sistema óptico es lineal,

$$g(x) = \int_{-\infty}^{\infty} f(\tau) T[\delta(x - \tau)] d\tau$$

La respuesta al impulso se define como la respuesta del sistema óptico a una fuente puntual, idealmente una función Delta:

$$h(x) = T[\delta(x)]$$

Entonces, debido a que el sistema óptico también es invariante,

$$g(x) = \int_{-\infty}^{\infty} f(\tau) h(x - \tau) d\tau$$

Esta última ecuación es precisamente la definición matemática de convolución.

$$g(x) = f(x) * h(x)$$

Este cálculo se puede extrapolar a una situación bidimensional, obteniendo un resultado análogo a la Ecuación 2.2.

$$g(x, y) = f(x, y) * h(x, y)$$

De este resultado se puede derivar lo siguiente: si el sistema tuviera una resolución y calidad perfectas, la respuesta del sistema coincidiría con la imagen ideal. Atendiendo a la Ecuación 2.2, esto es lo mismo que afirmar que la respuesta del sistema  $h(x, y)$  es una función delta. Lo que concluye en la idea de que la PSF da cuenta de la calidad de la imagen en función de su ancho espacial: cuanto más estrecha es la PSF, mejor es la imagen final.

### 2.1.2 Dominio de frecuencias

Como bien se ha comentado anteriormente, la PSF y OTF son funciones análogas en distintos dominios. En esta sección se aborda el proceso de obtención de imágenes desde la perspectiva frecuencial, es decir, estudiando la OTF en vez de la PSF.

Según el teorema de la convolución, una convolución en el dominio espacial es igual a una multiplicación en el dominio de frecuencia. De este modo, al aplicar una

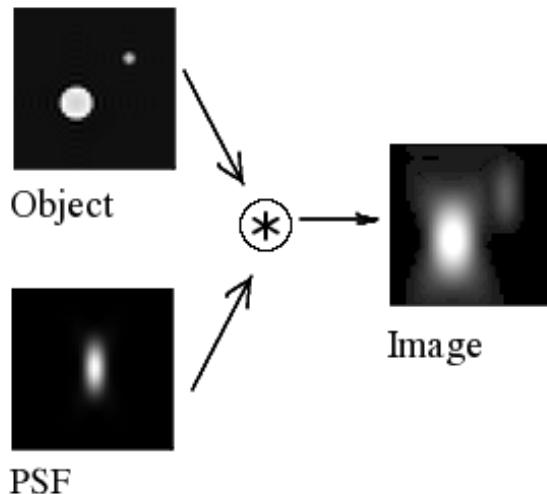


Figura 2.1: Ilustración mostrando la imagen ideal de dos puntos brillantes, la PSF del sistema óptico y la imagen resultante obtenida al hacer la convolución de los dos primeros.

transformada de Fourier en ambos lados de la Ecuación 2.2 se obtien la siguiente expresión:

$$\mathcal{F}[g(x, y)] = \mathcal{F}[f(x, y) * h(x, y)] \quad (2.3)$$

Se pueden definir las siguientes funciones en el dominio de frecuencia:

$$G(\xi, \eta) = F(\xi, \eta) \cdot H(\xi, \eta) \quad (2.4)$$

donde  $\xi = 1/X$  y  $\eta = 1/Y$ , suponiendo que  $X$  y  $Y$  son los períodos de las señales sinusoidales en  $x$  e  $y$ .  $H(\xi, \eta)$  es la respuesta del sistema óptico a una fuente puntual en el dominio de frecuencias, es decir, la OTF, que se expresa como  $H(\xi, \eta) = OTF(\xi, \eta)$ .

Si se analiza un sistema óptico compuesto por  $N$  subsistemas, la respuesta del sistema al completo será la multiplicación de la imagen ideal  $F(\xi, \mu)$  y la respuesta individual de cada subsistema  $H_i(\xi, \mu)$ .

$$G(\xi, \eta) = F(\xi, \eta) \cdot \prod_{i=1}^N H_i(\xi, \mu) \quad (2.5)$$

Por lo que, la multiplicación de todas las respuestas individuales de cada subsistema  $H_i(\xi, \mu)$ , se puede interpretar como la respuesta del sistema total.

$$H(\xi, \mu) = \prod_{i=1}^N H_i(\xi, \mu)$$

Comúnmente, esta función es compleja y se puede describir como la multiplicación de su módulo y su fase. En óptica, el módulo se conoce como Función de Transferencia de Modulación (MTF) y la fase como Función de Transferencia de Fase (PTF).

$$OTF = H(\xi, \eta) = |H(\xi, \eta)| e^{i\theta(\xi, \eta)} \quad (2.6)$$

donde

$$MTF = |H(\xi, \eta)| ; \quad PTF = \theta(\xi, \eta)$$

Como consecuencia de las propiedades de las transformadas de Fourier, el hecho de tener una PSF simétrica  $h(\xi, \mu) = h(-\xi, -\mu)$ , hace que su transformada (OTF) sea una función real pura [2]. En otras palabras, una PSF simétrica implica una OTF igual a la MTF y una PTF nula.

Como bien se acaba de introducir, la MTF se define como el módulo de la OTF. En concreto, la MTF es la función que define la capacidad de resolución del sistema óptico. Es decir, con esta herramienta se mide la efectividad que tiene el sistema para representar el contraste de la imagen original a diferentes frecuencias. De esta forma se define la cantidad  $M$  (en inglés **modulation depth**) como:

$$M = \frac{A_{max} - A_{min}}{A_{max} + A_{min}} \quad (2.7)$$

donde  $A_{max}$  y  $A_{min}$  son la irradiación máxima y mínima a contrastar.

Por su definición, la MTF incluye todos los posibles efectos de cualquier imperfección que contribuye a la degradación de la imagen, como la difracción, aberraciones cromáticas, distorsión, etc. Una posible consecuencia directa de estas imperfecciones es la pérdida en la simetría de la PSF. Cuando esto ocurre, la PTF ya no es nula y se vuelve más significativa. Por esta razón, la PTF se utiliza comúnmente para analizar la presencia e importancia de esas imperfecciones.

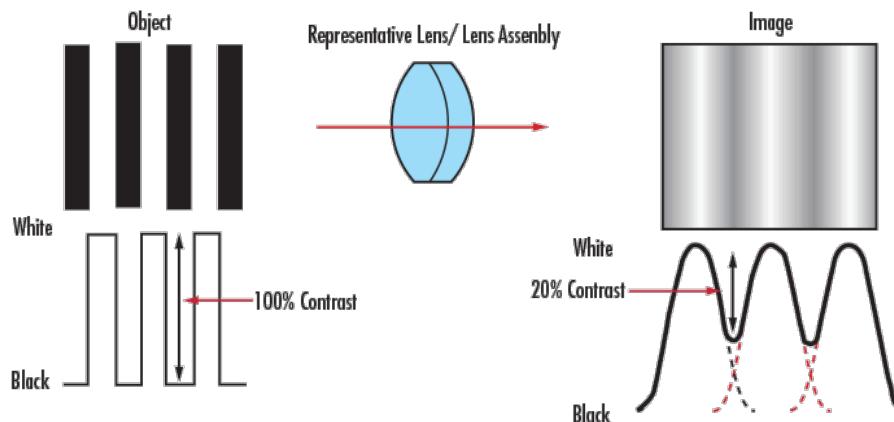


Figura 2.2: Comparación del contraste en los planos objeto e imagen.[3]

En la siguiente sección se dará una introducción a diferentes fuentes que dan lugar a estas imperfecciones. En concreto, se estudiará la distorsión óptica, que es la aberración causa de este trabajo.

## 2.2 Fuentes de degradación en la calidad de la imagen

Como bien se ha comentado, los sistemas ópticos cuentan con multitud de imperfecciones que afectan a la formación de imágenes. Dos grandes causas de una mala formación en las imágenes son las aberraciones ópticas y el ruido electrónico que, como se ha comentado en la Sección 2 afecta a la OTF.

### 2.2.1 Relación señal a ruido (SNR)

Hay tres fuentes principales de ruido que afectan a los sistemas electro-ópticos: corriente oscura, ruido de disparo y ruido de lectura.

- **Corriente oscura:** Es el efecto de la corriente eléctrica relativamente pequeña que fluye a través de dispositivos fotosensibles como los sensores, incluso cuando no entran fotones en el dispositivo. Físicamente, la corriente oscura es debida a la generación aleatoria de electrones y huecos dentro de la región de agotamiento del dispositivo. Estos pares electrón-hueco se generan por energía térmica, por lo que este ruido está directamente relacionado con la temperatura. Cuanto mayor es la temperatura a la que trabaja el detector, mayor será este ruido. Además, este ruido depende del tiempo de exposición, ya que se generarán más electrones cuanto más tiempo dure el uso. Se puede corregir restando a la imagen un fotograma (incluso se podría mejorar haciéndolo con la media de muchos) oscuro, denominado *Dark Frame*.
- **Ruido de disparo:** El ruido de disparo existe porque fenómenos como la luz y la corriente eléctrica consisten en el movimiento de paquetes discretos. Las fluctuaciones en la corriente de luz que se dan por la cuantización de la luz son la causa de este tipo de ruido. Con un ejemplo se entiende mejor. Un sensor es golpeado por una intensidad lumínica  $I$ , esto quiere decir que, de media, en un segundo golpearán un número de fotones proporcionales a  $I$ . Sin embargo, dependiendo del tiempo de exposición de la imagen y más variables aleatorias, la cantidad de fotones que recibe el sensor es *pseudo-aleatoria*, dando lugar a detecciones parásitas que son más significativas a bajas intensidades (por ejemplo al obtener una *Dark Frame*). Estas fluctuaciones siguen una distribución de Poisson.

$$P[X = k] = \frac{e^{-S} S^k}{k!} \quad (2.8)$$

donde  $k = 0, 1, 2, \dots$  es el número de fotones que golpean el sensor en un intervalo de tiempo concreto. La cantidad  $S$  se puede entender como el número de fotones por unidad de área y unidad de tiempo (en este contexto por  $\mu\text{m}^2$  por segundo). Normalmente, esto es interpretado como la "señal".

- **Ruido de lectura:** Es una combinación de ruido proveniente del píxel y del Convertidor Analógico a Digital (ADC). El Ruido de Lectura (RN) del sensor es el nivel equivalente de ruido en la salida de la cámara en oscuridad y con tiempo de integración cero, o dicho de otro modo, el ruido generado al convertir la señal de analógico a digital. Este tipo de ruido no depende del tiempo de exposición ni de la irradiancia, como los otros dos. El ruido de lectura básicamente determina

la resolución del contraste que la cámara puede alcanzar. Cuanto menor es el nivel de ruido de lectura, menor es el número mínimo de electrones de señal que se pueden detectar. Un menor ruido de lectura significa que pueden detectarse cambios más pequeños en la amplitud de la señal, y por tanto, se detectan detalles con diferencias de contraste más pequeñas. Este ruido sigue una distribución normal,

$$N(\mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \quad (2.9)$$

donde  $\mu$  es la media y  $\sigma$  la desviación estándar de la señal.

Como los diferentes ruidos no están correlacionados, el ruido total será simplemente la suma de todos ellos. Pero, evidentemente, todo el ruido descrito ahora, depende de la señal de entrada. Para cuantificar la relación entre la señal de entrada y el ruido generado se define la Relación señal a ruido (SNR). Esta cantidad cuenta la relación entre la potencia de la señal y del ruido.

$$SNR = \frac{P_{signal}}{P_{noise}} \quad (2.10)$$

Esta SNR varía dependiendo de la señal, y por lo tanto, en una sola imagen, pueden existir múltiples valores de SNR, principalmente porque depende de la reflectividad de la superficie del objeto. Por eso, es difícil definir la SNR de una sola imagen.

### 2.2.2 Aberraciones ópticas

Las aberraciones ópticas son desviaciones de la propagación ideal de los rayos de luz a través de un sistema óptico. Estas desviaciones provocan que los rayos no converjan adecuadamente, provocando errores en la imagen. Las aberraciones se clasifican comúnmente en dos grandes grupos: aberraciones monocromáticas y aberraciones cromáticas. La principal diferencia entre ambas es la respuesta que tienen a diferentes colores.

Las aberraciones monocromáticas aparecen incluso si la luz utilizada es de un solo color (longitud de onda fija) y son principalmente las siguientes:

- **Aberración esférica:** Se produce cuando los rayos incidentes alejados del eje óptico se enfocan en diferentes puntos que los rayos cercanos al eje.
- **Coma:** Afecta a los rayos que inciden oblicuamente al sistema óptico, generando imágenes de puntos como manchas asimétricas.
- **Astigmatismo:** Provoca que los rayos en diferentes planos focales tengan distintos puntos de enfoque.
- **Curvatura de campo:** Hace que una imagen plana se proyecte en una superficie curva.
- **Distorsión:** Provoca un desplazamiento de las posiciones de los puntos de la imagen respecto a su posición ideal, sin afectar la nitidez.

En cambio, las cromáticas se originan debido a que el índice de refracción de los materiales ópticos depende de la longitud de onda de la luz. Los rayos de diferentes colores se enfocan en distintos puntos, dando lugar a halos o bordes coloreados en las imágenes.

### 2.2.3 Distorsión Óptica

La distorsión óptica altera la posición de los puntos de la imagen. Dada la geometría esférica de las lentes, la distorsión es, a grandes rasgos, una aberración radial. Es decir, la desviación de la posición ideal de los puntos depende de la distancia a la que se encuentren del centro de distorsión. A diferencia de otras aberraciones, no degrada directamente la nitidez de la imagen, pero sí modifica su forma. De forma general, se pueden diferenciar dos tipos de distorsión radial que corresponden a un modelado matemático análogo (en la siguiente sección se explica más en detalle): ***pincushion*** y ***barrel***.

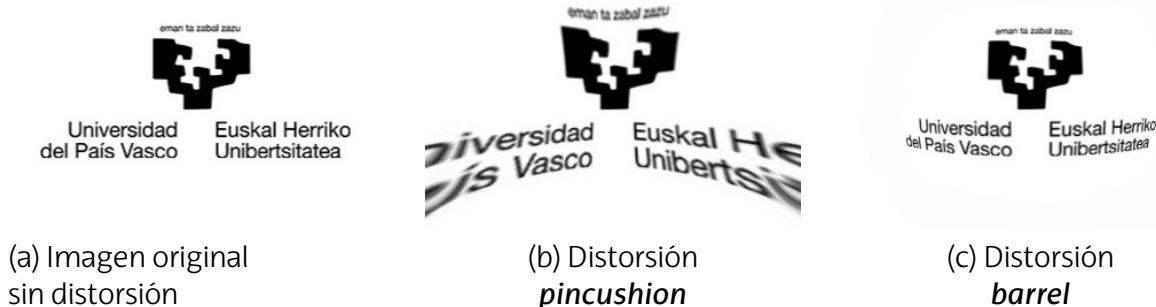


Figura 2.3: Comparación de los efectos de distorsión óptica

### 2.2.4 Modelado Matemático

En la actualidad, existen dos modelos dominantes en el contexto científico. El primero es el polinomio propuesto por Duane [4] que expresa la posición real (distorsionada) de cada punto  $X = (x, y)$  de la siguiente forma:

$$\begin{cases} x = (1 + k_1\|p\|^2 + k_2\|p\|^4 + k_3\|p\|^6 + \dots)u \\ y = (1 + k_1\|p\|^2 + k_2\|p\|^4 + k_3\|p\|^6 + \dots)v \end{cases} \quad (2.11)$$

donde  $k_1, k_2, k_3, \dots$  son los parámetros de distorsión radial (o coeficientes) y  $p = (u, v)$  son las coordenadas de píxel (ideales, sin distorsión) relativas al centro de distorsión del sistema. El otro modelo es una división propuesta por Fitzgibbon [5]:

$$\begin{cases} u = \frac{1}{(1 + \lambda_1\|X\|^2 + \lambda_2\|X\|^4 + \lambda_3\|X\|^6 + \dots)}x \\ v = \frac{1}{(1 + \lambda_1\|X\|^2 + \lambda_2\|X\|^4 + \lambda_3\|X\|^6 + \dots)}y \end{cases} \quad (2.12)$$

donde  $\lambda_1, \lambda_2, \lambda_3, \dots$  son los coeficientes del modelo.

Evidentemente, el nivel de precisión que se puede llegar a alcanzar con estos modelos es proporcional al número de coeficientes que se usen. En cualquier caso, para

la mayoría de aplicaciones donde la distorsión es mínima, basta con tener en cuenta únicamente el primer coeficiente.

En este contexto se diferencian dos formas de distorsión: ***barrel*** y ***pincushion*** (ver Figura 2.3), denominadas de esta forma por su similitud precisamente con un barril (en inglés, ***barrel***) y con un cojín (en inglés ***cushion***). Ambas, definidas por los modelos recién explicados, difieren exclusivamente en el signo de los coeficientes de cada modelo. En ambos modelos, coeficientes positivos darían lugar a una distorsión ***pincushion*** y negativos a una ***barrel***.

En el caso concreto que se estudia en este trabajo, el modelo utilizado es el propuesto por Duane de la Ecuación (2.11) y con un solo coeficiente (suficiente para cámaras profesionales) por motivos de optimización. Además, debido a la geometría esférica de la óptica, es razonable reducir las dos ecuaciones en una sola, de la siguiente forma:

$$R' = (1 + kR^2)R \quad (2.13)$$

donde se han eliminado los demás coeficientes del modelo,  $R = \|p\| = \sqrt{u^2 + v^2}$  y  $R' = \|X\| = \sqrt{x^2 + y^2}$ .

Reorganizando un poco la ecuación y definiendo la distorsión como  $d(u, v) = 1 + kR^2$ , se obtiene:

$$R' = R \cdot d \quad (2.14)$$

Esta ecuación será la que se utilizará en el modelo de detección de distorsión. En otras palabras, el método propuesto en este trabajo tiene como objetivo la consecución de un valor único del coeficiente:  $k_m$ . Cabe mencionar las limitaciones que muestra este modelo. En primer lugar, se presupone una distorsión exclusivamente radial y en segundo lugar, tal y como se muestra en (2.13) solo se tiene en cuenta un único coeficiente del desarrollo completo propuesto por Duane.

## 2.3 Métodos de corrección de distorsión

### 2.3.1 Métodos clásicos

Bukhari y Dailey [6] propusieron ajustar círculos en segmentos de líneas detectadas. En [7], los autores propusieron la Transformada de Hough Mejorada, que puede detectar líneas distorsionadas, y en [8], propusieron un método para corregir la distorsión radial usando la Transformada de Hough Mejorada (IHT) y un modelo de Fitzgibbon de un solo coeficiente. En [9], se propuso una extensión de la transformada de Hough modificada que puede optimizar iterativamente modelos de distorsión radial de dos parámetros, tanto polinomiales como de división.

### 2.3.2 Métodos de aprendizaje basados en regresión

Rong et al. [10] utilizaron un enfoque de clasificación para estimar el parámetro de distorsión con precisión finita como uno de 401 valores discretos posibles. López et al. [11] entrenaron exitosamente una red neuronal convolucional para estimar conjuntamente la orientación de la cámara (inclinación, balanceo) y los parámetros intrínsecos

(distancia focal y distorsión radial) a partir de imágenes individuales. Su método utilizó datos sintéticos generados a partir de imágenes panorámicas del conjunto de datos SUN360 presentado por Xiao et al. [12], que contiene objetos artificiales de la vida cotidiana. El método estimó dos parámetros de distorsión del modelo polinomial, expresando el parámetro  $k_2$  como función de  $k_1$ . Liao et al. [13] extendieron el marco propuesto por Rong et al. [10] añadiendo una nueva representación para la distorsión de lente llamada *Ordinal Distortion*.

### 2.3.3 Métodos de aprendizaje basados en reconstrucción

Se pueden encontrar varios métodos que abordan la corrección de distorsión como un problema de traducción de imagen a imagen. Liao et al. [14] utilizaron entrenamiento adversarial para entrenar una GAN que modela el mapeo entre imágenes distorsionadas y rectificadas. En los trabajos de Li et al. [15], Chao et al. [16] y Zhu et al. [17], los autores intentaron mejorar la interpretabilidad de las predicciones incorporando el **flujo de corrección**. Li et al. [15] ajustaron un modelo de división de un solo parámetro al flujo de corrección de baja resolución predicho, lo que permite calcular analíticamente la distorsión incluso para imágenes de alta resolución. En [16], los autores aprendieron un previo geométrico para mejorar la consistencia del flujo de corrección. En [17], los autores exploraron una arquitectura UNet basada en transformadores para la predicción del flujo de corrección. En [18], los autores proponen una red complementaria que contiene ramas paralelas para la predicción del flujo de corrección y la corrección de distorsión. En cada nivel del decodificador de distorsión, la estimación adecuada del flujo de corrección se utiliza para producir características no distorsionadas. En [19], los autores proponen un pre-entrenamiento consciente del modelo, donde un codificador inspirado en transformadores aprende a predecir el tipo de distorsión y los parámetros de distorsión. Después de completar el pre-entrenamiento, el decodificador de rectificación se ajusta para predecir el flujo de corrección.

## 2.4 Métricas

En este trabajo se propone estudiar un conjunto de coeficientes  $\{k_1, k_2, \dots, k_N\}$  asociado a un conjunto de posiciones de la imagen  $\{M_1, M_2, \dots, M_N\}$ , explicado más adelante en profundidad. La exactitud y precisión de este conjunto de datos se puede y se debe evaluar en base a unas métricas. En esta sección se proponen distintas cantidades que más adelante se utilizarán para validar el método de detección de distorsión.

A la hora de evaluar la precisión parece natural usar un parámetro que mida la diferencia entre el coeficiente de distorsión real y el medido, por ejemplo  $L = k_{real} - k_m$ . Sin embargo, para hacer una valoración más completa del perfil de distorsión medido, Liao et al. [13] proponen lo que ellos denominan como **mean distortion level deviation** (MDLD):

$$MDLD = \frac{1}{N} \sum_{i=1}^N |\hat{d}_i - d_i| \quad (2.15)$$

donde  $N$  es la cantidad de coeficientes a estudiar,  $\hat{d}_i$  la distorsión generada por el coeficiente  $k_i$  en la posición  $M_i$  y  $d_i$  la distorsión generada por el coeficiente  $k_{real}$  en la posición  $M_i$ . Esta cantidad (adimensional) mide el efecto práctico que produce el error

en la medida del coeficiente. En caso de querer recuperar la imagen original (sin distorsionar), cuanto mayor sea el MDLD, mayor será la diferencia absoluta entre la imagen distorsionada y la imagen sin "desdistorsionada".

Siguiendo con una propuesta más sencilla, se propone medir ***root mean square error*** (RMSE) entre los coeficientes reales y medidos[13].

$$RMSE = \frac{1}{N} \sum_{i=1}^N \sqrt{(k_i - k_{real})^2} \quad (2.16)$$

Esta medida se centra más en valorar la precisión del conjunto y se complementa con el ***error relativo*** común.

$$E_r = \left| \frac{k_m - k_{real}}{k_{real}} \right| \times 100 \quad (2.17)$$

En este caso  $k_m$  es el valor final que se mide con este método; por lo que, esta métrica es la más concluyente además de ser la más fácil de interpretar.

### 3 Método propuesto

El método propuesto en este trabajo se centra en el estudio de imágenes de un ***pin-hole grid***. A partir de adquisiciones reales y simuladas de los ***pinholes*** se podrá estimar el desajuste en las posiciones de estos y concluir con un perfil de distorsión del sistema óptico. A lo largo del trabajo se hará referencia a los pinholes sin distorsión como ***pinholes ideales*** y a los pinholes con distorsión como ***pinholes reales***. Además, es fundamental tener en cuenta algunos parámetros que se definen a continuación:

- *spacing*: distancia (en píxeles) entre pinholes ideales contiguos.
- *px*: *pixelsize*, dimensiones del píxel en metros.
- $n_{ph}$ : número de pinholes en cada dimensión,  $n_{px} \times n_{py}$ .
- $C_o: (C_x, C_y)$  centro óptico del sistema. Por simetría esférica de la óptica del sistema, este coincidirá con el centro de distorsión. Se dará generalmente en dimensiones de píxel.
- $P_i$ : posición del centroide del  $i$ -ésimo pinhole ideal en píxeles,  $(P_{i_x}, P_{i_y})$ .
- $M_i$ : posición del centroide del  $i$ -ésimo pinhole real en píxeles,  $(M_{i_x}, M_{i_y})$ .

#### 3.1 Primer paso: detección de los centroides de los pinholes

La primera etapa del método es detectar los centroides de cada pinhole en la imagen, dando su posición en píxeles:  $P_i = (P_{i_x}, P_{i_y})$ . Para detectar estos pinholes se ha propuesto un algoritmo con umbral adaptativo en el que se divide la imagen en ventanas que contengan únicamente un pinhole. Despues, se binariza la ventana en regiones con el umbral adaptativo y se escoge la región más extensa (esta será la región del pinhole). De esta forma se descartan píxeles brillantes fuera del pinhole. Finalmente, se calcula el centro de masa de la región del pinhole y se define este como el centroide

del pinhole [20].

Para dividir la imagen en ventanas se utiliza una máscara de máximos relativos con un umbral mínimo  $U_{min}$ , de esta forma se genera un primer conjunto de candidatos a pinhole. Esta primer conjunto de puntos  $\{Q_i\}$  serán una mezcla entre los máximos de los pinholes y ruido electrónico en puntos aleatorios de la imagen. Para eliminar el ruido se evalúan los píxeles vecinos para cada  $Q_i$ , eliminando los puntos que tengan menos de 4 vecinos con intensidad menor que el 40% de la intensidad de  $Q_i$ . Este umbral depende mucho del rango del ruido de las imágenes y, aunque, los valores utilizados en este trabajo son generales, se recomienda adaptarlos a cada caso. Este nuevo conjunto de puntos  $\{Q'_i\}$  serán los centros de las ventanas.

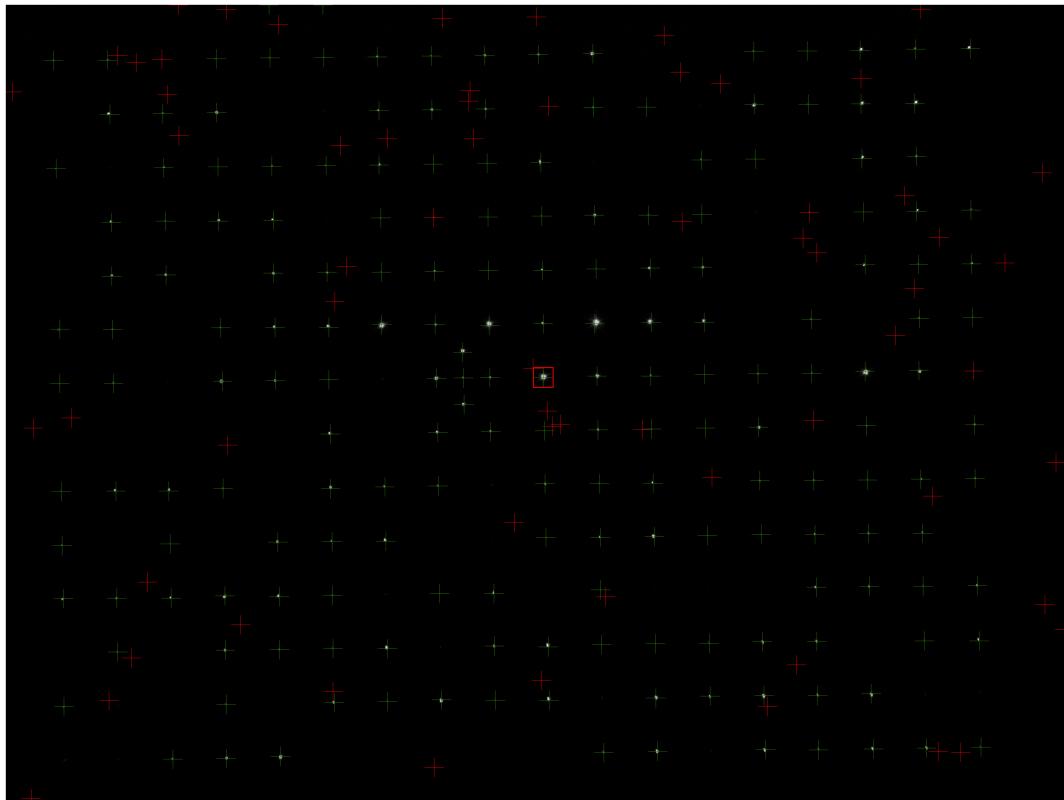


Figura 3.4: Imagen de un pinhole grid proporcionada por Satlantis Microsats. Representados el conjunto de puntos  $\{Q_i\}$  (cruces rojas y verdes) y  $\{Q'_i\}$  (cruces verdes). Dibujada una ventana en rojo centrada en un punto  $Q'_i$  aleatorio.

Para binarizar cada ventana el umbral adaptativo que se propone en [20] es el  $n$ -ésimo valor máximo de la intensidad en cada ventana. Sin embargo, esto puede acarrear problemas si los tamaños de los pinholes no son iguales, así que se propone otro umbral más acorde a lo que se puede ver en la Figura 3.4:

$$I_{umbral} = I_{max} \times 0.4 \quad (3.1)$$

donde  $I_{max}$  es la intensidad máxima de la ventana. Esto hace que sea cual sea el tamaño del pinhole se pueda binarizar con sentido.

Una vez binarizada la ventana, podría suceder que algún píxel no perteneciente al pinhole superase el umbral. Para evitar tenerlos en cuenta a la hora de calcular el centroide, se realiza el cálculo solo sobre la región de puntos más extensa.

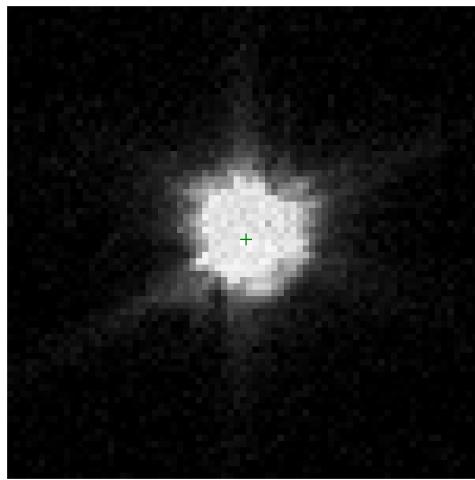
Las ecuaciones con las que se calcula el centroide en cada ventana son las siguientes:

$$M_{i_x} = \frac{\sum_{j=1}^U \sum_{k=1}^V (I(j, k) - I_{umbral}) H(j, k) x_{jk}}{\sum_{j=1}^U \sum_{k=1}^V (I(j, k) - I_{umbral}) H(j, k)} \quad (3.2)$$

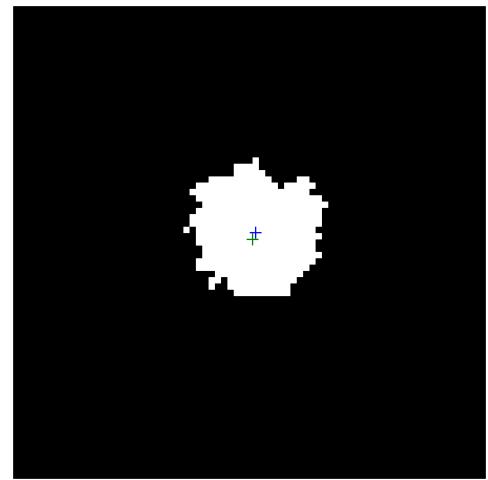
$$M_{i_y} = \frac{\sum_{j=1}^U \sum_{k=1}^V (I(j, k) - I_{umbral}) H(j, k) y_{jk}}{\sum_{j=1}^U \sum_{k=1}^V (I(j, k) - I_{umbral}) H(j, k)} \quad (3.3)$$

$$H(j, k) = \begin{cases} 1, & \text{si } I(j, k) \geq I_{umbral} \\ 0, & \text{si } I(j, k) < I_{umbral} \end{cases} \quad (3.4)$$

donde  $M_{i_x}$  y  $M_{i_y}$  son las coordenadas del centroide del  $i$ -ésimo pinhole  $M_i$ .  $I(j, k)$  representa el valor de intensidad del píxel  $(j, k)$ , mientras que  $I_{umbral}$  es el valor umbral de intensidad. Las variables  $x_{jk}$  e  $y_{jk}$  denotan las coordenadas del píxel  $(j, k)$  en la imagen completa de pinholes. Por último,  $U$  y  $V$  representan el número de píxeles en las direcciones  $X$  e  $Y$ , respectivamente, dentro de cada ventana de detección.



(a) Ventana ampliada de la Figura 3.4.



(b) Ventana ampliada de la Figura 3.4 binarizada con umbral adaptativo  $I = I_{max} \times 0.4$ . Marcado en azul el centroide medido  $M_i$ .

Como se puede observar en la Figura 3.5b el conjunto de puntos  $\{Q'_i\}$  no coincide con el conjunto  $\{M_i\}$ . Por eso es necesario realizar este segundo paso de binarizar la imagen para refinar el cálculo del centroide.

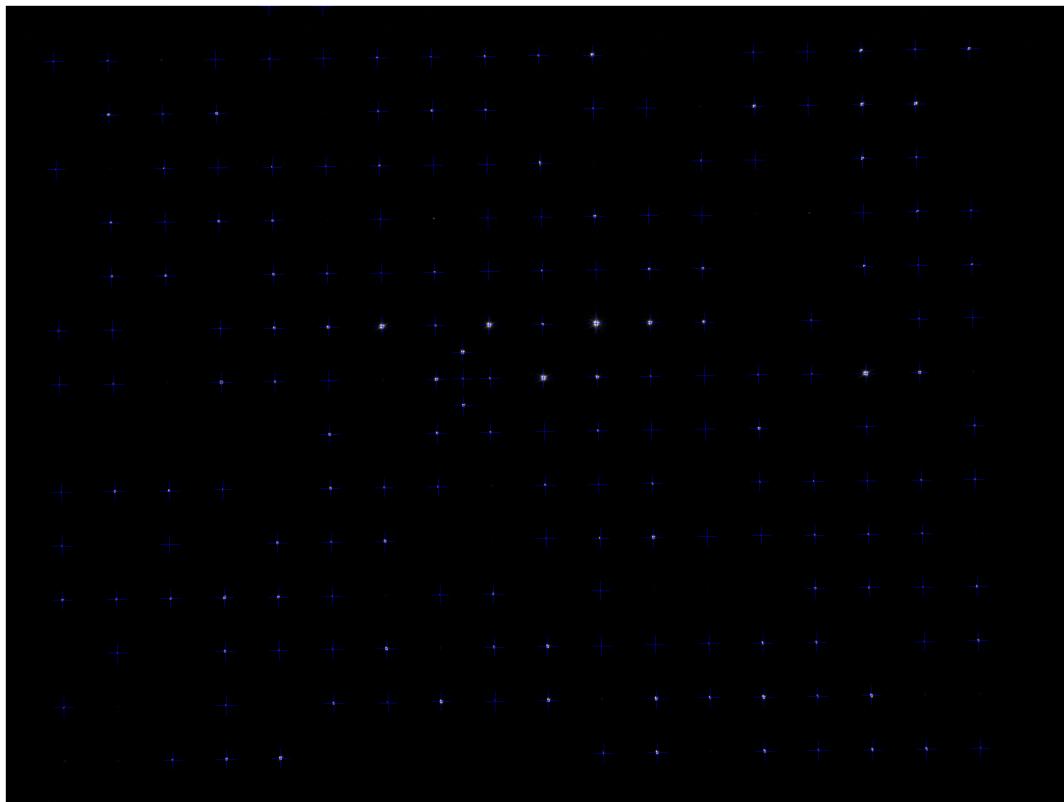


Figura 3.6: Imagen de un pinhole grid proporcionada por Satlantis Microsats. Representados el conjunto de puntos  $\{M_i\}$ , es decir, los centroides de los pinholes.

### 3.2 Segundo paso: caracterización de la placa de pinholes, y modelización de las pinholes ideales

En el caso de cámaras profesionales como es este, la distorsión medida será de apenas unos pocos píxeles en los pinholes más alejados del centro óptico; por lo que, es esencial realizar un modelo de las posiciones de los pinholes ideales (sin distorsión) excelente. Para ello, en este método se evalúa y modeliza: la orientación de la placa, su posición y posibles pinholes que la cámara no haya podido detectar por errores de fabricación de la placa. Se seguirán los siguientes pasos: creación de una malla de pinholes regular, determinar su orientación y girarla, calcular su desplazamiento y corregirlo, emparejar las posiciones  $M_i$  con las  $P_i$ .

Para calcular el ángulo de orientación de la placa se propone estudiar los pinholes reales más cercanos al centro óptico. En concreto, se propone estudiar los centros que distan hasta  $3 \times \text{spacing}$  desde  $C_o$  para que tengan la mínima distorsión y haya suficientes. Emparejando a pares los pinholes contiguos y estudiando los vectores que los unen se obtienen los ejes de orientación de la placa (ver Figura 3.7).

Para calcular el desplazamiento se va a optar por suponer que la posición del pinhole real mas cercano al  $C_o$  coincide con la posición de su pinhole ideal correspondiente. Es decir, si el  $j$ -ésimo pinhole real es el más cercano a  $C_o$ :  $M_j = P_j$ , salvando una distancia de  $\Delta C_o$ :

$$\Delta C_o = \|M_j - C_o\| \quad (3.5)$$

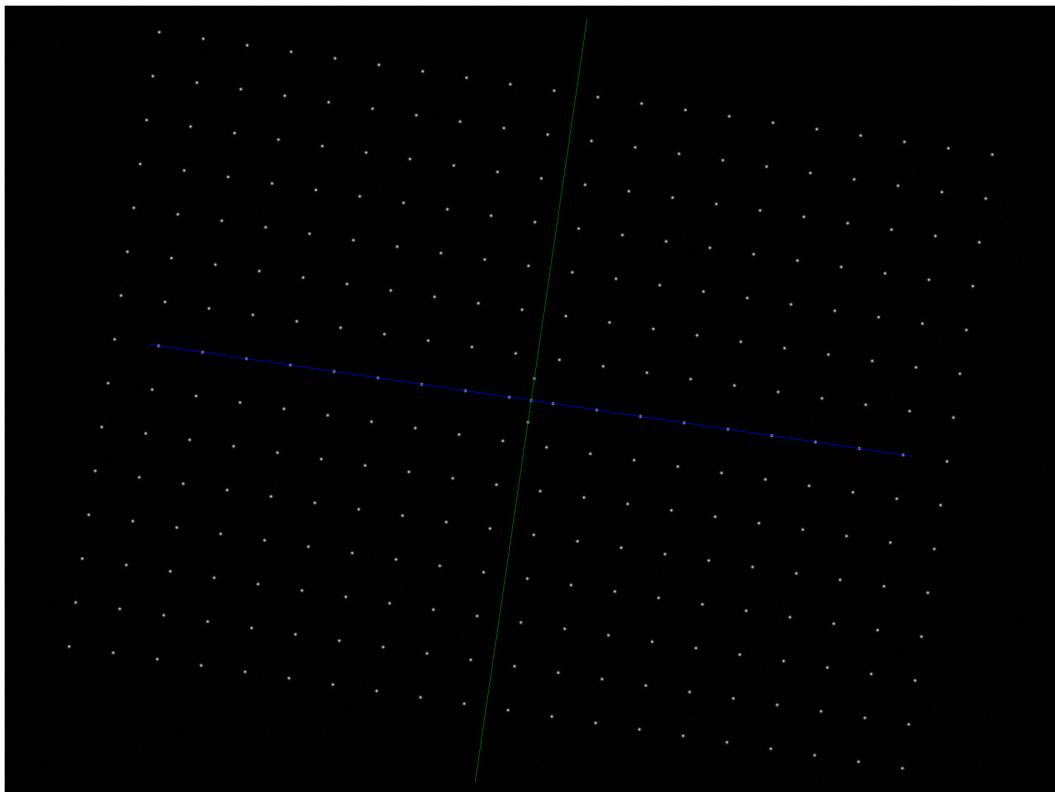


Figura 3.7: Ejes principales de la placa calculados.

Esta suposición parece ser válida solo si la distorsión es pequeña y si se ha detectado un pinhole lo suficientemente cerca del  $C_o$ . Más adelante se verificará o no.

En este momento surge un problema de cómo emparejar cada pinhole real con su correspondiente ideal. Se ha propuesto utilizar el método de asignación lineal propuesto por Kuhn, comúnmente conocido como método húngaro [21]. Este algoritmo empareja cada  $M_i$  con un único  $P_i$  (por lo que no hay problema si hay pinholes no detectados) de modo que se minimice la suma de todas las distancias  $\|P_i - M_i\|$ . En la práctica funciona porque la distorsión es pequeña y el pinhole ideal más cercano a un pinhole real, es precisamente su pinhole correspondiente. Si la distorsión fuera más grande, la asignación fallaría; ya que, podría ocurrir que un pinhole real tuviera más cerca otro pinhole ideal que su correspondiente pinhole ideal. Para su implementación se utiliza la función `linear_sum_assignment` de la librería SciPy [22], que proporciona una solución óptima y computacionalmente eficiente.

### 3.3 Tercer paso: cálculo de la constante de distorsión.

En este paso se calcula la constante de distorsión mediante un ajuste de mínimos cuadrados. Se tienen  $N$  pares de puntos ideal-real:  $(P_i, M_i)$  que obedecen a (2.13) que, si se reordena queda:

$$R' - R = kR^3 \quad (3.6)$$

donde  $R(i) = \sqrt{M_{i_x}^2 + M_{i_y}^2}$  y  $R'(i) = \sqrt{P_{i_x}^2 + P_{i_y}^2}$ .

Definiendo:

$$y_i = R'(i) - R(i) = \sqrt{M_{i_x}^2 + M_{i_y}^2} - \sqrt{P_{i_x}^2 + P_{i_y}^2}$$

$$x_i = R^3 = \left( \sqrt{M_{i_x}^2 + M_{i_y}^2} \right)^{\frac{3}{2}}$$

la constante queda como:

$$k_m = \frac{\sum_{i=0}^n x_i y_i}{\sum_{i=0}^n x_i^2} \quad (3.7)$$

con un error de dispersión definido por:

$$R^2 = 1 - \frac{\sum_{i=0}^n (y_i - y'_i)^2}{\sum_{i=0}^n (y_i - \bar{y}_i)^2} \quad (3.8)$$

donde  $y'_i = k_m \cdot x_i$  es el valor ajustado a la medida y  $\bar{y}_i$  es la media de valores  $y_i$ . Se interpreta de la siguiente manera: si  $R^2 = 1$  significa que el modelo explica toda la variabilidad (error de dispersión cero). Sin embargo, si  $R^2 = 0$  quiere decir que el modelo explica la misma variabilidad que la media (el modelo no ajusta mejor que una línea horizontal).

A este proceso se le añade un filtro de *outliers* utilizando el método **Z-Score** [23]. Este método consiste en normalizar cada muestra con respecto a la media y la desviación típica del conjunto de datos, de acuerdo con la siguiente expresión:

$$z_i = \frac{x_i - \bar{x}_i}{\sigma}, \quad (3.9)$$

donde  $\bar{x}_i$  es la media del conjunto y  $\sigma$  su desviación típica.

Aquellos valores cuyo  $|z_i|$  supera un umbral predefinido (comúnmente se utiliza como umbral 3) se consideran atípicos y se descartan del conjunto de datos. Es un modelo ampliamente utilizado en estadística para la identificación de outliers.

## 4 Etapa de simulación

Los objetivos principales de esta etapa se basan en el desarrollo y validación del método propuesto en la sección previa. Para ello, se ha desarrollado un entorno de simulación que permite generar imágenes de los pinhole grids bajo condiciones controladas. Esta etapa se ha subdividido en 4 subetapas con diferentes funciones:

- Generación de una imagen de pinholes grid ideal, sin distorsión.
- Introducción de la distorsión radial en la imagen ideal.
- Aplicación del método propuesto para la detección de la distorsión.
- Evaluación de los resultados obtenidos.

Todas las imágenes generadas en las simulaciones se almacenan en formato FITS. Este formato de archivos es comúnmente utilizado para procesamiento de imágenes. Además, las imágenes se han generado en 12 bits en grayscale, por lo que, las imágenes computacionalmente son arrays bidimensionales con valores (intensidades) entre 0 y 4095.

## 4.1 Generación de la imagen ideal

En esta etapa se generan imágenes de los pinholes sin distorsión. Las simulaciones se han diseñado para reproducir de forma fiel las imágenes reales proporcionadas por Satlantis Microsats, como se muestra en la Figura 4.8.

De todos los parámetros que constituyen la imagen, se han diferenciado dos tipos: las fijas y las variables. Los parámetros fijos permanecen constantes en todas las simulaciones y son los siguientes:

- $px$ : pixelsize, medida de un píxel en metros.
- $sendim$ : dimensiones de la imagen en píxeles.
- $n_{ph}$ : número de pinholes en cada dimensión  $n_{px} \times n_{py}$ .
- $spacing$ : distancia entre pinholes

A partir de estos parámetros se calculan las posiciones teóricas de los centros de los pinholes, asegurando un espaciado uniforme.

En cambio, los parámetros variables serán aquellos que irán variando y que permiten simular las incertidumbres presentes en un sistema real:

- $O$ : centro de la placa de pinholes. Se situará en el centro de la imagen con una incertidumbre de  $\delta h, \delta w$  en cada dimensión.
- $C_o$ : centro óptico. Se situará en un punto concreto con una incertidumbre de  $\delta cx, \delta cy$  en cada dimensión.
- $I_i$ : Intensidad de cada pinhole,  $I_i \in [I_{min}, I_{max}]$ .
- $\theta$ : rotación global de los pinholes, acotada por un ángulo máximo  $\theta_{max}$ .

La etapa comienza calculando los centros de los pinholes ideales, estos se modelan como fuentes puntuales dándoles una intensidad aleatoria  $I_i \in [I_{min}, I_{max}]$ . Con esto se simula una posible falla del sistema óptico o de la placa de pinholes (pinholes demasiado pequeños) en la que no se detectan algunos pinholes de forma clara. Esto es análogo a la imagen ideal  $f(x, y)$  en la Ecuación 2.2; por lo que, obteniendo la PSF o  $h(x, y)$  del sistema se podría obtener la imagen resultante convolucionando ambas funciones (ver Figura 2.1). En esta simulación, la PSF del sistema se aproxima mediante un disco de Airy, representativo de un sistema óptico limitado por difracción.

Al mismo tiempo, se aplica un desenfoque suave mediante un filtrado gaussiano para reproducir imperfecciones del sistema óptico y efectos de desenfoque residual. Para finalizar con la imagen ideal, se rota un ángulo  $\theta$  y se desplaza la placa entera para descentrarla una distancia  $(\delta h, \delta w)$ .

## 4.2 Introducción de la distorsión radial en la imagen ideal

Se aplica la distorsión caracterizada por un único coeficiente de distorsión  $k_{real}$ . Dicho coeficiente se selecciona aleatoriamente dentro de un rango razonable en este contexto, definido de forma que el desplazamiento máximo inducido en los pinholes más alejados del centro no supere unos pocos píxeles. La distorsión se aplica utilizando un remapeo que permite deformar la imagen ideal y obtener la imagen distorsionada final.

Como pequeño apunte, es después de distorsionar la imagen cuando se añade ruido (gaussiano) para simular el ruido electrónico del sensor. Esto es coherente ya que, la distorsión es un efecto geométrico de la lente, mientras que el ruido es un defecto de muestreo del sensor 2.2.1. Añadir el ruido al final simula el proceso real donde el sensor captura la imagen ya distorsionada.

Como resultado de esta etapa se obtienen, para cada simulación, tanto la imagen ideal como la imagen distorsionada (ambas con el mismo ruido), junto con los parámetros reales utilizados en la generación, lo que permite una evaluación precisa del método propuesto.

Analizando las imágenes (Figuras 4.8 y 4.9) se observa que la única diferencia notable es la diferencia entre tamaños de pinholes. Mientras que en la imagen simulada el tamaño de los pinholes es siempre el mismo, en las reales el tamaño varía. Por motivos de optimización, se ha optado por mantener de esta forma las simulaciones, pero se tiene en cuenta a la hora de calcular el umbral adaptativo 3.1

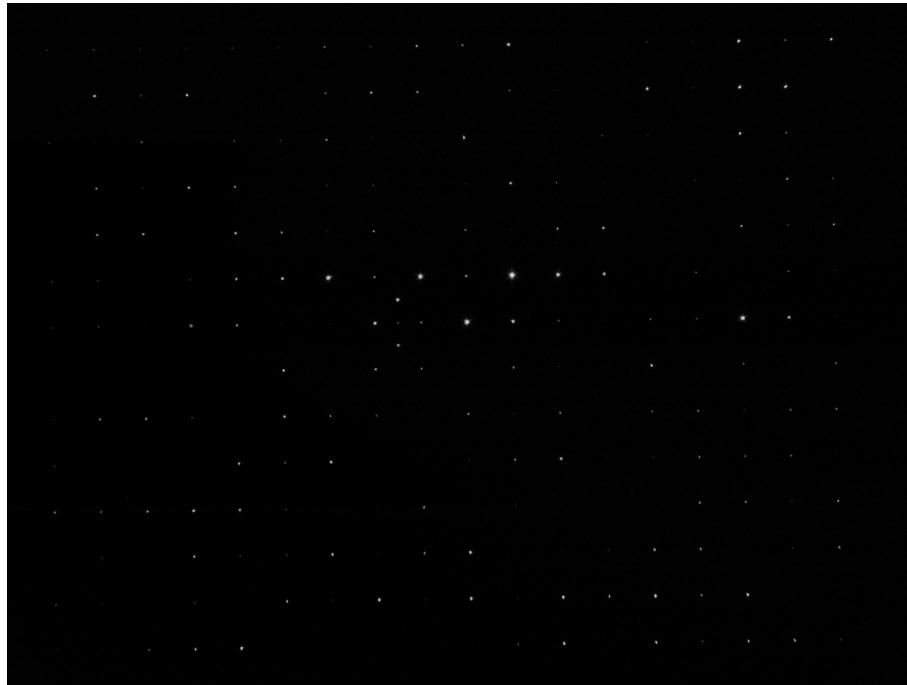


Figura 4.8: Imagen de un pinhole grid real hecha con xxxx.

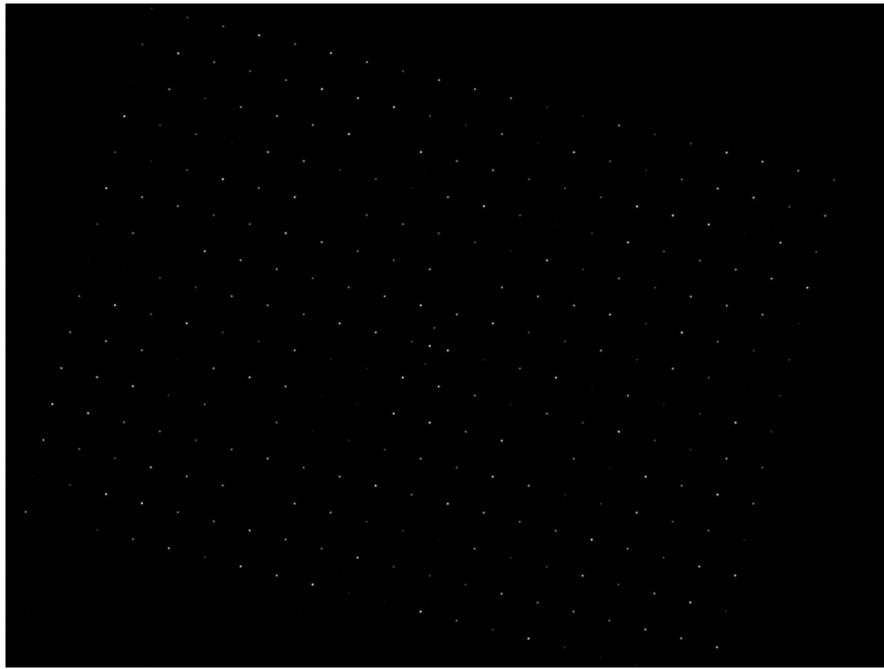


Figura 4.9: Imagen de un pinhole grid simulada.

### 4.3 Aplicación del método propuesto para la detección de la distorsión

En esta etapa se aplica el método propuesto a las imágenes simuladas. Para hacer un estudio completo y ver cuándo y cómo funciona mejor este método, se propone estudiar los parámetros variables citados en 4.1. Se han realizado tiradas de 100 simulaciones con las siguientes condiciones:

- Simulación 1: todas las variables controladas:  $\theta = 0$ ,  $C_o$  centrado en un pinhole,  $O$  centrado en la mitad de la imagen,  $I_i$  lo suficientemente grande como para que se detecten todos los pinholes y misma distorsión  $k_{real}$ .
- Simulación 2: todas las variables controladas menos la distorsión  $k_{real}$ .
- Simulación 3: todas las variables controladas menos el centro óptico  $C_o$  centrado en un pinhole.
- Simulación 4: todas las variables controladas menos la rotación  $\theta$ .
- Simulación 5: todas las variables controladas menos la intensidad  $I_i$ .
- Simulación 6: ninguna variable controlada.

Además, se ha hecho un estudio sobre cómo afecta al resultado un posible error en la medida del parámetro fijo *spacing*:

- Simulación 7: todas las variables controladas pero error en la medida del *spacing*.
- Simulación 8: ninguna controlada y error en la medida del *spacing*

## 4.4 Evaluación de los resultados obtenidos.

### 4.4.1 Simulación 1

En esta tirada de simulaciones se han fijado los parámetros de la Tabla 1.

Tabla 1: Parámetros utilizados en la simulación 1.

Parámetro	Valor
Coeficiente de distorsión $k_{\text{real}}$	$15.8145 \text{ m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $\text{sen\_dim}$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	[1536, 2048] px
Centro óptico $C_o$	$O + [0, 258] \text{ px}$
Intensidad de pinhole $I_i$	3000
Ángulo de rotación $\theta$	0

Tabla 2: Resultados medios de la simulación 1 en base a las métricas propuestas de la.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	$15.8116 \text{ m}^{-2}$
Error en el centro óptico $\Delta C_o$	3.5 px
MDLD	$2.1077 \times 10^{-9}$
RMSE	$1.9906 \times 10^{-7}$
$E_r$	0.3734
$R^2$	0.9989

Los resultados mostrados en la Tabla 2 demuestran un excelente funcionamiento del método propuesto. En condiciones controladas el perfil de distorsión medido es totalmente fiable; sin embargo, conseguir este marco de parámetros controlados es muy complicado en la práctica. Es por ello que se han realizado más simulaciones con el fin de determinar en qué condiciones es fiable este método.

### 4.4.2 Simulación 2

Se observa en la Tabla 4 como no funciona tan bien como en el contexto anterior, evidentemente. Aún así, los resultados arrojados siguen siendo muy fiables.

De la Figura 4.10 se puede estimar un intervalo muy fiable de coeficientes de distorsión  $|k_{\text{real}}| \in (5, \infty) \text{ m}^{-2}$ , en el que el error relativo es menor a 5. Aunque, como ya se ha comentado previamente, una distorsión (en valor absoluto) muy grande puede dar otros problemas como por ejemplo, en la asignación de puntos  $(M_i, P_i)$ .

En la Figura 4.11 se muestra la enorme diferencia que existe en las medidas si se comparan coeficientes de distorsión más altos y más bajos en valor absoluto. Los coeficientes más pequeños distorsionan menos los pinholes; en consecuencia, el error que

Tabla 3: Parámetros utilizados en la simulación 2.

Parámetro	Valor
Coeficiente de distorsión $k_{\text{real}}$	$\in [-39, 5364, 39, 5364] \text{m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $sen\_dim$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	[1536, 2048] px
Centro óptico $C_o$	$O + [0, 258] \text{ px}$
Intensidad de pinhole $I_i$	3000
Ángulo de rotación $\theta$	0

Tabla 4: Resultados medios de la simulación 2 en base a las métricas propuestas.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	-
Error en el centro óptico $\Delta C_o$	3.5 px
MDLD	$1.8935 \times 10^{-9}$
RMSE px	$1.91415 \times 10^{-7}$
$E_r$	1.1194
$R^2$	0.9798

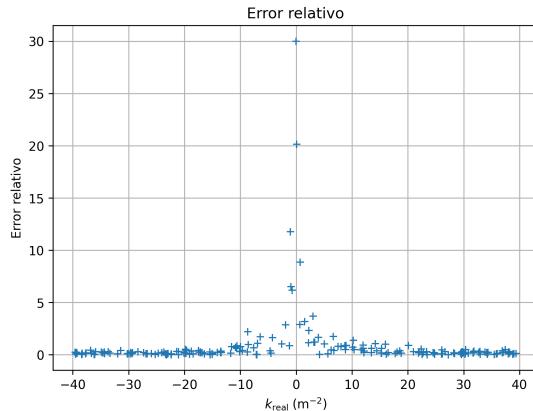
existe en la etapa de detección de centroides del algoritmo es más notorio y por eso se ven más dispersos los puntos del ajuste.

#### 4.4.3 Simulación 3

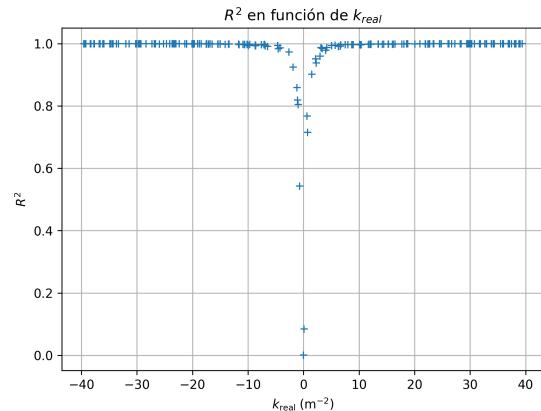
Tabla 5: Parámetros utilizados en la simulación 3.

Parámetro	Valor
Coeficiente de distorsión $k_{\text{real}}$	$19.7682 \text{m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $sen\_dim$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	[1536, 2048] px
Centro óptico $C_o$	$O + [0, 258] \text{ px}$
Intensidad de pinhole $I_i$	$\in [0, 3000]$
Ángulo de rotación $\theta$	0

Se observa en la Tabla 6 un aumento notorio en  $\Delta C_o$  y esto tiene sentido. En alguna simulación puede ocurrir que el pinhole fijo (aquel que se aproxima como centro óptico) no haya sido detectado. En ese caso el pinhole fijo pasa a ser el siguiente más cercano que, como cerca estará a  $spacing = 172$  píxeles de distancia. Esto quiere decir que  $\Delta C_o$  ha cogido valores de  $\approx 0$  y  $\approx 172$ , lo que resulta en la media mostrada en la Tabla 6. No obstante, se obtiene un  $E_r$  medio muy bajo que significa que el método de detección funciona correctamente bajo esta condición.

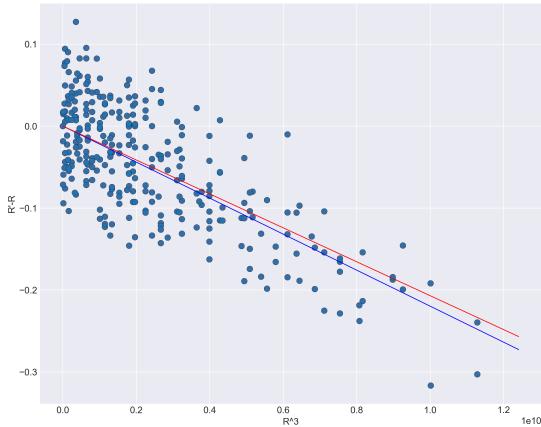


(a) Error relativo  $E_r$  cometido en la medición en función del coeficiente de la distorsión real.

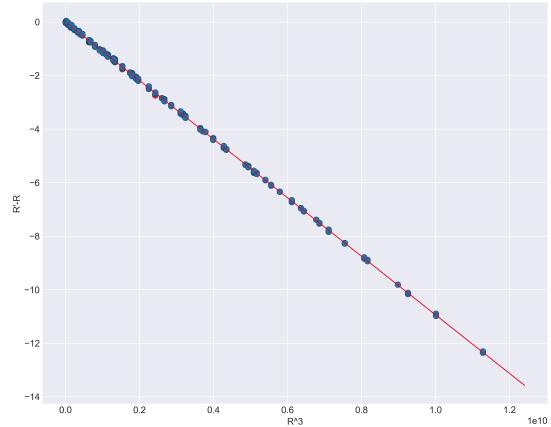


(b) Error del ajuste lineal  $R^2$  cometido en la medición en función del coeficiente de la distorsión real.

Figura 4.10: Errores en la medida en función del coeficiente de distorsión en la simulación 2.



(a) Representados los valores del ajuste 3.7 calculados para  $k_{real} = -0.6844\text{m}^{-2}$ ,  $R^2 = 0.5432$  y  $E_r = 6.1800$ .



(b) Representados los valores 3.7 calculados para  $k_{real} = -36.1677\text{m}^{-2}$ ,  $R^2 = 0.9999$  y  $E_r = 0.0371$ .

Figura 4.11: Se observa la diferencia en la precisión y exactitud de la medida para  $k_{real}$  grande y pequeño en valor absoluto.

Tabla 6: Resultados medios de la simulación 3 en base a las métricas propuestas.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	$19.7457\text{m}^{-2}$
Error en el centro óptico $\Delta C_o$	3.5 61.9480 px
MDLD	$4.5937 \times 10^{-9}$
RMSE $px$	$4.5773 \times 10^{-6}$
$E_r$	0.6502
$R^2$	0.9957

#### 4.4.4 Simulación 4

Tabla 7: Parámetros utilizados en la simulación 4.

Parámetro	Valor
Coeficiente de distorsión $k_{real}$	$19.7682\text{m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $sen\_dim$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	[1536, 2048] px
Centro óptico $C_o$	$O + [\pm 100, 258 \pm 100] \text{ px}$
Intensidad de pinhole $I_i$	3000
Ángulo de rotación $\theta$	0

Tabla 8: Resultados medios de la simulación 4 en base a las métricas propuestas.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	$19.7502\text{m}^{-2}$
Error en el centro óptico $\Delta C_o$	3.5 76.4961 px
MDLD	$2.0860 \times 10^{-9}$
RMSE $px$	$1.8843 \times 10^{-7}$
$E_r$	0.2944
$R^2$	0.9994

La Tabla 8 muestra resultados muy buenos. Esto no hace más que reforzar la suposición inicial de fijar un el pinhole más cercano al centro óptico. Sin embargo, esto puede dar errores si los pinholes más cercanos, desafortunadamente, no son lo suficientemente brillantes para que el algoritmo los detecte.

#### 4.4.5 Simulación 5

En estas simulaciones se vuelve a ver el aumento de  $\Delta C_o$ , esta vez debida a desalineamiento de la placa de pinholes con el centro óptico. Sin embargo, esto no influye casi en los resultados. Esto tiene mucho sentido; ya que, se ha modelado la distorsión exclusivamente radial, lo que quiere decir que las distancias  $\|M_i - C_o\| = R$  no van a variar y en consecuencia  $k_{real}$  tampoco (2.13).

Tabla 9: Parámetros utilizados en la simulación 5.

Parámetro	Valor
Coeficiente de distorsión $k_{\text{real}}$	$19.7682 \text{ m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $\text{sen\_dim}$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	$[1536, 2048] \text{ px}$
Centro óptico $C_o$	$O + [0, 258] \text{ px}$
Intensidad de pinhole $I_i$	3000
Ángulo de rotación $\theta$	$\in [-15, 15]$

Tabla 10: Resultados medios de la simulación 5 en base a las métricas propuestas.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	$19.7657 \text{ m}^{-2}$
Error en el centro óptico $\Delta C_o$	3.5 px
MDLD	$1.8737 \times 10^{-9}$
RMSE $px$	$2.0017 \times 10^{-6}$
$E_r$	0.2653
$R^2$	0.9993

#### 4.4.6 Simulación 6

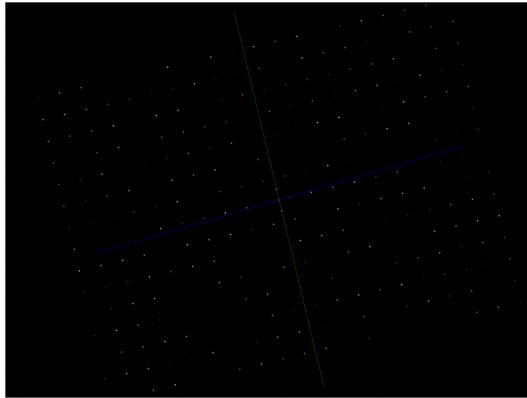
Tabla 11: Parámetros utilizados en la simulación 6.

Parámetro	Valor
Coeficiente de distorsión $k_{\text{real}}$	$ k_{\text{real}}  \in [5.93, 31.63] \text{ m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $\text{sen\_dim}$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	$[1536, 2048] + [\pm 100, \pm 100] \text{ px}$
Centro óptico $C_o$	$O + [\pm 100, 258 \pm 100] \text{ px}$
Intensidad de pinhole $I_i$	$\in [0, 3000]$
Ángulo de rotación $\theta$	$\in [-15, 15] \text{ rad}$

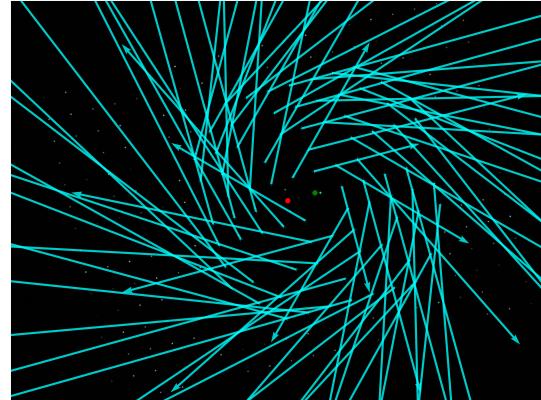
Este es un escenario bastante realista y los resultados de la Tabla 12 muestran que el algoritmo es bastante fiable en este contexto, lo cual es muy positivo. Además, se ha observado como al mezclar rotación y eliminación de pinholes, el algoritmo es más propenso a fallar en el cálculo de la rotación, dando resultados como el mostrado en la Figura 4.12. Estos **outliers** pueden eliminarse de manera más o menos sencilla porque se identifican muy fácil, por lo que no suponen una gran preocupación.

Tabla 12: Resultados medios de la simulación 6 en base a las métricas propuestas.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	-
Error en el centro óptico $\Delta C_o$	3.593 px
MDLD	$4.1747 \times 10^{-9}$
RMSE $px$	$4.2894 \times 10^{-7}$
$E_r$	0.7586
$R^2$	0.9926



(a) Ejes principales de la placa medidos por el algoritmo. Se aprecia la mala medida de la orientación por la falta de pinholes detectados.



(b) Mapa de distorsión donde las flechas (zoom x100) representan el desplazamiento de los pinholes  $M_i - P_i$ . Los puntos rojo y verde son el centro de la placa y óptico respectivamente. Se aprecian las desviaciones de los pinholes debidas a la mala orientación de los pinholes ideales.

Figura 4.12: Representaciones asociadas a simulación 6 con parámetros:  $k_{real} = 29.9392\text{m}^{-2}$  y  $\theta = -0.2351 \text{ rad}$

#### 4.4.7 Simulacion 7

Estos resultados son muy claros, medir mal el **spacing** repercute en un aumento del  $E_r$  y del MDLD sobre todo. Este error, aún por muy pequeño que sea (0.01% del **spacing** en estas simulaciones) empaña los resultados notoriamente. Esto tiene mucho sentido; ya que, en la segunda etapa del algoritmo 3.2 el modelo de los pinholes ideales se crea de forma que se va propagando y multiplicando el error hacia los pinholes más alejados del centro óptico. De esta forma el algoritmo mide una distorsión falsa provocada por el mal modelo de los pinholes ideales.

Esta es, sin duda, la condición más sensible que tiene este algoritmo. Otros autores como Wang et al. [24] proponen distintos mecanismos para calibrar, alinear y medir la distorsión que se podrían implementar en un caso práctico de detección de distorsión.

Tabla 13: Parámetros utilizados en la simulación 7.

Parámetro	Valor
Coeficiente de distorsión $k_{\text{real}}$	$19.7682 \text{ m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $\text{sen\_dim}$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	[1536, 2048] px
Centro óptico $C_o$	$O + [0, 258] \text{ px}$
Intensidad de pinhole $I_i$	3000
Ángulo de rotación $\theta$	0 rad

Tabla 14: Resultados medios de la simulación 7 en base a las métricas propuestas.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	$18.3018 \text{ m}^{-2}$
Error en el centro óptico $\Delta C_o$	3.5 px
MDLD	$5.1698 \times 10^{-8}$
RMSE $px$	$4.0255 \times 10^{-7}$
$E_r$	7.4181
$R^2$	0.9965

#### 4.4.8 Simulacion 8

Los resultados obtenidos en 16 muestran que el error cometido en la medida es pequeño. Evidentemente, ha aumentado respecto a otras condiciones más controladas pero, un error relativo  $E_r = 8.3555$  es lo suficientemente pequeño para considerar esta metodología existosa en todas las condiciones presentadas.

Como pequeña conclusión, los parámetros más sensibles y los que más hay que cuidar para que el algoritmo arroje resultados fiables son el *spacing* y el brillo de los pinholes.

## 5 Aplicación en sistemas ópticos de Satlantis

Aunque gran parte del trabajo desarrollado en este proyecto se ha basado en simulaciones, el objeto del trabajo requiere un contexto realista. En esta sección se ha puesto a prueba con imágenes satelitales hechas por Satlantis Microsats. En concreto, se ha evaluado una tirada de 10 imágenes.

La Figura 5.13 muestra una captura en concreto de las que se han analizado. Como se puede observar la imagen en bruto es mala, cuenta con demasiado ruido y sería bastante complicado que el algoritmo detectara los pinholes con precisión. Para eliminar este exceso de ruido se procesan las imágenes utilizando un *Dark Frame* (5.14) tal y como se ha explicado en la Sección 2.2.1. La imagen procesada (5.15) es el resultado de restar el *Dark Frame* a la imagen en bruto.

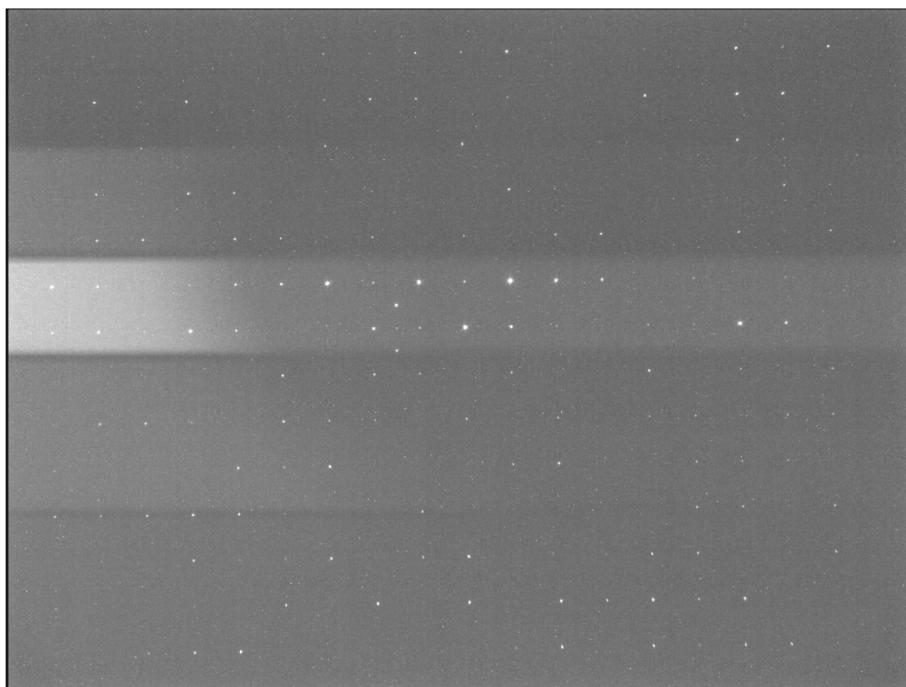


Figura 5.13: Imagen en bruto de un pinhole grid hecha con xxx.

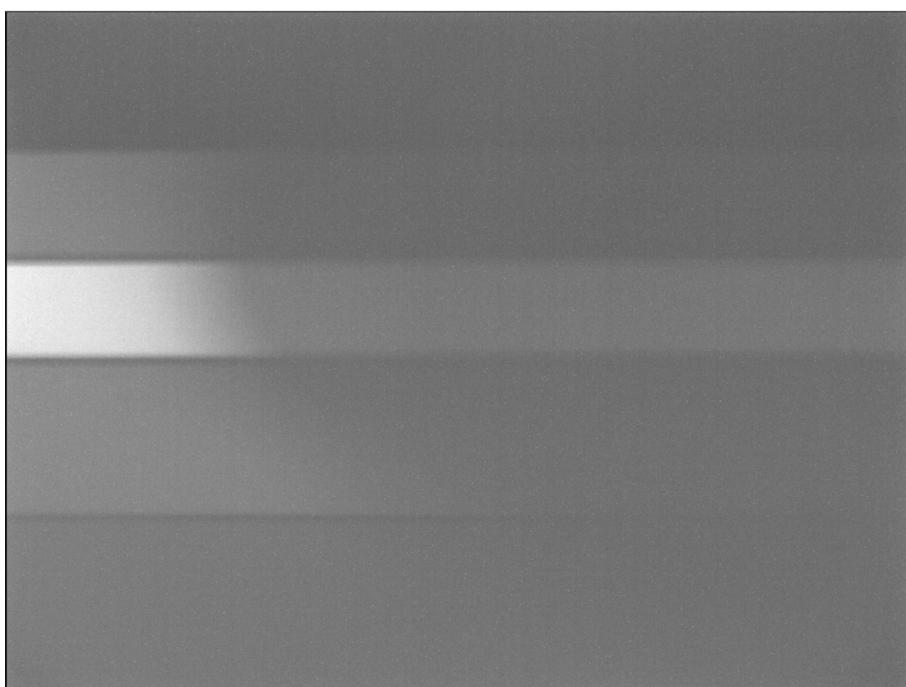


Figura 5.14: *Dark Frame* asociado a 5.13.

Tabla 15: Parámetros utilizados en la simulación 8.

Parámetro	Valor
Coeficiente de distorsión $k_{\text{real}}$	$ k_{\text{real}}  \in [5.93, 31.63] \text{ m}^{-2}$
Tamaño de píxel $px$	$5.5 \mu\text{m}$
Dimensión del sensor $\text{sen\_dim}$	$3072 \times 4096 \text{ px}$
Número de pinholes $n_{ph}$	$20 \times 15$
Espaciado entre pinholes $spacing$	172 px
Centro de la placa $O$	$[1536, 2048] + [\pm 100, \pm 100] \text{ px}$
Centro óptico $C_o$	$O + [\pm 100, 258 \pm 100] \text{ px}$
Intensidad de pinhole $I_i$	$\in [0, 3000]$
Ángulo de rotación $\theta$	$\in [-15, 15] \text{ rad}$

Tabla 16: Resultados medios de la simulación 8 en base a las métricas propuestas.

Parámetro	Valor
Coeficiente de distorsión medido $k_m$	-
Error en el centro óptico $\Delta C_o$	3.5 px
MDLD	$5.0418 \times 10^{-8}$
RMSE px	$5.9380 \times 10^{-7}$
$E_r$	8.3555
$R^2$	0.9857

Es apreciable que las imágenes no han sido realizadas controlando los parámetros variables de 4.3. La placa no está centrada (el centro se diferencia con unos pinholes en forma de cruz) ni recta (hay algo de rotación), no se ha alineado previamente con el eje óptico y el centro óptico es una incognita, no todos los pinholes son lo suficientemente brillantes y tampoco se ha medido con precisión el **spacing**. Es decir, esta imagen entra en el marco de simulaciones 8 y los resultados tendrán, probablemente, los errores mostrados en la Tabla 16.

Para dar una estimación del centro óptico se ha propone utilizar el **Flat Frame**. Esto es una imagen de calibración utilizada en astrofotografía para corregir irregularidades ópticas como el viñeteo (esquinas oscuras), manchas de polvo en el sensor o filtro, y desalineaciones del campo. Se obtienen tomando una foto de una fuente de luz uniforme, permitiendo al software de procesado eliminar estos defectos. Debido a su simetría radial, es posible ajustar un círculo y usar el centro de este como centro óptico. En la Figura 5.16 se puede observar un ejemplo de flat con su correspondiente ajuste y centro óptico marcado. Esto es simplemente una proposición y no se ha implementado en este caso concreto debido a la falta del flat. En cambio, se ha recurrido a la siguiente opción que es suponer como centro óptico el centro de la imagen que. Esta es una muy pobre aproximación y puede alterar notoriamente los resultados. Aunque, ante la falta de un flat se puede utilizar como opción de último recurso.

Después de pasar las imágenes por el algortimo se han obtenido los resultados mostrados en la Tabla 17. La medida obtenida entra dentro del rango razonable que se esperaba: una distorsión de pocos píxeles a lo sumo. El error medio de los ajustes  $R^2$ , no es bueno, pero es razonable y el RMSE es notoriamente bajo. El  $\Delta C_o$  no tiene mu-

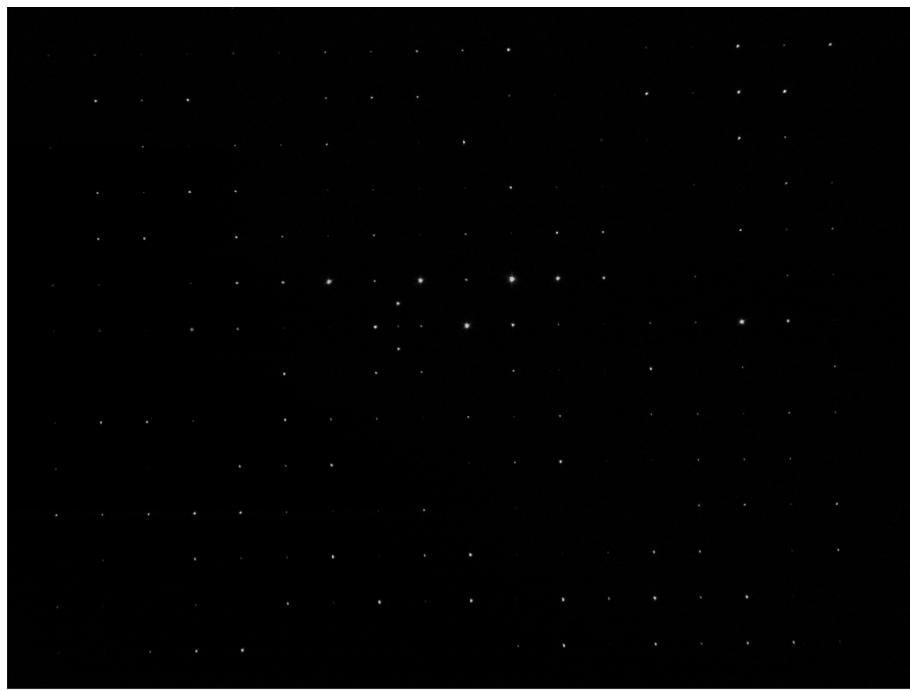


Figura 5.15: Imagen procesada de un pinhole grid hecha con xxx.

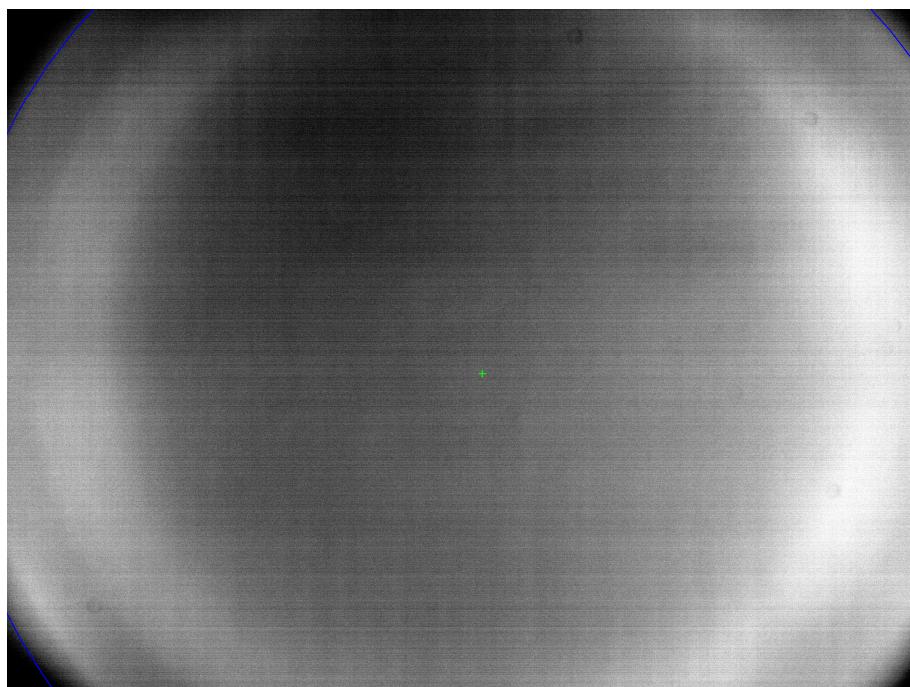


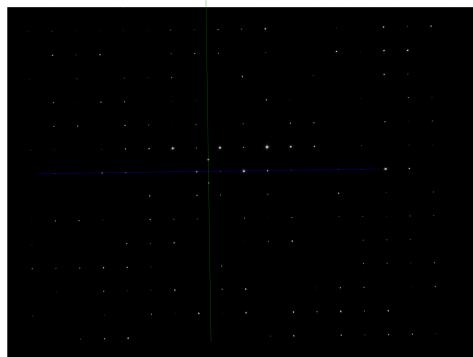
Figura 5.16: Ejemplo de flat con su ajuste de círculo dibujado en azul y el centro óptico en verde.

Tabla 17: Resultados medios obtenidos después de aplicar el método propuesto a imágenes proporcionadas por Satlantis Microsats.

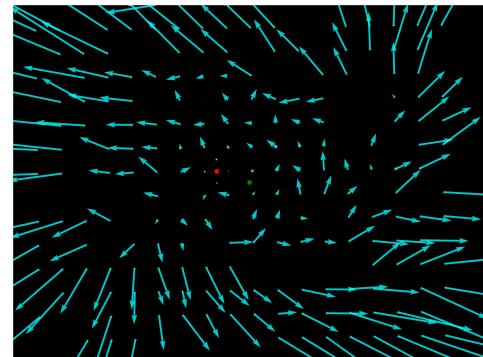
Parámetro	Valor
Coeficiente de distorsión medido $k_m$	$29.2157 \text{ m}^{-2}$
Error en el centro óptico $\Delta C_o$	3.5
MDLD	-
RMSE $px$	$6.1427 \times 10^{-6}$
$E_r$	-
$R^2$	0.7859

cho sentido en este caso; ya que, se ha partido de un  $C_o$  sin demasiado rigor. El MDLD y el  $E_r$  no se pueden calcular; ya que, no se tienen los valores verdaderos.

En la Figura 5.17 se muestra el mapa de distorsión y los ejes principales de la placa calculado por el algoritmo. Observando ambas figuras se puede intuir un pequeño error en el cálculo de la orientación; sobre todo, si nos fijamos en la silueta tipo "vórtice" del mapa de distorsión. Este pequeño error ha podido empeorar la medida del coeficiente de distorsión.



(a) Ejes principales calculados por el algoritmo en la imagen proporcionada por Satlantis Microsats.



(b) Mapa de distorsión calculado por el algoritmo en la imagen proporcionada por Satlantis Microsats. Las flechas (zoom x100) representan el desplazamiento de los pinholes  $M_i - P_i$ . Los puntos rojo y verde son el centro de la placa y óptico respectivamente.

Figura 5.17: Representaciones asociadas a la imágenes proporcionadas por Satlantis Microsats

## 6 Conclusiones

En este trabajo se ha propuesto un algoritmo de detección de distorsión y se ha puesto en práctica con imágenes proporcionadas por Satlantis. Además, se han estudiado los

límites y condiciones en los que la metodología propuesta trabaja mejor y peor. Exitosamente se ha validado el algoritmo para calcular el perfil de distorsión de un único parámetro que para cámaras profesionales es suficiente. El método, aún teniendo sus limitaciones y requerimientos como preparar el banco experimental (placa de pinholes, alineamiento de centro óptico, medidas del *spacing* ...) se ha demostrado que es muy fiable. Para futuros trabajos se propone mejorar la modelización y alineamiento de los pinholes ideales.

## References

- [1] C. S. Williams, *Introduction to the optical transfer function*, (2002) <https://archive.org/details/introductiontoop0000will>.
- [2] R. N. Bracewell, *The fourier transform and its applications*, 3rd (McGraw-Hill, 2000).
- [3] E. Optics, *Introduction to modulation transfer function*, <https://www.edmundoptics.eu/knowledge-center/application-notes/optics/introduction-to-modulation-transfer-function/>, Accedido el 17 de junio de 2025, n.d.
- [4] D. C. Brown, "Close-range camera calibration", *Photogrammetric Engineering* **37**, [Accedido el 15 de septiembre de 2021, 855–866 \(1971\)](#).
- [5] A. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion", in *Proceedings of the 2001 ieee computer society conference on computer vision and pattern recognition. cvpr 2001*, Vol. 1 (2001), pp. I–I.
- [6] F. Bukhari and M. Dailey, "Automatic radial distortion estimation from a single image", *Journal of Mathematical Imaging and Vision* **45**, 31–45 (2013).
- [7] M. Alemán-Flores, L. Alvarez, L. Gomez, and D. Santana-Cedrés, "Line detection in images showing significant lens distortion and application to distortion correction", *Pattern Recognition Letters* **36**, 261–271 (2014).
- [8] M. Alemán-Flores, L. Alvarez, L. Gomez, and D. Santana-Cedrés, "Automatic lens distortion correction using one-parameter division models", *Image Processing On Line* **4**, 327–343 (2014).
- [9] D. Santana-Cedrés, L. Gomez, M. Alemán-Flores, A. Salgado, J. Esclarín, L. Mazorra, and L. Alvarez, "An iterative optimization algorithm for lens distortion correction using two-parameter models", *Image Processing On Line* **6**, 326–364 (2016).
- [10] J. Rong, S. Huang, Z. Shang, and X. Ying, "Radial lens distortion correction using convolutional neural networks trained with synthesized images", in *Asian conference on computer vision* (2016), pp. 35–49.
- [11] M. Lopez, R. Mari, P. Gargallo, Y. Kuang, J. Gonzalez-Jimenez, and G. Haro, "Deep single image camera calibration with radial distortion", in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (2019), pp. 11817–11825.
- [12] J. Xiao, K. Ehinger, A. Oliva, and A. Torralba, "Recognizing scene viewpoint using panoramic place representation", in *Proceedings of the ieee conference on computer vision and pattern recognition* (2012), pp. 2695–2702.
- [13] K. Liao, C. Lin, and Y. Zhao, "A deep ordinal distortion estimation approach for distortion rectification", *IEEE Transactions on Image Processing* **30**, 3362–3375 (2021).

- [14] K. Liao, C. Lin, Y. Zhao, and M. Gabbouj, "DR-GAN: automatic radial distortion rectification using conditional GAN in real-time", *IEEE Transactions on Circuits and Systems for Video Technology* **30**, 725–733 (2019).
- [15] X. Li, B. Zhang, P. Sander, and J. Liao, "Blind geometric distortion correction on images through deep learning", in *Proceedings of the ieee conference on computer vision and pattern recognition* (2019), pp. 4855–4864.
- [16] C.-H. Chao, P.-L. Hsu, H.-Y. Lee, and Y.-C. Wang, "Self-supervised deep learning for fisheye image rectification", in *Icassp 2020-2020 ieee international conference on acoustics, speech and signal processing* (IEEE, 2020), pp. 2248–2252.
- [17] F. Zhu, S. Zhao, P. Wang, H. Wang, H. Yan, and S. Liu, "Semi-supervised wide-angle portraits correction by multi-scale transformer", in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (2022), pp. 19689–19698.
- [18] S. Yang, C. Lin, K. Liao, C. Zhang, and Y. Zhao, "Progressively complementary network for fisheye image rectification using appearance flow", in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (2021), pp. 6348–6357.
- [19] W. Wang, H. Feng, W. Zhou, Z. Liao, and H. Li, "Model-aware pre-training for radial distortion rectification", *IEEE Transactions on Image Processing* (2023).
- [20] X. Yin, L. Zhao, X. Li, and Z. Fang, "Automatic centroid detection and surface measurement with a digital Shack–Hartmann wavefront sensor", *Measurement Science and Technology* **21**, 015304 (2010).
- [21] H. W. Kuhn, "The hungarian method for the assignment problem", *Naval Research Logistics Quarterly* **2**, 83–97 (1955).
- [22] P. Virtanen et al., "Scipy 1.0: fundamental algorithms for scientific computing in python", *Nature Methods* **17**, 261–272 (2020).
- [23] D. C. Montgomery and G. C. Runger, *Applied statistics and probability for engineers*, 6th ed. (Wiley, 2014).
- [24] F. Wang, Z. Zhang, R. Wang, X. Zeng, Y. Xu, S. Lv, F. Zhang, D. Xue, J. Yan, and X. Zhang, "Distortion measurement of optical system using phase diffractive beam splitter", *Optics Express* **27**, 29803–29817 (2019).