



UNIVERSIDAD POLITÉCNICA DE CARTAGENA

# Estudio comparativo de algoritmos de reconstrucción de imágenes basados en la detección del frente de onda

Estudio analítico y comparativo de algoritmos de reducción de ruido atmosférico con métodos de reducción de imágenes de corta exposición y métodos de deconvolución ciega y derivados de la detección del frente de onda.

**Jose Abraham Caravaca Fernández**  
Septiembre de 2012

<b>Autor</b>	Jose Abraham Caravaca Fernández
<b>E-mail del Autor</b>	Jab.caravaca@gmail.com
<b>Director(es)</b>	Francisco Javier Garrigós Guerrero
<b>E-mail del Director</b>	javier.garrigos@upct.es
<b>Título del PFC</b>	Estudio comparativo de algoritmos de reconstrucción de imágenes basados en la detección del frente de onda
<b>Descriptor(es)</b>	Frente de onda, Astronomía, Software
<b>Resumen</b> <p>Los telescopios sufren gran cantidad de perturbaciones e interferencias que maltratan la calidad de las imágenes que ofrecen. La mayor fuente de ruido en los telescopios de base terrestre se encuentra a menudo en el efecto de la atmosfera. Para eliminar el ruido y las deformaciones existen muchos métodos, algunos especializados en la eliminación de los efectos perniciosos de la turbulencia atmosférica son los basados en la detección del frente de onda de la luz que incide en el momento de la captura. Estos métodos son el principal objeto de estudio de este PFC.</p> <p>Las posibilidades de rendimiento de cada método suele estar ligada a un entorno concreto que fija las condiciones de funcionamiento. Rara vez se encuentran estudios de distintos algoritmos bajo un entorno parametrizable, siendo en la mayoría de los casos específicos a ciertos ejemplos de estrellas o conjuntos de ellas.</p> <p>Este estudio tiene por propósito crear un marco común para evaluar comparativamente varios algoritmos de restauración y de deconvolución de imágenes tomadas por telescopios. Se han creado herramientas en ImageJ, un programa multiplataforma de tratamiento de imágenes de licencia GNU, para simular el comportamiento de los algoritmos en entornos artificiales parametrizados. El estudio arroja resultados para inspección visual y una serie de estadísticos cruzados.</p>	
<b>Titulación</b>	Ingeniería de Telecomunicación
<b>Departamento</b>	Electrónica, tecnología de computadores y proyectos
<b>Fecha de Presentación</b>	Septiembre- 2012

## Contenido

<b>Índice de ilustraciones</b> .....	<b>- 4 -</b>
<b>Índice de tablas</b> .....	<b>- 6 -</b>
<b>1. MOTIVACIÓN DEL PROYECTO</b> .....	<b>- 7 -</b>
1.1    Introducción .....	- 7 -
1.2    Motivación.....	- 8 -
1.3    Objetivos.....	- 9 -
<b>2 ESTADO DE LA TÉCNICA.....</b>	<b>- 10 -</b>
2.1    Introducción al problema del seeing .....	- 10 -
2.2    Algunas técnicas de corrección .....	- 11 -
<b>3. ALGORITMOS BAJO ESTUDIO .....</b>	<b>- 14 -</b>
<b>3.1    Introducción al modelo de deconvolución</b> .....	<b>- 14 -</b>
3.2    Primera aproximación: cociente de Fourier .....	- 15 -
3.3    Regularización lineal: filtrado de Wiener.....	- 16 -
3.4    Regularización lineal: regularización de Tikhonov .....	- 17 -
3.5    Reducción de ruido por promediado.....	- 19 -
<b>4. PARAMETROS DEL ESTUDIO.....</b>	<b>- 21 -</b>
4.1    Creación del escenario de test .....	- 21 -
4.1.1    Generador de patrones de Airy .....	- 21 -
4.1.2    Generador de patrones de dispersión .....	- 26 -
4.2    Enumeración de los algoritmos en prueba .....	- 31 -
<b>5 ESTUDIO DE CORTA ESCALA .....</b>	<b>- 32 -</b>
5.1    Comparativa visual Tip&Tilt de 20 muestras .....	- 32 -
5.1.1    Niveles de ruido.....	- 32 -
5.1.2    Resultado algoritmo FASTCAM [1] .....	- 34 -
5.1.3    Resultado deconvolución estándar [2] y [5] .....	- 35 -
5.1.4    Resultado deconvolución Primot [4] y [7].....	- 36 -
5.1.5    Resultado deconvolución de Wiener con regularización constante [3] y [6]... -	39 -
5.1.6    Resultado deconvolución de Wiener con regularización paso alto [8].....	- 45 -
5.2    Comparativa visual de modelo atmosférico de 20 muestras .....	- 47 -
5.2.1    Niveles de ruido.....	- 47 -
5.2.2    Resultado algoritmo FASTCAM [1] .....	- 47 -
5.2.3    Resultado deconvolución estándar [2] y [5] .....	- 49 -
5.2.4    Resultado deconvolución Primot [4] y [7].....	- 50 -
5.2.5    Resultado deconvolución de Wiener con regularización constante [3] y [6]... -	53 -
5.2.6    Resultado deconvolución de Wiener con regularización paso alto [8].....	- 56 -
5.3    Estudio estadístico de los modelos de 20 muestras .....	- 57 -
5.3.1    Figuras de mérito de los algoritmos .....	- 57 -

5.3.2	Gráficos del estudio .....	- 58 -
5.3.3	Estudio de coste computacional.....	- 63 -
<b>6.</b>	<b>ESTUDIO DE LARGA ESCALA .....</b>	<b>- 64 -</b>
6.1	Comparativa visual de modelo atmosférico de 1000 muestras .....	- 64 -
6.1.1	Niveles de ruido .....	- 64 -
6.1.2	Resultado algoritmo FASTCAM [1] .....	- 65 -
6.1.3	Resultado deconvolución estándar [2] y [5] .....	- 67 -
6.1.4	Resultado deconvolución Primot [4] y [7].....	- 69 -
6.1.5	Resultado deconvolución de Wiener con regularización constante [3] y [6]...	- 70 -
6.1.6	Resultado deconvolución de Wiener con regularización paso alto [8].....	- 73 -
6.2	Estudio estadístico de modelo atmosférico de 1000 muestras.....	- 75 -
6.2.1	Figuras de mérito del estudio .....	- 75 -
6.2.2	Gráficos del estudio .....	- 75 -
6.2.3	Estudio de coste computacional.....	- 78 -
<b>7.</b>	<b>CONCLUSIONES .....</b>	<b>- 79 -</b>
7.1	Líneas futuras de investigación .....	- 80 -
<b>LA HERRAMIENTA IMAGEJ .....</b>	<b>LA HERRAMIENTA IMAGEJ .....</b>	<b>- 81 -</b>
A.1	Introducción .....	- 81 -
A.2	Conceptos básicos .....	- 82 -
A.2.1	ImageJ es software libre.....	- 82 -
A.2.2	Funcionalidad básica de la herramienta ImageJ.....	- 82 -
A.3	ImageJ y la computación de la FFT .....	- 88 -
<b>Bibliografía .....</b>	<b>Bibliografía .....</b>	<b>- 89 -</b>

## Índice de ilustraciones

Ilustración 1. Comparativa seeing.....	- 10 -
Ilustración 2. Orografía y el seeing.....	- 11 -
Ilustración 3: Diagrama de caja negra del modelo .....	- 14 -
Ilustración 4. Diagrama del filtrado de Wiener.....	- 17 -
Ilustración 5. Imagen original .....	- 18 -
Ilustración 6. Ruido aditivo .....	- 18 -
Ilustración 7. Imagen con ruido .....	- 18 -
Ilustración 8. Filtrado Paso Alto de la imagen ruidosa .....	- 18 -
Ilustración 9. Diagrama de bloques algoritmo Primot.....	- 20 -
Ilustración 10: Caso ideal.....	- 22 -
Ilustración 11: Función de Bessel de primera especie utilizada en el patrón .....	- 23 -
Ilustración 12: Escena inicial sin ajuste de contraste .....	- 25 -
Ilustración 13: Escena original cubriendo todo el rango ( <i>adjusted contrast</i> ) .....	- 25 -
Ilustración 14. Histograma del caso ideal.....	- 25 -
Ilustración 15: Sensor Frontsurfer de OKO TECHNOLOGIES.....	- 26 -
Ilustración 16. Visualización de la función pupila en 2D y 3D .....	- 28 -
Ilustración 17. Captura del interfaz indicando los modos .....	- 28 -
Ilustración 18. Ejemplo de una PSF totalmente aleatoria.....	- 29 -
Ilustración 19. Ejemplo de una PSF generada con el modelo de Noll.....	- 29 -
Ilustración 20. Ajuste de verosimilitud respecto al patrón .....	- 29 -
Ilustración 21. PSF con patrón desde instrumentación .....	- 30 -
Ilustración 22. Niveles de intensidad de las seis estrellas originales .....	- 32 -
Ilustración 23. Una de las imágenes ruidosas $\sigma_n = 5 \cdot 10^6$ .....	- 33 -
Ilustración 24. Resultado de FASTCAM TipTilt .....	- 34 -
Ilustración 25. Histograma FASTCAM TipTilt.....	- 35 -
Ilustración 26. Deconvolución estándar con Prepromediado en TipTilt.....	- 35 -
Ilustración 27. Deconvolución estándar con Postpromediado en TipTilt .....	- 35 -
Ilustración 28. Histograma std Prepromediado TipTilt.....	- 36 -
Ilustración 29. Histograma std Postpromediado TipTilt .....	- 36 -
Ilustración 30. Resultado de algoritmo Primot (prepromediado) TipTilt.....	- 36 -
Ilustración 31. Modificación de Postpromediado de Primot TipTilt .....	- 37 -
Ilustración 32. Histograma Primot Prepromediado TipTilt .....	- 38 -
Ilustración 33. Histograma Primot Postpromediado TipTilt .....	- 38 -
Ilustración 34. Resultado Wiener Prepromediado TipTilt ( $\gamma = 5.2 \cdot 10^6$ ) .....	- 39 -
Ilustración 35. Representación 3D Wiener Prepromediado TipTilt ( $\gamma = 5.2 \cdot 10^6$ ) .....	- 40 -
Ilustración 36. Wiener Postpromediado TipTilt ( $\gamma = 5.2 \cdot 10^6$ ) .....	- 41 -
Ilustración 37. Representación 3D de Wiener Postpromediado TipTilt .....	- 42 -
Ilustración 38. Comparación de histogramas, a la izquierda prepromediada, a la derecha postpromediada.....	- 42 -
Ilustración 39. Arriba $\gamma = 1 \cdot 10^6$ , medio $\gamma = 5.2 \cdot 10^6$ , abajo $\gamma = 10 \cdot 10^6$ .....	- 43 -
Ilustración 40. Comparación del efecto del parámetro de regularización, de izquierda a derecha se aumenta el parámetro.....	- 44 -
Ilustración 41. Comparación de histogramas en variación del parámetro $\gamma$ .....	- 44 -
Ilustración 42. Resultado de regularización paso alto en el modelo tip tilt de 20 muestras.....	- 45 -
Ilustración 43. Imagen de modelo atmosférico con ruido de $\sigma_N = 4.2e6$ .....	- 47 -
Ilustración 44. Solución FASTCAM 50% en modelo atmosférico.....	- 47 -
Ilustración 45. Surface plot de FASTCAM 50% en modelo atmosférico .....	- 48 -
Ilustración 46. Histograma de FASTCAM 50% en modelo atmosférico .....	- 48 -
Ilustración 47. Caso de deconvolución estándar con modelo atmosférico. Izquierda: promediado anterior. Derecha: promediado posterior. ....	- 49 -
Ilustración 48. Comparación de histogramas de deconvolución estándar, a la izquierda prepromediada, a la derecha postpromediada .....	- 49 -
Ilustración 49. Caso Primot prepromediado en modelo atmosférico.....	- 50 -
Ilustración 50. Surface plot de Primot con prepromediado en modelo atmosférico.....	- 51 -
Ilustración 51. Caso Primot Postpromediado con modelo atmosférico .....	- 52 -
Ilustración 52. Visualización 3D del caso Primot Postpromediado con modelo atmosférico .....	- 52 -

Ilustración 53. Caso Wiener con prepromediado en modelo atmosférico de 20 muestras. ( $\gamma = 4.2 \cdot 106$ )	- 53 -
Ilustración 54. Trazado 3D del caso Wiener con prepromediado en el modelo atmosférico de 20 muestras ( $\gamma = 4.2 \cdot 106$ )	- 54 -
Ilustración 55. Caso Wiener con postpromediado en modelo atmosférico de 20 muestras. ( $\gamma = 4.2 \cdot 106$ )	- 54 -
Ilustración 56. Trazado 3D del caso Wiener con postpromediado en el modelo atmosférico de 20 muestras.	- 55 -
Ilustración 57. Comparación histogramas Pre y Post promediados de Wiener en el modelo atmosférico de 20 muestras. A la izquierda: Prepromediado, a la derecha Postpromediado. ( $\gamma = 4.2 \cdot 106$ )	- 55 -
Ilustración 58. Resultado de regularización paso alto en el caso de modelo atmosférico de 20 muestras	- 56 -
Ilustración 59. Gráfica correlación en modelo TipTilt de 20 muestras	- 58 -
Ilustración 60. Gráfica correlación en modelo atmosférico de 20 muestras	- 59 -
Ilustración 61. Comparativa correlación de modelo atmosférico versus Tip&Tilt	- 59 -
Ilustración 62. Gráfica MSE en modelo Tip&Tilt de 20 muestras	- 60 -
Ilustración 63. Gráfica MSE en modelo atmosférico de 20 muestras	- 60 -
Ilustración 64. Gráfica SNR en modelo Tip&Tilt de 20 muestras	- 61 -
Ilustración 65. Gráfica SNR en modelo atmosférico de 20 muestras	- 61 -
Ilustración 66. Gráfica sensibilidad en modelo Tip&Tilt de 20 muestras	- 62 -
Ilustración 67. Gráfica sensibilidad en modelo atmosférico de 20 muestras	- 62 -
Ilustración 68. Gráfico de comparación de tiempos de computación en 20 muestras.	- 63 -
Ilustración 69. Imagen de modelo atmosférico con ruido de $\sigma N = 4e6$	- 64 -
Ilustración 70. Resultado de FASTCAM 10% de larga escala en modelo atmosférico	- 65 -
Ilustración 71. Histograma de FASTCAM 10% de larga escala en modelo atmosférico	- 66 -
Ilustración 72. Representación 3D de FASTCAM 10% de larga escala en modelo atmosférico	- 66 -
Ilustración 73. Resultado de deconvolución estándar prepromediada en modelo atmosférico de larga escala	- 67 -
Ilustración 74. Resultado de deconvolución estándar postpromediada en modelo atmosférico de larga escala	- 68 -
Ilustración 75. Comparación de histogramas de deconvolución estándar en modelo atmosférico de larga escala, a la izquierda caso prepromediado, a la derecha caso postpromediado	- 68 -
Ilustración 76. Caso Primot prepromediado en modelo atmosférico de 1000 muestras	- 69 -
Ilustración 77. Caso Wiener con prepromediado en modelo atmosférico de 1000 muestras. ( $\gamma = 4 \cdot 106$ )	- 70 -
Ilustración 78. Representación 3D de caso Wiener con prepromediado en modelo atmosférico de 1000 muestras	- 70 -
Ilustración 79. Caso Wiener con postpromediado en modelo atmosférico de 1000 muestras. ( $\gamma = 4 \cdot 106$ )	- 71 -
Ilustración 80. Representación 3D de caso Wiener con postpromediado en modelo atmosférico de 1000 muestras	- 71 -
Ilustración 81. Comparación histogramas Pre y Post promediados de Wiener en el modelo atmosférico de 1000 muestras. A la izquierda: Prepromediado, a la derecha Postpromediado. ( $\gamma = 4 \cdot 106$ )	- 72 -
Ilustración 82. Gráfico de comparación de correlación en modelo atmosférico de 1000 muestras	- 75 -
Ilustración 83. Gráfico de comparativa de correlación según muestras	- 76 -
Ilustración 84. Gráfico de comparación del error cuadrático de los métodos en escala logarítmica	- 76 -
Ilustración 85. Gráfico comparativo de la SNR para el modelo atmosférico de 1000 muestras	- 77 -
Ilustración 86. Gráfica comparativa de la sensibilidad para el modelo atmosférico de 1000 muestras	- 77 -
Ilustración 87. Gráfico de comparación de tiempos de computación en 1000 muestras	- 78 -
Ilustración 88: Diferentes tipos de ventanas de ImageJ	- 82 -
Ilustración 89: Barra de herramientas de ImageJ	- 83 -
Ilustración 90: Barra de estado de ImageJ	- 83 -
Ilustración 91: Barra de progreso de ImageJ	- 83 -
Ilustración 92: Imagen mostrada por ImageJ	- 84 -
Ilustración 93: Stack mostrado por ImageJ	- 84 -
Ilustración 94: Distintos tipos de selecciones que pueden elegirse en ImageJ	- 85 -
Ilustración 95: Tabla de búsqueda en ImageJ	- 87 -

## Índice de tablas

Tabla 1. Tabla de polinomios de Zernike .....	- 27 -
Tabla 2. Tabla comparativa de los algoritmos a evaluar .....	- 31 -
Tabla 3. Valores de intensidad de la muestra típica en caso Tip&Tilt .....	- 32 -
Tabla 4. Valores de intensidad de la muestra típica en caso atmosférico .....	- 47 -
Tabla 5. Figuras de mérito del estudio comparativo .....	- 57 -
Tabla 6. Parámetros estadísticos de los algoritmos en modelo de TipTilt para 20 muestras .....	- 57 -
Tabla 7. Parámetros estadísticos de los algoritmos en modelo de Noll para 20 muestras .....	- 58 -
Tabla 8. Tabla de comparación de tiempos de computación .....	- 63 -
Tabla 9. Valores de intensidad de la muestra típica en modelo atmosférico de 1000 muestras .....	- 64 -
Tabla 10. Parámetros estadísticos de los algoritmos en el modelo atmosférico para 1000 muestras .....	- 75 -
Tabla 11. Tabla de comparación de tiempos de computación .....	- 78 -

## 1. MOTIVACIÓN DEL PROYECTO

### 1.1 Introducción

En la exploración astronómica del universo la herramienta fundamental es el telescopio y su instrumentación auxiliar. Un telescopio es un instrumento que recoge los rayos de luz provenientes de objetos lejanos (estrellas, galaxias...) y con el que se pueden obtener imágenes definidas y nítidas de los mismos.

**En el espacio**, un telescopio puede producir imágenes de alta resolución limitadas básicamente por su **diámetro** y por la longitud de onda de la luz que incide sobre él. **En la tierra** sin embargo, la densidad y fluctuación de la **atmósfera** causan **perturbaciones** sobre la luz que disminuyen la calidad de las imágenes, volviéndose las mismas más borrosas. Las rápidas fluctuaciones de la atmósfera, del orden de las décimas de milisegundo, producen variaciones en los patrones de difracción de la luz que ocasionan el desenfoque de las imágenes.

Este efecto distorsionador de la atmosfera se conoce en astronomía como el *seeing* atmosférico y es el límite más importante de la calidad de las imágenes adquiridas por dispositivos terrestres.

El uso de **cámaras de alta velocidad** permite utilizar periodos de exposición tan cortos como para poder considerar que la **atmósfera** queda "**congelada**" durante el mismo, y así poder **elegir** aquellas imágenes que estén menos afectadas por la distorsión ("**Lucky Images**, estadísticamente, entre un 5 y un 15%, según las condiciones") y **combinarlas** para formar imágenes de mucha mayor resolución. Con esta técnica, y bajo condiciones atmosféricas normales, **es posible aumentar la resolución de las imágenes de un telescopio en un factor de 5 a 7**. Un telescopio terrestre de 2.5 metros podría alcanzar la resolución conseguida por el telescopio espacial Hubble (2.5 m) de manera muy económica.

Los métodos de "*Lucky Imaging*" siguen sin embargo siendo muy costosos. El alto número de imágenes necesarias para la obtención de una buena reconstrucción del objeto **requiere mucho tiempo de observación**. El tiempo de observación de grandes telescopios es un recurso muy valioso. Esto supone una limitación de los algoritmos de reducción de imágenes, pues **supone un desperdicio** muy elevado de tiempo de observación.

Es necesario intentar **reducir** estos largos **tiempos de observación** hasta que se llega a una calidad aceptable, es decir, de intentar usar todas las imágenes que pueden proporcionar las cámaras. Para hacer esto se va a recurrir a **sensores del frente de onda**. La información de estos sensores puede ser usada para **influir en el sistema óptico de recepción**, de manera que **contrarreste las aberraciones introducidas por la atmosfera**, obteniendo imágenes más certeras, y alcanzando una convergencia aceptable con el objeto en un rango de observaciones muy inferior al necesario en el caso de los algoritmos de reducción de imágenes.



Estos algoritmos de compensación de distorsión del camino óptico pueden llevarse a cabo de manera concurrente a la observación o con carácter posterior. La opción concurrente supone el uso de **Óptica Adaptativa (A.O)** que conlleva unos niveles de **inversión económica** en instrumentación óptica (espejos deformables de rápida y sensible respuesta mecánica de grandes radios) que suelen ser **inasumibles** para presupuestos algo austeros.

La segunda opción es el **tratamiento posterior**, que resulta ser un método mucho más económico ya que no supone inversión en hardware de óptica adaptativa. En este tipo de método se realiza una **deconvolución** posterior a la captura de la imagen aberrada usando una captura de la deformación a que la atmosfera sometió el frente de onda antes de llegar al objetivo. Este método se basa en la creación de un **filtro inverso** a la aberración que supuso la atmosfera en cada momento de captura. Como parece lógico, este método ha de **sufrir un retardo** importante, el retardo dependerá casi íntegramente del poder de computación del sistema, siendo útiles los sistema de computación paralela en combinaciones GPU+CPU o FPGAs.

## 1.2 Motivación

De entre todas las opciones que ofrece el amplio abanico de **métodos de restauración** de imágenes distorsionadas, o **eliminación del seeing**, no se puede sencillamente determinar cuál de ellos es el mejor absoluto, dado que cada método se comporta de manera distinta frente según qué situaciones, ofreciendo cada cual un rendimiento altamente influido por las condiciones de aplicación.

Ante esta tesitura, es conveniente **determinar las mejores circunstancias de funcionamiento de cada algoritmo**. Sabiendo que desempeño arrojará cada método concreto de deconvolución en una situación concreta, podremos crear un árbol de decisión para intentar encontrar el algoritmo de tratamiento óptimo, que nos podrá permitir obtener un mejor resultado final aplicando el algoritmo más adecuado a cada situación.

## 1.3 Objetivos

Los objetivos de este Proyecto de Fin de Carrera son:

- Comprender la base de diversas técnicas de Lucky Imaging.
- Comprender otros algoritmos existentes para la reducción de imágenes de corta exposición sin información del frente de onda.
- Comprender los modelos atmosféricos y las técnicas de óptica básica para la evaluación del frente de onda, PSF, función de transferencia óptica, etc.
- Analizar la herramienta de procesamiento de imágenes ImageJ y su lenguaje de desarrollo de macros mediante scripts, que será la seleccionada para codificar nuestros desarrollos propios.
- Realizar un generador parametrizable de funciones de dispersión puntual (Point Spread Function) artificiales que permita recrear situaciones adecuadas a nuestras simulaciones en la herramienta ImageJ.
- Desarrollar una implementación propia de los algoritmos más utilizados en la bibliografía para la deconvolución basada en la detección del frente de onda en el entorno ImageJ.
- Realizar una evaluación comparativa de los algoritmos más utilizados para la deconvolución de imágenes basada en la detección del frente de onda, fundamentalmente, el algoritmo de Primot [6] y los filtros de Wiener [7].
- Realizar un estudio comparativo de los rendimientos y límites de ruido asumibles para los métodos de restauración de que trata este trabajo.
- Si es posible, proponer alternativas a los algoritmos tradicionales, que mejoren sus resultados.

## 2 ESTADO DE LA TÉCNICA

### 2.1 Introducción al problema del seeing

Antes de la aparición de las cámaras digitales en la astronomía, era necesario unos tiempos de exposición muy largos para poder adquirir imágenes de las débiles fuentes de luz de los objetos astronómicos. Como ocurre en la adquisición de imágenes con largos periodos de exposición, es necesario que la imagen no cambie para que se preserve la nitidez y definición de los detalles de ésta.

Sin embargo, la atmosfera sufre durante estos largos tiempos de exposición una gran cantidad de fluctuaciones producidas por la mezcla de capas de aire a diferentes temperaturas y densidades. Estas fluctuaciones son altamente aleatorias y dan lugar a efectos indeseables que son diferentes según la altura del aire respecto del suelo. Este comportamiento, a la vista del observador en el objetivo del telescopio, se traduce en una imagen difuminada y en constante vibración. Este efecto sobre la imagen obtenida se conoce como **seeing** astronómico.

Las técnicas de adquisición de imágenes de baja intensidad requieren que la luz incida durante un periodo moderado de tiempo en el objetivo, sin embargo al tratarse de una imagen cambiante debido a que el frente de onda se distorsiona por efecto de la atmosfera, lo que se obtiene en cada instante es una imagen distorsionada del objeto (*speckle*) que se va acumulando dando lugar a una imagen difusa del objeto (*specklegram*).

Este efecto supone una limitación de la calidad de las imágenes astronómicas. En el caso de estar capturando una estrella puntual, lo que obtendremos será la función de dispersión que provoca el seeing existente en el momento de la captura. Una manera muy sencilla de cuantificar el seeing es medir la dispersión que sufre la imagen esta estrella, comparándola con la anchura de la estrella (*Ilustración 1*).

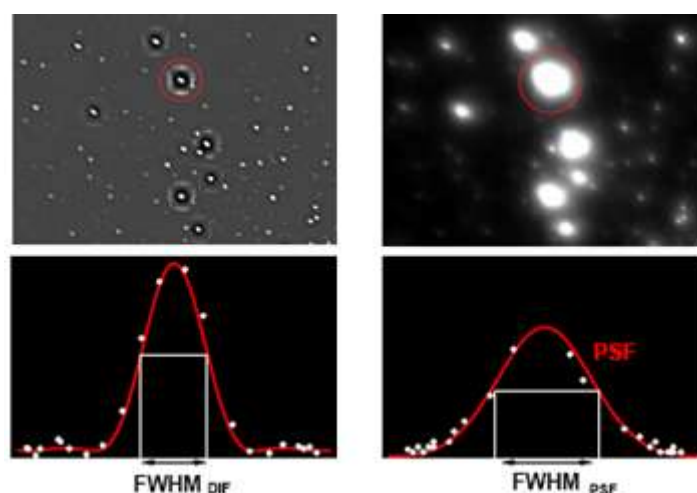


Ilustración 1. Comparativa seeing

Durante muchos años el seeing ha sido un factor determinante en la calidad de las imágenes que se adquieren a través de telescopios, siendo a menudo un punto esencial para la decisión del emplazamiento de los observatorios.

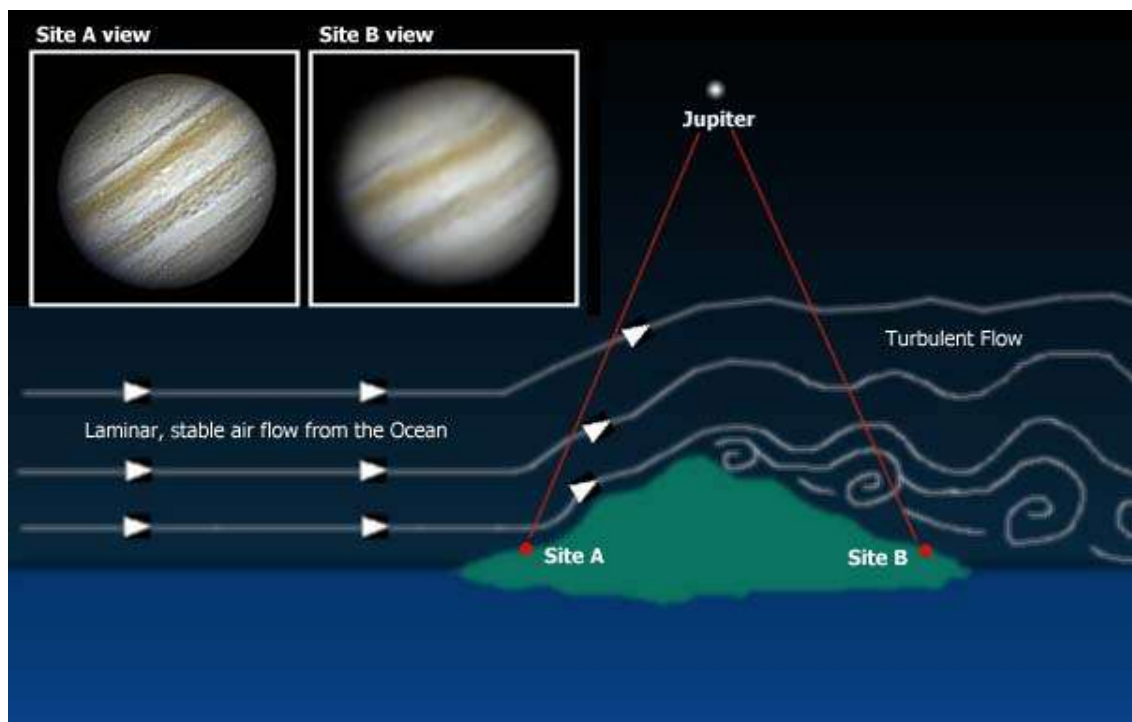


Ilustración 2. Orografía y el seeing

La elección de un buen sitio para el observatorio, no puede sin embargo más que reducir un poco los efectos del seeing pues siempre dependerá de la climatología y las posibilidades de que no haya seeing serán siempre, en última instancia, de carácter estocástico. El seeing es por tanto un **factor de ruido que, a priori, siempre estará presente** en mayor o menor medida.

## 2.2 Algunas técnicas de corrección

Con el perfeccionamiento de los sensores electrónicos de luz de extrema sensibilidad y la llegada de la fotografía digital, es posible utilizar otra estrategia. La idea es **reducir tanto el tiempo de exposición** que las variaciones de la atmosfera no suponga apenas ningún cambio en la muestra. De esta manera será posible obtener un gran número de imágenes donde los objetos estarán poco definidos debido a la baja cantidad de fotones capturados en el corto tiempo de apertura del sistema óptico. Estas imágenes deberán ser acumuladas para aumentar la intensidad de los objetos, obteniendo así mayor contraste, de una manera muy similar a la acumulación analógica tradicional, sin embargo, ahora podemos hacer una selección de aquellas imágenes que sean adecuadas para ser acumuladas debido a que hayan sufrido menor dispersión.

Estas imágenes tomadas en momentos adecuados son conocidas como "*Lucky images*" y existen una gran cantidad de algoritmos para diferenciar estas imágenes entre el conjunto total de imágenes. Estos algoritmos hacen una selección de las

mejores imágenes para obtener un promedio de baja distorsión. Se les conoce como **algoritmos de reducción de imágenes** de Corta Exposición.

Es posible demostrar como mediante estas técnicas **se puede llegar al límite teórico de difracción** en telescopios terrestres, sin necesidad de hacer grandes inversiones económicas en equipamiento. Sin embargo, esta técnica tiene un gran contrapunto: la baja **eficiencia temporal**, pues es necesario llegar a acumular un grandísimo número de imágenes, siendo además desechadas muchas de ellas posteriormente. Generalmente en grandes telescopios el tiempo de observación es un recurso muy valioso y esto se traduce en un bajo rendimiento de las instalaciones que son de por sí bastante caras.

Existe otra alternativa, no necesariamente excluyente a las técnicas de reducción de imágenes, que se apoya en información adicional para intentar restaurar las imágenes que están difuminadas en lugar de desecharlas. Esta información adicional es la referente a la **dispersión que provoca la atmosfera a los rayos al atravesarla** o a las posibles deformaciones en la óptica y se expresa mediante **funciones de dispersión puntual** (Point Spread Function, en adelante PSF).

La PSF describe la respuesta de un sistema óptico a la entrada de una fuente puntual de luz, generalmente conocida como *respuesta impulsional del sistema*. El grado de dispersión o emborronado de la fuente puntual de luz (objeto puntual) es una **medición de la calidad de un sistema óptico**.

La observación de un objeto complejo (no puntual), es decir la imagen del objeto, puede contemplarse como la convolución del objeto real y la PSF correspondiente al sistema óptico que genera dicha imagen. En el campo que nos ocupa, la observación astronómica, la determinación empírica de la PSF es a menudo una cuestión muy simple debido a la abundancia de puntos de luz (estrellas o quásares). La forma y origen de la PSF puede variar mucho dependiendo de los instrumentos y el contexto en que se usa.

Para **radiotelescopios** y **telescopios limitados por difracción** el factor dominante de la PSF suele ser la configuración de la apertura del telescopio en términos de Fourier. En la práctica suele haber muchos factores afectando a distintos componentes complejos del sistema óptico. Una descripción completa de la PSF debe incluir también información sobre difusión de la luz (o foto/electrones) así como información de posibles errores de apuntamiento en el telescopio o satélite.

Para **telescopios terrestres**, las turbulencias atmosféricas dominan sobre el resto de factores que pueden determinar la PSF. En telescopios terrestres de alta resolución la PSF suele estar afectada por la posición en la imagen, efecto conocido como *anisoplanatismo*. En telescopios terrestres con sistemas de óptica adaptativa la PSF suele estar compuesta por una combinación de los efectos de la apertura y los efectos residuales de factores atmosféricos aberrantes.

El efecto de la turbulencia atmosférica (Dong feng Shi, 2011) degrada la calidad de las imágenes, causando, como ya se comentó, fluctuaciones estocásticas espaciotemporales en el índice de refracción de la atmosfera y da lugar a funciones de dispersiones del punto variantes en tiempo. La contaminación de las imágenes por

efectos de la atmosfera supone un severo problema cuando se trata de imágenes de alta resolución en las que se desea un alto grado de detalle.

A lo largo de los años se ha propuesto un amplio elenco de algoritmos para solucionar este problema. Para poder evaluar la eficacia de los distintos algoritmos vamos a necesitar una herramienta que sea capaz de generar PSF artificiales que respondan a distintos casos de reconstrucción.

Hay diferentes formas de obtener estas funciones, muchos algoritmos de deconvolución buscan una **estimación** del objeto que atraviesa el sistema óptico suponiendo una **PSF** proveniente de una estimación empírica. Estas estimaciones sin embargo pueden sufrir desviaciones respecto de la PSF original demasiado importantes haciendo el resultado de la reconstrucción inaceptable. Esto hace que el proceso de la estimación de la PSF sea muy importante en el resultado final de la reconstrucción.

Oros hacen uso de los **sensores de frente de onda** (Wave Front Sensors) para averiguar que deformaciones ha sufrido la imagen en su paso por la atmosfera y reconstruir la PSF a través de estos. Estos métodos consiguen imágenes más definidas sin necesidad de desechar parte de las imágenes adquiridas y por tanto tardan menos tiempo en obtener una imagen nítida del objetivo, haciendo así que el tiempo de observación sea mucho más eficiente en resultados.

Una vez que se tiene una PSF se trata de deshacer las deformaciones ya sea mediante espejos deformables (**técnicas de óptica adaptativa**) que actúan en tiempo real o mediante un sistema óptico rígido con **filtrado de imagen inverso posterior** a la adquisición.

Los **métodos de óptica adaptativa** estiman la deformación del frente de onda y generan una deformación inversa (corregida) en el sistema óptico, de tal forma que el frente de onda quede "plano" y la PSF sea, idealmente, el patrón de Airy. Para corregir las aberraciones necesitan usar espejos deformables generalmente de precios muy elevados. Además los espejos deformables son difíciles de controlar en tiempo real (detectar el frente de onda, hacer los cálculos, enviar a los actuadores las señales necesarias para corregir la deformación de los espejos, etc.). Estos métodos son los que ofrecen mejores resultados pero suponen inversiones millonarias solo a disposición de unos pocos centros de observación con grandes presupuestos.

La adquisición de un sensor de frente de ondas es más barata y realizable para instituciones menos punteras con presupuestos más austeros. Usando estos sensores de frente de onda y equipos de óptica rígida (no adaptativa) haciendo un **procesado de las imágenes posterior**, se llega a una solución a medio camino entre el coste y la eficiencia en retardo del equipo. Existe también una buena cantidad de maneras de tratar de restaurar la imagen usando las PSF recuperadas gracias a los sensores del frente de onda, varias de las cuales serán estudiadas durante esta obra.

### 3. ALGORITMOS BAJO ESTUDIO

#### 3.1 Introducción al modelo de deconvolución

El punto de partida de cualquier algoritmo pasa por el modelo usado. En este epígrafe se va a describir un modelo sencillo en el que se apoyara toda la nomenclatura posterior de los algoritmos de tratamiento.

Considérese una imagen caracterizada por una distribución de intensidad (los datos)  $I$ , correspondientes a la observación de una “imagen real”  $O$  a través de un sistema óptico. Si el sistema óptico es lineal e invariante en el espacio, la relación entre los datos y la imagen en la misma base es una convolución:

$$I(x, y) = \int_{x_1=-\infty}^{\infty} \int_{y_1=-\infty}^{\infty} P(x - x_1, y - y_1) \cdot O(x_1, y_1) dx_1 dy_1 + N(x, y) \quad [1]$$

$$I(x, y) = (P * O)(x, y) + N(x, y) \quad [2]$$

Siendo  $P$  la PSF del sistema óptico y  $N$  el ruido aditivo. Esta función  $P(x, y)$  se conoce como **función de dispersión del punto** o en inglés Point Spread Function, en adelante y por comodidad PSF. La PSF describe la respuesta de un sistema óptico a una fuente de luz puntual. Una definición más general sería la respuesta impulsiva de un sistema óptico focal.

En el dominio de Fourier tendremos:

$$\tilde{I}(u, v) = \tilde{O}(u, v) \tilde{P}(u, v) + \tilde{N}(u, v) \quad [3]$$

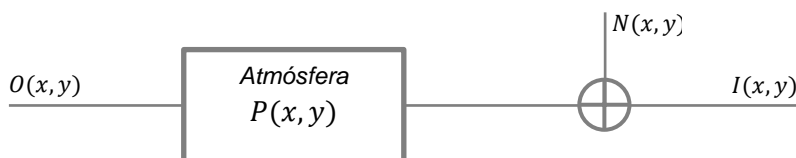


Ilustración 3: Diagrama de caja negra del modelo

Así pues, el problema consiste en determinar  $O$  sabiendo  $I$  y estimando  $P$  lo mejor que sea posible. Las principales dificultades para esto son la existencia de por un lado el ruido aditivo y por otro la existencia de una frecuencia de corte de la PSF.

Una solución puede estimarse si se computa la transformada de Fourier de la deconvolución de  $O$  simplemente como la división de la imagen y la PSF en espectro, sin tener en cuenta el factor de ruido:

$$\tilde{O}(u, v) = \frac{\tilde{I}(u, v)}{\tilde{P}(u, v)} - \frac{\tilde{N}(u, v)}{\tilde{P}(u, v)} \quad [4]$$

$$\tilde{O}_e(u, v) = \frac{\tilde{I}(u, v)}{\tilde{P}(u, v)} \quad [5]$$

Este método que, a menudo se conoce como el de “cociente de Fourier”, es muy rápido. Solo es necesario hacer una FFT y una IFFT. Para frecuencias cercanas a la

frecuencia de corte, el término de ruido se hace predominante sobre la señal y el ruido se amplifica. Es por eso que el método de cociente de Fourier, que se explica a continuación, no se puede usar en la práctica.

## 3.2 Primera aproximación: cociente de Fourier

En primera instancia vamos a plantear el problema suponiendo que disponemos de la imagen del objeto sin adulterar, es decir sin considerar ningún término de ruido  $N(x, y)$  y de una PSF que se supone exactamente igual a la que introdujo la atmosfera.

Bajo estas premisas la recuperación de la imagen real del objeto  $O(x, y)$  se puede plantear de manera simple como:

$$O(x, y) = \mathcal{F}^{-1} \left\{ \frac{\hat{I}(u, v)}{\hat{P}(u, v)} \right\} \quad [6]$$

El problema de esta solución es que hemos supuesto que disponemos de la PSF exacta que afecto a la imagen real del objeto y esto es imposible en la realidad. Al pasar la función  $P(x, y)$  al dominio de la frecuencia los errores entre la  $P(x, y)$  y la original, por pequeños que sean van a dar lugar a una deconvolución que en general poco o nada se parecerá a la imagen real del objeto  $O(x, y)$ :

$$O'(x, y) = \mathcal{F}^{-1} \left\{ \frac{\hat{I}}{\hat{P}_r} \right\} \neq O(x, y) \quad [7]$$

Si se carece de información o se tiene poca información para caracterizar la PSF, caso de no usar sensores del frente de onda o de usar sensores de baja resolución respecto al sistema óptico utilizado, se puede hacer una estimación de la PSF e intentar deconvolucionar con ella, esta familia de métodos se conoce por el nombre de algoritmos de deconvolución ciega (*blind deconvolution*). Aunque existen muchas variantes en la forma de estimar la PSF lo más común en los algoritmos de deconvolución ciega es hacer una estimación de máxima verosimilitud de la PSF.

$$P \rightarrow \hat{P}_{ML} \quad [8]$$

Estas estimaciones estadísticas son, no obstante, muy complicadas de realizar en un software orientado a procesado de imagen sencilla como es ImageJ. En el estudio llevado a cabo por este trabajo, se ha implementado la deconvolución sencilla con la PSF recuperada a través de los cálculos del sensor de frente de onda. Como se dijo en el apartado anterior este algoritmo a priori no va a aguantar casi ningún nivel de ruido sobre los cálculos de la PSF, como se podrá comprobar en el capítulo que recoge los resultados de las pruebas, pero será muy útil a efectos comparativos para el resto de algoritmos.



### 3.3 Regularización lineal: filtrado de Wiener

En 1949 *Norbert Wiener*, un matemático americano, propuso un filtro que tenía por propósito de reducir el ruido presente en una señal en comparación a una estimación de la señal deseada carente de ruido. Su versión discreta fue desarrollada por *Kolmogorov* unos años más tarde. Hay que señalar que no se trata de un filtro adaptativo puesto que se suponen entradas estacionarias.

La deconvolución de Wiener es la aplicación del filtro de Wiener a los problemas de ruido inherentes a la deconvolución. El fundamento de corrección de ruido funciona en el dominio de la frecuencia intentando minimizar el impacto del ruido de deconvolución a frecuencias que tienen baja relación señal a ruido.

Bajo la teoría de los sistemas lineales invariantes en el espacio (*Linear Space Invariant systems*) el filtrado se puede expresar como:

$$y(x, y) = h(x, y) * x(x, y) + n(x, y) \quad [9]$$

Siendo:

- $h(x, y)$  la respuesta al impulso del sistema LTI.
- $x(x, y)$  la señal de entrada en el instante  $t$ .
- $y(x, y)$  la señal de salida observada en el instante  $t$ .
- $n(x, y)$  el término aditivo de ruido independiente de  $x(t)$ .

El objetivo es encontrar una función  $g(x, y)$  tal que sea posible  $\hat{x}(x, y)$  para obtener un mínimo error cuadrático (ajuste por mínimos cuadrados):

$$\hat{x}(x, y) = g(x, y) * y(x, y) \quad [10]$$

El filtro de deconvolución de Wiener proporciona una  $g(x, y)$  tal que:

$$G(u, v) = \frac{H^*(u, v)S(u, v)}{|H(u, v)|^2 S(u, v) + N(u, v)} \quad [11]$$

Colocando todos los términos respecto al espectro de la señal deseada:

$$G(u, v) = \frac{H^*(u, v)}{|H(u, v)|^2 + \frac{N(u, v)}{S(u, v)}} \quad [12]$$

Para simplificar llamaremos a  $\frac{N(u, v)}{S(u, v)}$  factor de regularización de Wiener y lo denotaremos por la letra  $\sigma$ , llegando a:

$$G(u, v) = \frac{H^*(u, v)}{|H(u, v)|^2 + \sigma} \quad [13]$$

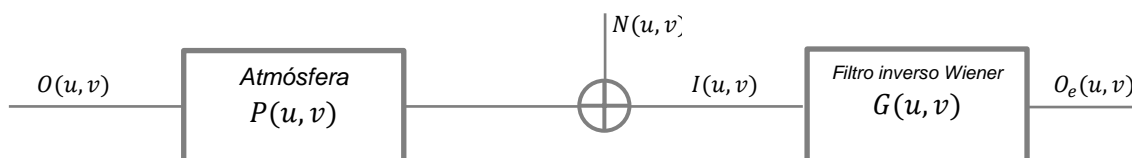


Ilustración 4. Diagrama del filtrado de Wiener

Así para la creación del filtro más adecuado, es indispensable disponer de una estimación de la cantidad de ruido presente en la imagen  $I(u, v)$ . Bajo estas condiciones el algoritmo de deconvolución de Wiener es capaz de crear un filtro inverso que da muy buenos resultados en la eliminación de los efectos de aberración de la atmosfera, sin embargo, no siempre será tan fácil conseguir buenas estimaciones del ruido que afecta a la imagen como ya veremos en los resultados.

### 3.4 Regularización lineal: regularización de Tikhonov

Como se ha visto en el caso del filtro de Wiener, más concretamente en la expresión [12], es necesario disponer de la potencia espectral de la relación ruido a señal. Esta relación ruido a señal es difícil de calcular dado que a priori desconocemos las características del ruido presente en  $I$ .

Sin embargo es sabido que en las imágenes provenientes de estrellas o quásares con puntos luminosos, poseen baja variación en los contrastes y la información se encuentra generalmente en bandas espectrales bajas y medias.

Atendiendo al trabajo de (Starck & Pantin, 2002) es posible hacer una buena estimación de la relación ruido señal de un filtrado paso alto de la imagen  $I$ , se hace la suposición de que a frecuencias suficientemente altas, lo que queda de la imagen filtrada será predominantemente ruido.

Lo que se propone en este trabajo es realizar una deconvolución usando un filtro de Wiener creado a partir de la información de alta frecuencia de la imagen ruidosa:

$$\tilde{\sigma}_e = \frac{\tilde{I}_i \cdot \tilde{P}_i^*}{|\tilde{P}_i|^2 + \lambda \tilde{H} \tilde{P}} \quad [14]$$

A continuación se ilustra cómo puede llevarse a cabo la estimación de ruido presente en las imágenes a través del filtrado paso alto:



Ilustración 5. Imagen original



Ilustración 7. Imagen con ruido

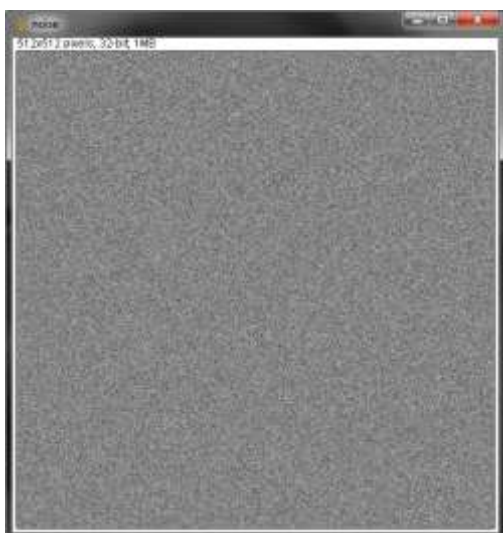


Ilustración 6. Ruido aditivo

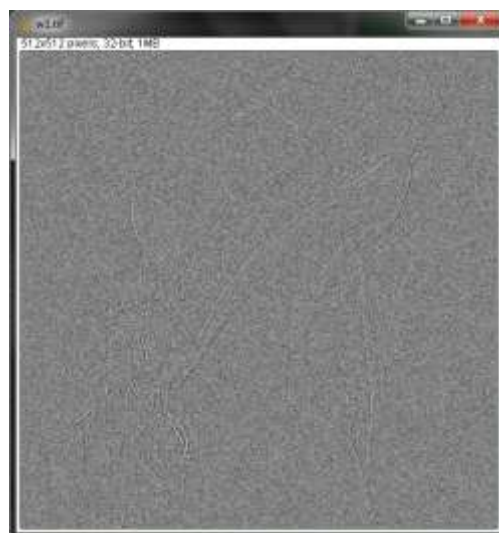


Ilustración 8. Filtrado Paso Alto de la imagen ruidosa

El factor  $\lambda$  sirve para modular el efecto de suavizado de la regularización pasoalto (High Pass). El valor óptimo de  $\lambda$  requiere el uso de métodos numéricos como la validación-cruzada y será específico al caso bajo estudio. Este método arroja por lo general buenos resultados pero las imágenes están demasiado suavizadas y esto puede ser su mayor contrapunto en imágenes en las que se quiere una nitidez alta.

El método se puede generalizar como:

$$\tilde{O}_e = \frac{\tilde{I}(u,v) \cdot \tilde{P}^*(u,v)}{|\tilde{P}(u,v)|^2 + \lambda \tilde{H}\tilde{P}(u,v)} \cdot \frac{P(u,v)}{P(u,v)} = \tilde{W}(u,v) \cdot \frac{\tilde{I}(u,v)}{\tilde{P}(u,v)} \quad [15]$$

Esto equivale al filtrado de Wiener cuando el filtro  $W$  depende a la vez del comportamiento de la señal y del ruido. El filtro  $W$  debe cumplir las condiciones:

1.  $|\tilde{W}(u,v)| \leq 1 \forall (u,v) > 0$
2.  $\lim_{(u,v) \rightarrow (0,0)} \tilde{W}(u,v) = 1 \forall (u,v)$  tal que  $\tilde{P}(u,v) \neq 0$

3.  $\frac{\tilde{W}(u,v)}{\tilde{P}(u,v)}$  acotado para cualquier par  $(u, v)$

Las ventanas más usadas son las conocidas: Gaussiana, Hamming, Hanning y Blackman. La función puede también derivarse directamente de la PSF.

Los métodos de regularización lineal tienen la ventaja de ser muy atractivos desde el punto de vista computacional y son por tanto buenos candidatos para computación en tiempo real. Más aun, tiene la ventaja de que el ruido presente en la solución se puede obtener y/o acotar con la expresión de  $W$  y el ruido presente en la imagen. Por ejemplo si se tratase una imagen con un ruido blanco gaussiano de desviación típica  $\sigma_d$  el ruido en la solución tendría por varianza:

$$\sigma_e^2 = \sigma_d^2 \sum W_k^2 \quad [16]$$

### 3.5 Reducción de ruido por promediado

La resolución angular de los telescopios terrestres está esencialmente limitada por las turbulencias atmosféricas. La resolución óptica está limitada en el orden de 1 arcsec sin importar el diámetro de la apertura, esto es debido a las fluctuaciones espaciotemporales del frente de onda.

En 1991, J. Primot, G. Rousset, y J. C. Fontanella propusieron un método para intentar reducir el error espectral que suponen las turbulencias usando un método de promediado y deconvolución. El principio del método que propusieron es el siguiente.

Partiendo de la ecuación [1] en la que se plantea el escenario inicial de los algoritmos entendido desde el punto de vista de una serie de imágenes y en el espacio de Fourier:

$$\tilde{I}_i = \tilde{O} \cdot \tilde{P}_i \quad [15]$$

Siendo  $\tilde{P}_i$  la OTF (Optic Transfer Function). Teóricamente se podría computar la  $O$  como la división de la imagen con la respuesta OTF, sin embargo, esto no será posible en aquellos puntos en los que la OTF presente valores nulos. Es más aconsejable considerar la expresión:

$$\tilde{O}_e = \frac{\sum_{i=1}^M \tilde{I}_i \cdot \tilde{P}_i^*}{\sum_{i=1}^M \tilde{P}_i \cdot \tilde{P}_i^*} \quad [16]$$

Si  $M$  tiende a infinito el denominador es igual a la función de transferencia del *speckle*. El comportamiento de la función puede ser aproximado por la expresión:

$$\sum_{i=1}^M |\tilde{P}_i|^2 = \left( \sum_{i=1}^M |\tilde{P}_i| \right)^2 + 0.342 \cdot \left( \frac{r_0}{D} \right)^2 T_0 \quad [17]$$

Dónde:

- $T_0$  es la OTF ideal del telescopio
- $r_0$  es el diámetro de la zona coherente en la pupila en presencia de turbulencia

- $D$  es el diámetro del telescopio

El segundo sumando representa la aportación del telescopio a la OTF final. Es posible colegir que la estimación  $O_e$  es la que minimiza el error cuadrático respecto del campo muestral disponible.

En el siguiente diagrama se refleja un diagrama que explica someramente el algoritmo:

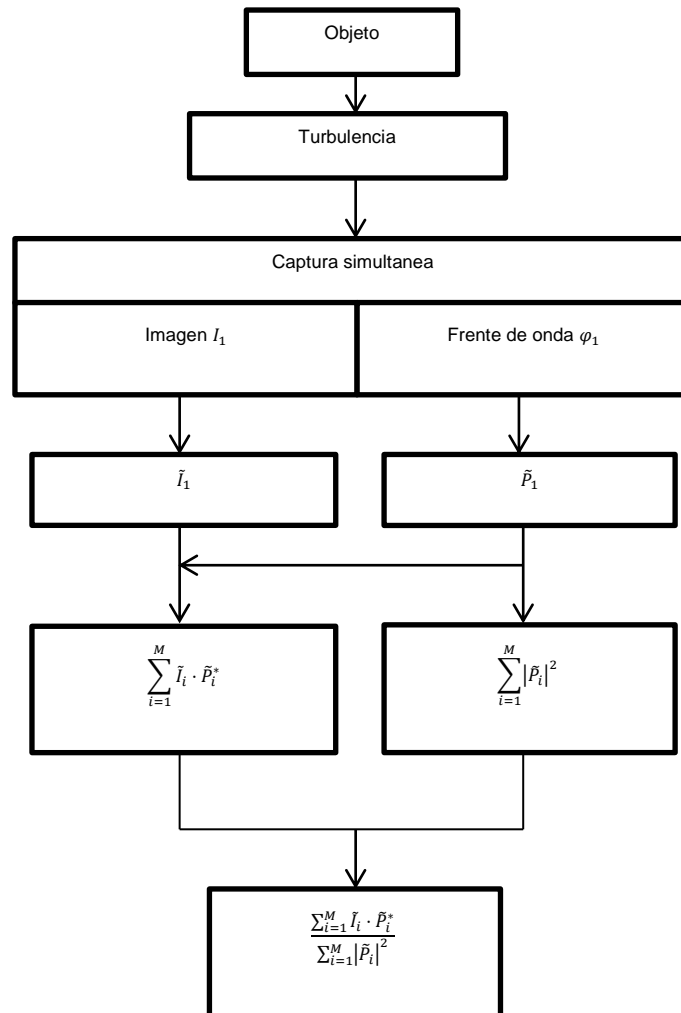


Ilustración 9. Diagrama de bloques algoritmo Primot

## 4. PARAMETROS DEL ESTUDIO

### 4.1 Creación del escenario de test

Para poder realizar una comparación de los algoritmos es necesario disponer de un marco común de evaluación. En este marco de evaluación, en lugar de emplear imágenes provenientes de instrumentación, se han desarrollado una serie de programas con la idea de tener la posibilidad de crear patrones específicos según el tipo de simulación del desempeño de los algoritmos que se quiera desarrollar.

Para crear patrones a voluntad serán necesarias varias herramientas:

1. Una herramienta para crear patrones de difracción perfectos. Estos simularán un **patrón controlado de luz** atravesando un sistema óptico sin aberraciones en el camino ni en el sistema instrumental.
2. Una vez que se tiene este patrón ideal, es necesario crear un patrón de dispersión similar a la aberración típica que introduce la atmosfera. Estos dos patrones juntos simulan un telescopio ideal con **condiciones controladas de seeing**.
3. Una herramienta para **añadir patrones de ruido** a la muestra para simular interferencias y ruido en la instrumentación de captura, ya sea óptica o del frente de onda. Para esto no es necesario crear ninguna macro, pues el proceso es bastante común y cuenta como parte de la funcionalidad principal de ImageJ.

Se aborda a continuación la creación de ambas herramientas.

#### 4.1.1 Generador de patrones de Airy

En primer lugar se trata de generar una imagen que simule el caso ideal en el que se enfoca una fuente puntual de luz en el infinito con un **telescopio limitado por difracción**, es decir, sin aberraciones y con unas condiciones de seeing perfectas.

Debido a la naturaleza ondulatoria de la luz, cuando ésta atraviesa una apertura circular se difracta produciendo un patrón de interferencia de regiones iluminadas y oscuras sobre una pantalla alejada de la apertura. A causa de esta difracción, el punto más pequeño en el que se puede enfocar un rayo de luz usando una lente tiene el tamaño concreto: es el tamaño del disco de Airy.

El resultado, en el caso de imágenes estelares, muestra una PSF afilada rodeada de anillos concéntricos, de acuerdo con el patrón de difracción esperado [4], como se observa en la ilustración:

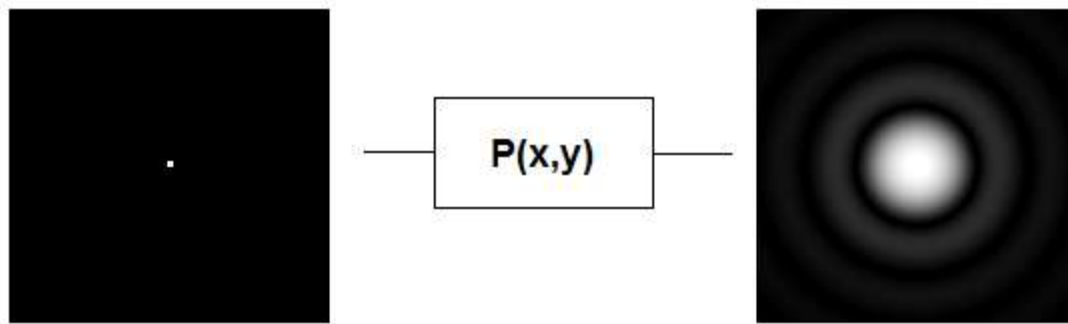


Ilustración 10: Caso ideal

Si suponemos un sistema óptico perfecto de radio  $a$ , su **función de dispersión puntual** (PSF) queda definida por la siguiente función:

$$P(\rho, \varphi) = \begin{cases} 0 & \text{si } \rho > a \\ 1 & \text{si } \rho < a \end{cases} \rightarrow P(x, y) = \begin{cases} 0 & \text{si } \sqrt{x^2 + y^2} > a \\ 1 & \text{si } \sqrt{x^2 + y^2} < a \end{cases} \quad [18]$$

Si trasladamos esta expresión de polares a coordenadas cartesianas y transformamos al espectro de Fourier obtenemos su **función de transferencia óptica** (OTF), que sería:

$$\mathcal{F}\{P(x, y)\} = \text{Airy}(x, y) = 2I_0 \frac{J_1\left\{\frac{1.22\pi}{R\alpha}\sqrt{(x-x_0)^2+(y-y_0)^2}\right\}}{\frac{1.22\pi}{R\alpha}\sqrt{(x-x_0)^2+(y-y_0)^2}} \quad [19]$$

Siendo:

- $I_0$  el valor máximo (central) de la fuente luminosa (estrella)
- $(x_0, y_0)$  es el punto central de la fuente luminosa
- $J_1$  la función de Bessel de primera especie y orden 1.
- $r_\alpha$  el radio del primer disco de Airy

Con combinaciones lineales de [18] es posible simular la aparición de cualquier número de estrellas de cualquier magnitud, en cualquier punto del plano imagen, estando el sistema de captura limitado únicamente por difracción.

Esta función sin embargo tiene la peculiaridad y la contrariedad de depender de la función de Bessel de primera especie y de primer orden:

$$J_\alpha(x) = \frac{1}{\pi} \int_0^\pi \cos(n\tau - x \sin(\tau)) d\tau \quad [20]$$

Estas funciones son de complicada computación analítica. Generalmente se suelen usar desarrollos en serie para aproximar estas funciones.

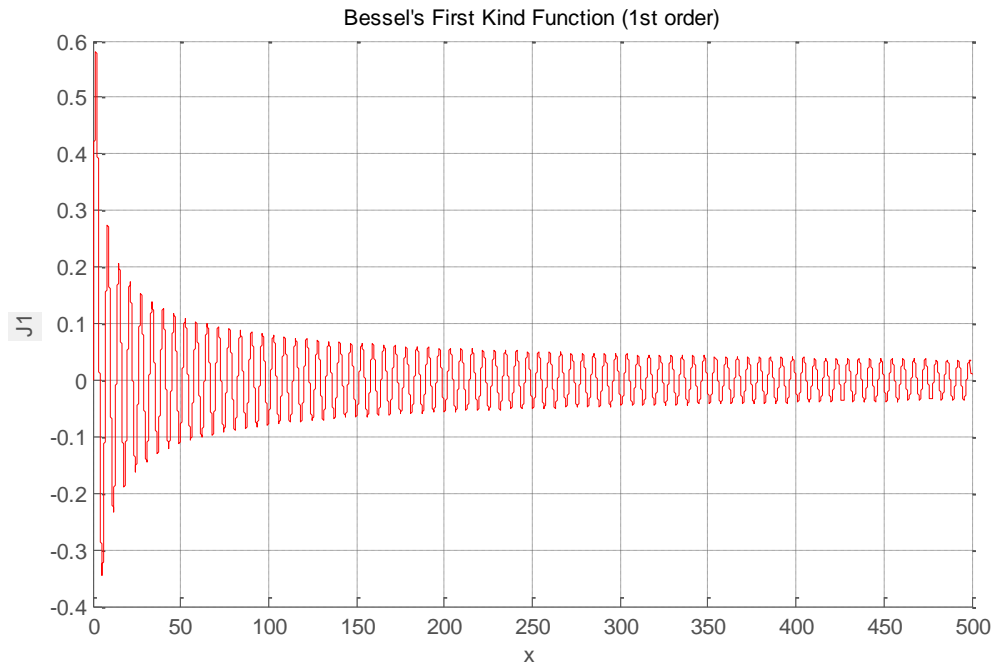


Ilustración 11: Función de Bessel de primera especie utilizada en el patrón

Para la computación de esta función se puede hacer una aproximación por polinomios de Taylor como:

$$J_{\alpha} = \sum_{m=0}^{\infty} \frac{(-1)^m}{m! \Gamma(m+\alpha+1)} \left(\frac{x}{2}\right)^{2m+\alpha} \quad [21]$$

Donde  $\Gamma(x)$  es la función de Euler y se define como  $\Gamma(x) = (x-1)!$ , quedando si sustituimos en la expresión:

$$J_1 = \sum_{m=0}^M \frac{(-1)^m}{m!(m+1)!} \left(\frac{x}{2}\right)^{2m+1} \quad [22]$$

Este desarrollo en serie de Taylor puede aproximar la función de Bessel tan bien como sea deseable, sin embargo hay que ser muy cuidadoso en cuanto a la precisión. Como es evidente, **no** se puede dar un **número demasiado alto a M**, pues el **coste computacional** sería inasumible. **Tampoco** se puede fijar un tamaño de **M muy pequeño** pues se **dispararía el error** de aproximación si el dominio de la función se sale del rango adecuado de aproximación.

Una aproximación empírica para el número adecuado de términos de Taylor nacería de examinar el rango de entrada de la función de Bessel en nuestra aplicación:

$$x = \frac{1.22\pi}{r_a} \sqrt{(x_{imagen} - x_0)^2 - (y_{imagen} - y_0)^2} \quad [23]$$

En nuestro caso:

$$x \in \left[0, \frac{1.22\pi}{r_a} \sqrt{(512 - x_0)^2 - (512 - y_0)^2}\right] \quad [24]$$



Con lo que se usará un número de términos de Taylor igual o ligeramente superior al valor máximo de [23].

Teniendo en cuenta que la computación de la escena requiere una evaluación de la función de Bessel por cada pixel y foco luminoso, amén de otras multiplicaciones y divisiones, y que cada función de Bessel requiere por su parte la suma de un número de componentes de Taylor de al menos:

$$n = \left\lceil \frac{1.22\pi}{r_a} \sqrt{(512 - x_0)^2 + (512 - y_0)^2} \right\rceil \quad [25]$$

Es sencillo imaginar que el coste computacional es enorme del algoritmo implementado mediante extensión de polinomios es inasumible para imágenes de 512x512 pixeles.

Por esta razón y para intentar dar sencillez y rapidez al script, se ha generado usando MATLAB una tabla de búsqueda con valores equiespaciados de la función de Bessel. Con MATLAB se generan 1 millón de valores de la función en el rango 0 a 500 (sobra con 500 en vez de 512 ya que se ha establecido un centro de referencia de al menos 15 pixeles) aunque la modificación de la tabla no requiere cambios muy complejos.

#### Código MATLAB:

```
1 x= linspace(0,500,1e6);
2 J1= besselj(1,x);
3
4 fi = fopen('lut.txt', 'w');
5 for i=1:length(y)
6     fprintf(fi, '%d \n', J1(i));
7 end
8 fclose(fi);
```

Una vez dispuestos los valores separados por retorno de carro en un archivo de texto tan solo es necesario hacer un acceso al archivo desde la macro de ImageJ por cada evaluación de la función, reduciendo así drásticamente el coste computacional del algoritmo resultante.

Con esto ya es posible simular la aparición de cualquier número de estrellas de cualquier magnitud, en cualquier punto del plano imagen. Se va a recrear un escenario con seis estrellas de magnitudes descendientes tabuladas de izquierda a derecha en dos filas y tres columnas. El resultado es el siguiente:

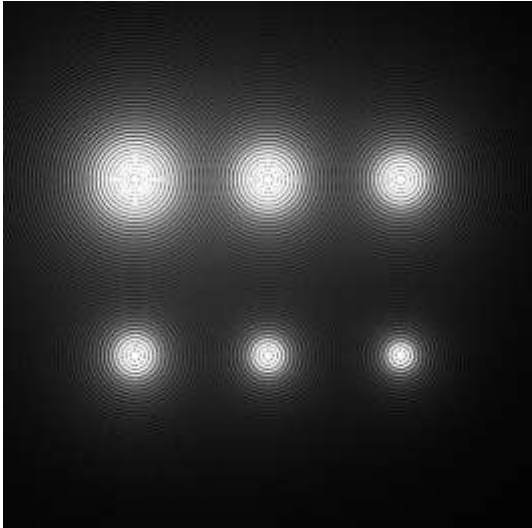


Ilustración 12: Escena inicial sin ajuste de contraste

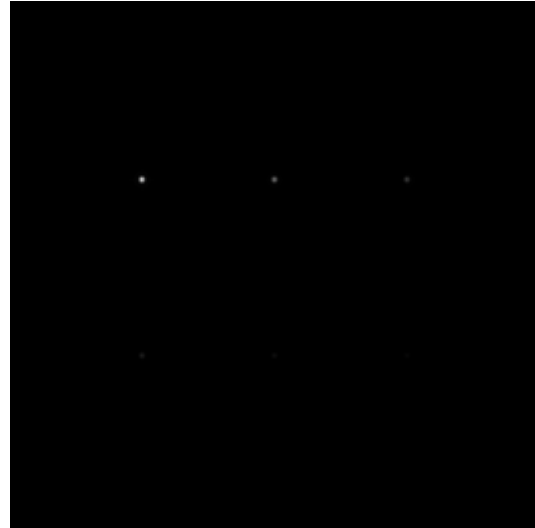


Ilustración 13: Escena original cubriendo todo el rango  
(adjusted contrast)

En esta imagen, antes de hacer ningún ajuste de contraste se pueden observar los discos de Airy con más claridad.

Esta será la imagen de referencia para todos los algoritmos, es decir, el algoritmo perfecto, de existir, devolvería esta imagen.

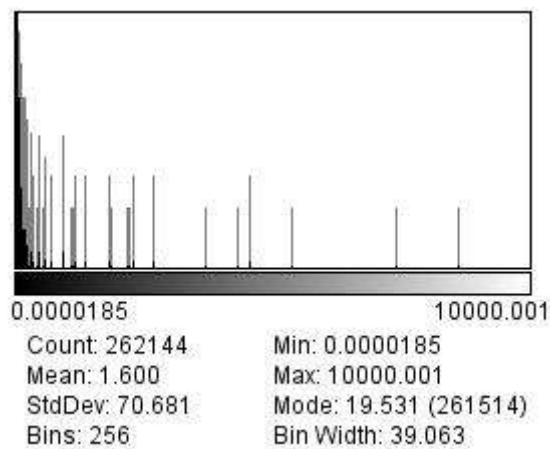


Ilustración 14. Histograma del caso ideal

### 4.1.2 Generador de patrones de dispersión

En el epígrafe anterior se ha obtenido la escena de estrellas como se vería por un telescopio limitado solo por difracción, con un seeing nulo. Para probar los algoritmos de reducción de ruido y cancelación del efecto seeing, es necesario primeramente desfigurar esta imagen como lo haría la atmosfera.

Las deformaciones mencionadas se representan con asiduidad con una PSF que expresa la distorsión del frente de onda de la luz al atravesar espacios con muy distintos índices de refracción. Para la medida del frente de onda se ha usado un sensor Hartmann-Shack, capaz de recrear el frente de onda en términos de los polinomios de Zernike.



Ilustración 15: Sensor Frontsurfer de OKO TECHNOLOGIES

Este hardware da como resultado un archivo de texto que contiene un número configurable de coeficientes de los polinomios de Zernike para el frente de onda medido:

r.txt	Ejemplo del informe del sensor Hartman-Shack
<pre>Parameters: ----- Decomposition area diameter = 6.160764e+000 mm Aperture: circular, auto <b>Diameter = 5.852726e+000 mm</b> Reconstructed from 18 wavefront slopes Wavelength = 6.300000e-001 mic Strehl factor: 9.852244e-001 Phase peak-to-valley = 1.097927e-001 waves Phase RMS = 1.729108e-002 waves Distance to focus = 2.229085e+002 m Beam displacement:   DX = 4.881296e-003 mm, DY = 1.491078e-003 mm  Zernike coefficients: ----- C[1,1] = <b>-1.354259e-002</b> waves (tip) C[1,-1] = <b>-1.809017e-002</b> waves (tilt) C[2,0] = <b>1.689204e-002</b> waves (focus) ***C[2,2] = <b>3.198823e-002</b> waves (astigmatism) C[2,-2] = <b>-3.073708e-002</b> waves (astigmatism) C[3,1] = <b>-5.372054e-003</b> waves (coma)</pre>	

```
C[3,-1] = 8.811838e-003 waves (coma)
C[3,3] = 7.557883e-003 waves (trifoil)
C[3,-3] = -1.759460e-002 waves (trifoil)
C[4,0] = 1.133886e-002 waves (spherical aberration)

Zernike polynomials are in the form given in the manual of
FrontSurfer.
Decomposition is made over the decomposition area (outer red circle).

[...]
```

Este informe sirve de entrada a nuestra herramienta para generar la función pupila. Los polinomios de Zernike son un conjunto ortonormal usados en el desarrollo de una función de frente de onda en sistemas ópticos con pupilas con simetría circular. Normalmente suelen estar definidos en coordenadas polares ( $\rho, \phi$ ).

Los primeros 10 términos de los polinomios de Zernike sin normalizar usados son:












	$Z_0 = 1$		$Z_6 = \rho(-2 + 3\rho^2) \cos(\phi)$
	$Z_1 = \rho \cos(\phi)$		$Z_7 = \rho(-2 + 3\rho^2) \sin(\phi)$
	$Z_2 = \rho \sin(\phi)$		$Z_8 = 1 - 6\rho^2 + 6\rho^4$
	$Z_3 = -1 + 2\rho^2$		$Z_9 = \rho^3 \cos(\phi)$
	$Z_4 = \rho^2 \cos(2\phi)$		$Z_{10} = \rho^3 \sin(\phi)$
	$Z_5 = \rho^2 \sin(2\phi)$		

Tabla 1. Tabla de polinomios de Zernike

Sumando las aportaciones de cada término pesadas respecto a cada coeficiente obtenemos la función pupila. Para este caso concreto, es decir el caso de los coeficientes del informe de prueba, se reconstruye una función de pupila con la siguiente forma:

$$F_{pupila} = \sum_{i=0}^N C_n Z_n$$

**Ecuación.** Conviene reseñar que los índices de los coeficientes de Zernike coinciden en orden con los ilustrados en la tabla 1, por ejemplo  $C[1,1]$  será  $\rho \cos(\phi)$  y  $C[1,-1]$  será  $\rho \cos(-\phi) = \rho \sin(\phi)$ .

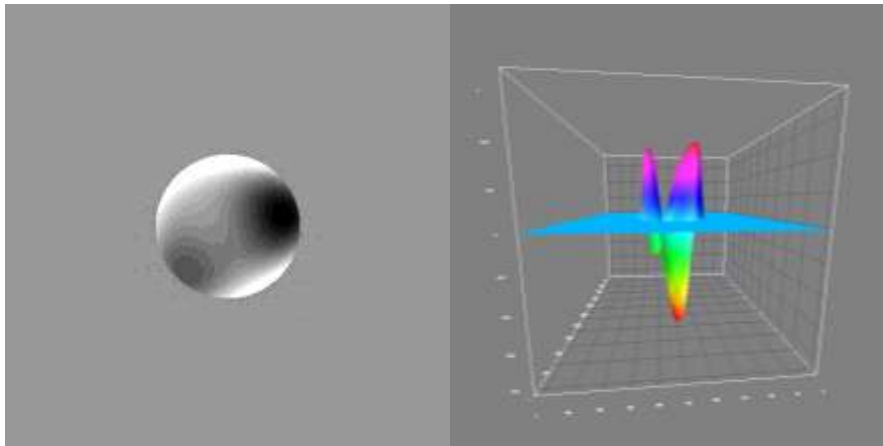


Ilustración 16. Visualización de la función pupila en 2D y 3D

Una vez que se dispone de la función pupila  $F_{pupila}$  hay que asegurar su continencia al círculo unidad y obtener la función pupila generalizada:

$$F_g(x, y) = unit(x, y) \cdot e^{-\frac{i2\pi}{waves} F_{pupila}(x, y)}$$

$$unit(x, y) = \begin{cases} 1 & \text{si } \sqrt{x^2 + y^2} \leq 1 \\ 0 & \text{si } \sqrt{x^2 + y^2} > 1 \end{cases}$$

Una vez hecho esto, basta con reconstruir la PSF como:

$$PSF = \left( \sqrt{\text{Re}\{FFT\{F_g(x, y)\}\} + \text{Im}\{FFT\{F_g(x, y)\}\}} \right)^2$$

Además de la herramienta que genera la PSF desde los informes del hardware de instrumentación sensor de onda, resultaba interesante crear una herramienta capaz de crear PSF aleatorias o modeladas.

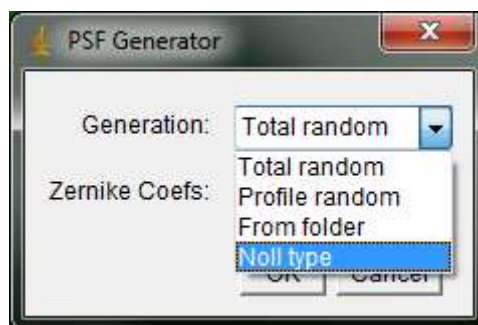


Ilustración 17. Captura del interfaz indicando los modos

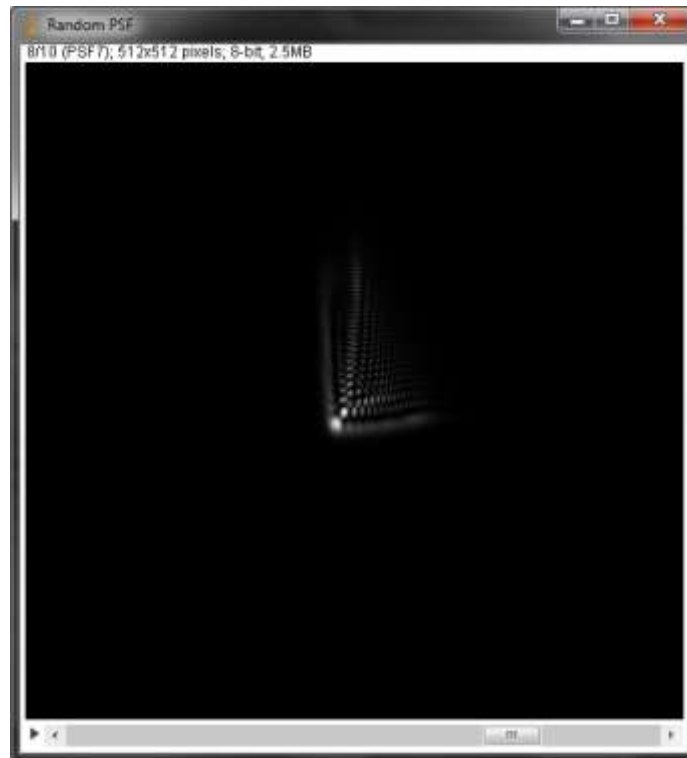


Ilustración 18. Ejemplo de una PSF totalmente aleatoria

Existen estudios sobre el tipo de deformaciones que supone la atmosfera en los patrones de difracción. Según Robert J. Noll en su estudio (Noll, 1976), en el que se realizó un estudio estadístico exhaustivo de la distribución que siguen los polinomios de Zernike de la descomposición del frente de onda que supone la atmosfera típicamente.

Del estudio se pueden calcular las desviaciones típicas para los 10 primeros coeficientes de Zernike típicos de frente de onda atmosférico son:

$$\begin{aligned}\sigma_0 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.671507 & \sigma_4 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.152721 & \sigma_8 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.0788645 \\ \sigma_1 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.671507 & \sigma_5 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.0788645 & \sigma_9 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.04964 \\ \sigma_2 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.152721 & \sigma_6 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.0788645 & & \\ \sigma_3 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.152721 & \sigma_7 &= \frac{D^{\frac{5}{6}}}{r_0} \cdot 0.0788645 & & \end{aligned}$$

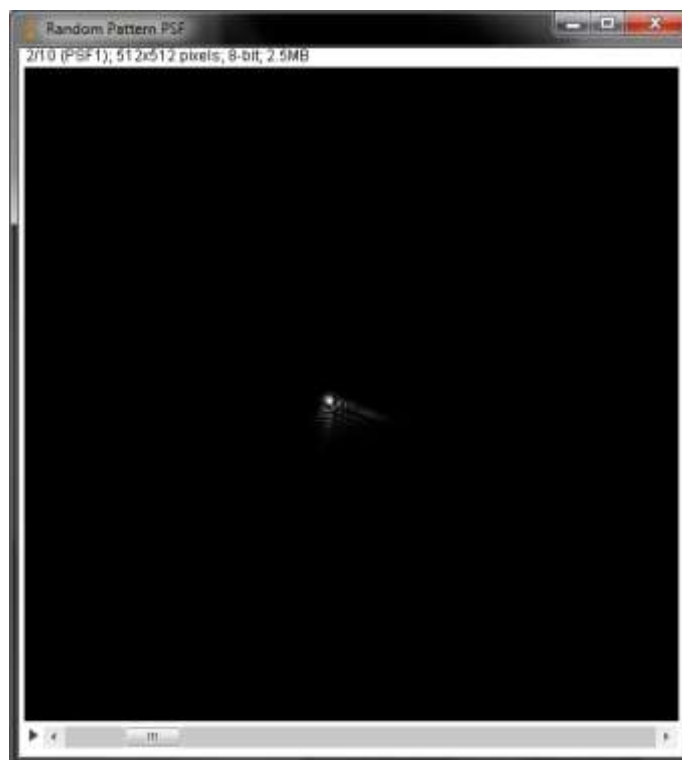


Ilustración 19. Ejemplo de una PSF generada con el modelo de Noll

En la herramienta desarrollada se ha incluido un generador de funciones puntuales de dispersión correspondientes a frentes de onda atmosféricos que sigue los modelos estadísticos de Noll y Kolmogorov (similares a una PSF real), pero también existe la posibilidad de usar una reconstrucción de sensor con modificaciones aleatorias, es decir, usar el sensor como un patrón y generar desde esta una batería de PSF más o menos aleatorias. En la herramienta una vez que se ha elegido el archivo r.txt que contiene los coeficientes que servirán como patrón es posible elegir el grado de similitud que ofrecerán las PSF generadas con el patrón así como el número de PSFs.



Ilustración 20. Ajuste de verosimilitud respecto al patrón



**Ilustración 21.** PSF con patrón desde instrumentación

Se ha creado una herramienta para generar PSFs, por un lado para reconstruir las PSF que provengan del sensor del frente de onda y por otro para generar PSF de patrones típicos atmosféricos. De esta manera en el caso de carecer de un hardware de detección del frente de onda se puede utilizar esta herramienta para estimar una PSF del estado de la atmosfera en el momento de la captura.

Resumiendo las opciones del software son:

- Total random: PSF totalmente aleatoria.
- Profile random: PSF aleatorizada desde una detectada.
- From folder: Esta opción escanea un directorio y crea una PSF por cada informe que encuentre.
- Noll type: Esta opción sigue los estudios estadísticos de Noll para crear una PSF atmosférica típica.

Para aplicar aberraciones a la imagen de partida de las seis estrellas, bastara con convolucionar con una PSF similar a la que presenta típicamente una atmosfera turbulenta.



## 4.2 Enumeración de los algoritmos en prueba

Los algoritmos evaluados se pueden resumir en la siguiente tabla comparativa:

Métodos de regularización lineal: Wiener, Tikhonov

Métodos de promediado: Primot y FASTCAM

Mezclas: promediar métodos de regulación lineal

Ref	Nombre	Fórmula
1	<b>FASTCAM</b>	Recentrar y acumular. Sin deconvolución.
2	<b>Estándar Postpromediado</b>	$\bar{O}_e = \left\langle \frac{\tilde{I}_i}{\tilde{P}_i} \right\rangle$
3	<b>Wiener Postpromediado</b>	$\bar{O}_e = \left\langle \frac{\tilde{I}_i \cdot \tilde{P}_i^*}{ \tilde{P}_i ^2 + \overline{SNR}_i} \right\rangle$
4	<b>Primot Postpromediado</b>	$\bar{O}_e = \left\langle \frac{\tilde{I}_i \cdot \tilde{P}_i^*}{ \tilde{P}_i ^2} \right\rangle$
5	<b>Estándar Prepromediado</b>	$\bar{O}_e = \frac{\langle \tilde{I}_i \rangle}{\langle \tilde{P}_i \rangle}$
6	<b>Wiener Prepromediado</b>	$\bar{O}_e = \frac{\langle \tilde{I}_i \cdot \tilde{P}_i^* \rangle}{\langle  \tilde{P}_i ^2 + \overline{SNR} \rangle}$
7	<b>Primot Prepromediado</b>	$\bar{O}_e = \frac{\langle \tilde{I}_i \cdot \tilde{P}_i^* \rangle}{\langle  \tilde{P}_i ^2 \rangle}$
8	<b>Wiener con regularización paso alto</b>	$\bar{O}_e = \frac{\langle \tilde{I}_i \cdot \tilde{P}_i^* \rangle}{\langle  \tilde{P}_i ^2 + \gamma \overline{HP} \rangle}$

Tabla 2. Tabla comparativa de los algoritmos a evaluar

## 5 ESTUDIO DE CORTA ESCALA

### 5.1 Comparativa visual Tip&Tilt de 20 muestras

#### 5.1.1 Niveles de ruido

Con el propósito de testear los algoritmos se ha creado como se ve en la imagen 13 un escenario con seis estrellas cuyos valores máximos se representan en la siguiente proyección 3D de la imagen 13:

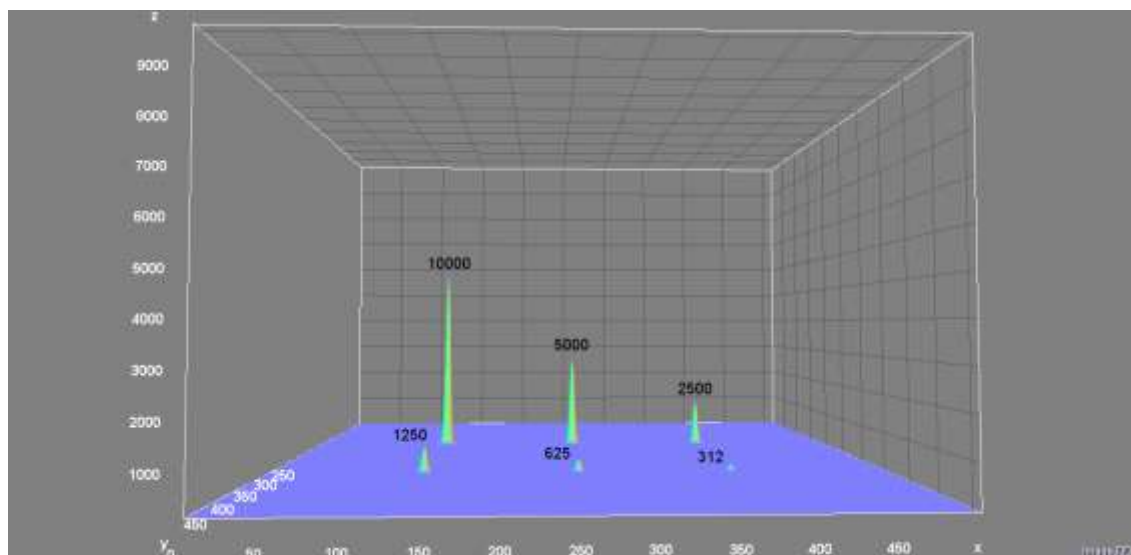


Ilustración 22. Niveles de intensidad de las seis estrellas originales

Con estos niveles se decide barajar un único nivel de ruido, que se equipare en media a la magnitud de la tercera estrella en magnitud.

Una vez que se aplican operaciones sobre los niveles de intensidad de la imagen original estos cambian y es necesario saber el rango de la intensidad del caso ejemplo:

Original	Convolucionada
10000	40877210
5000	20403546
2500	10202132
<b>1250</b>	<b>5100956</b>
625	2557671
312,5	1273354

Tabla 3. Valores de intensidad de la muestra típica en caso Tip&Tilt

En las siguientes imágenes se muestran lo que llegaría al telescopio en el caso de tener una deformación con Tip&Tilt según el nivel de ruido arriba establecido.

Se ha recreado una imagen con un ruido de desviación típica del mismo valor aproximado de la tercera estrella en magnitud  $\sigma_n = 5 \cdot 10^6$ . Quedando la imagen que llega al telescopio como sigue:



**Ilustración 23.** Una de las imágenes ruidosas  $\sigma_n = 5 \cdot 10^6$

Se puede apreciar, tal como cabía esperar, solo se pueden apreciar las tres estrellas superiores que tienen una magnitud superior a la del ruido.

A continuación se muestran los resultados de los algoritmos evaluados.

### 5.1.2 Resultado algoritmo FASTCAM [1]



Ilustración 24. Resultado de FASTCAM TipTilt

Podemos ver en este caso como se ha obtenido un buen resultado con el algoritmo original de FASTCAM. Este resultado se debe a que la PSF aplicada solo contiene aberraciones de primer orden, esto es, tip and tilt, o lo que es lo mismo desplazamientos en x-y, pero no tiene aberraciones de mayor orden (astigmatismo, coma...). Al solo verse errores de desplazamiento el algoritmo puede tomar el 100% de las imágenes como imágenes buenas y así obtener un buen resultado al recentrar y sumar.

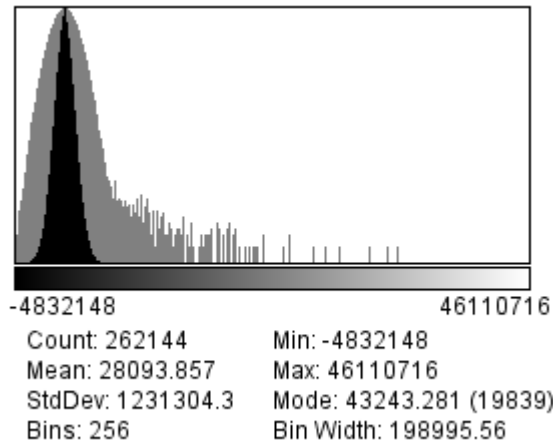


Ilustración 25. Histograma FASTCAM TipTilt

### 5.1.3 Resultado deconvolución estándar [2] y [5]

Ante estos resultados vamos a ver a continuación como el método del cociente de Fourier o deconvolución simple no llega a soportar estos niveles de ruido.

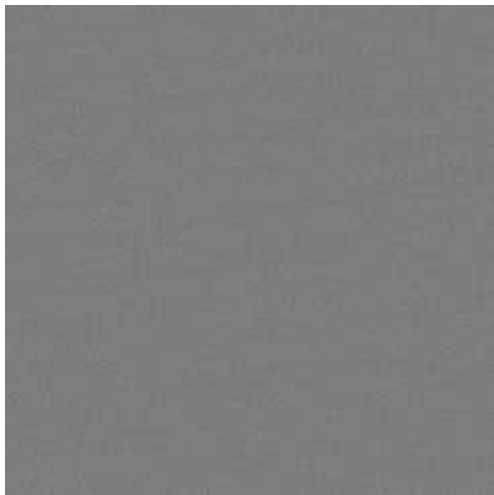


Ilustración 26. Deconvolución estándar con Prepromediado en TipTilt

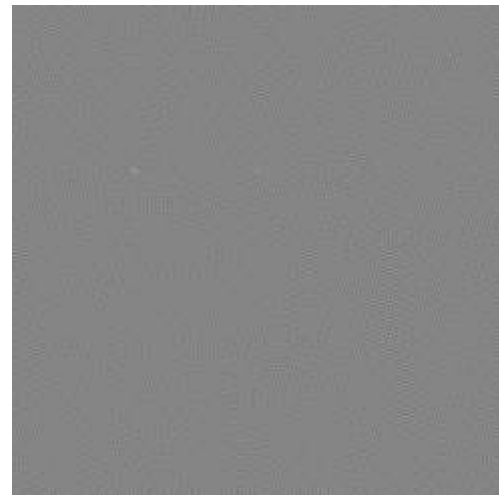


Ilustración 27. Deconvolución estándar con Postpromediado en TipTilt

En las imágenes se aprecia que el ruido ha desbordado los niveles de señal y apenas se puede apreciar la estrellad de mayor intensidad.

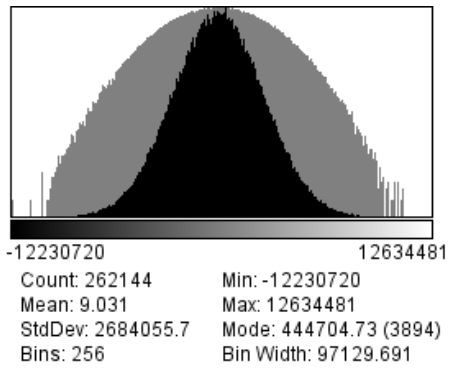


Ilustración 28. Histograma std Prepromediado TipTilt

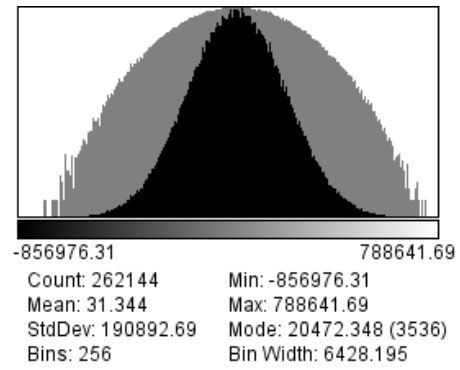


Ilustración 29. Histograma std Postpromediado TipTilt

En los histogramas se aprecia que lo que se ha recuperado es la campana gaussiana típica que corresponde a la distribución del ruido.

#### 5.1.4 Resultado deconvolución Primot [4] y [7]

A continuación vamos a ver cómo responde el método de promediado propuesto por (Primot, Rousset, & Fontanella, 1990).

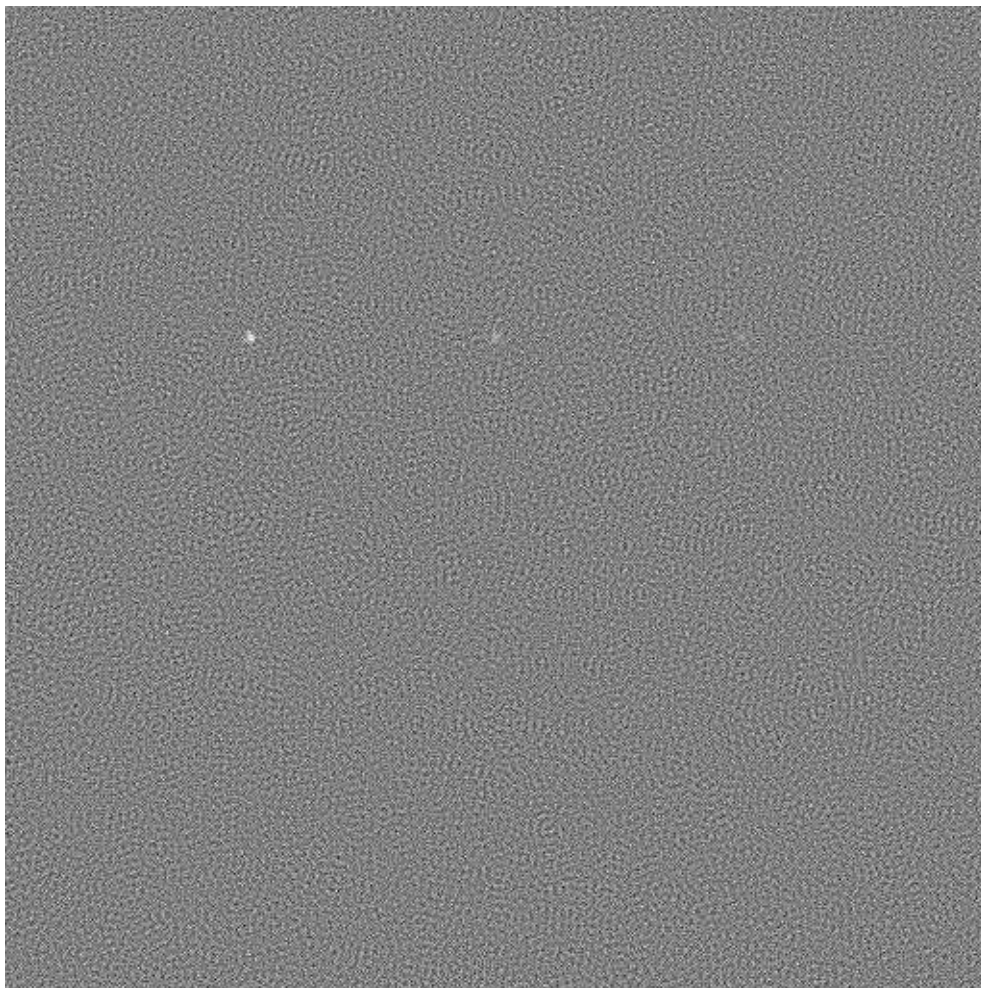
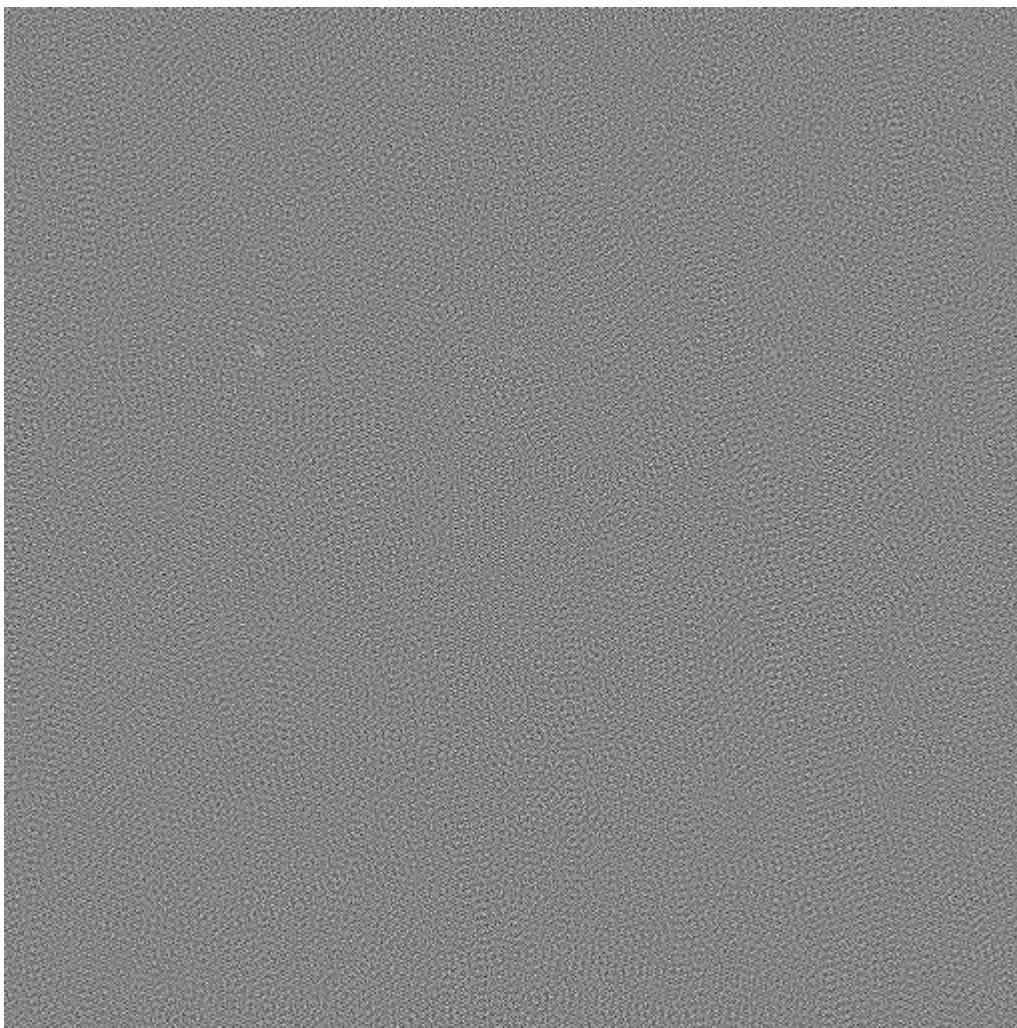


Ilustración 30. Resultado de algoritmo Primot (prepromediado) TipTilt

Se puede ver que en este caso el ruido no se ha sobrepuesto tanto a la señal como en el caso anterior aun así el resultado es pobre. Esto se debe a que el algoritmo no tiene ningún filtro para regularizar y la única manera que tiene de reducir el ruido sería mediante un número mayor de muestras.

Como una prueba de referencia y por tanto, no incluida en los trabajos de (Primot, Rousset, & Fontanella, 1990), se ha modificado el algoritmo para que el promediado de las imágenes se haga con carácter posterior a deconvolución (caso 7 de la *tabla 2*), el resultado es el siguiente:



**Ilustración 31.** Modificación de Postpromediado de Primot TipTilt

J.L. Primot afirmaba en su obra que el hecho de realizar el promediado antes de realizar la división en espectro (la deconvolución en sí) mejoraba los niveles de señal respecto a ruido, y en efecto se puede apreciar como el resultado es algo mejor visualmente hablando.





### 5.1.5 Resultado deconvolución de Wiener con regularización constante [3] y [6]

Dejando a un lado los resultados de los métodos de promediado, los métodos de regularización lineal se espera que tengan un efecto corrector que ayude a minimizar los efectos del ruido en la deconvolución.

En un primer lugar se supondrá que el método de regularización de Wiener tiene una idea muy acertada del ruido que está afectando a la imagen. En este caso se fijará lo mejor que cabe esperar del método de regularización de Wiener, pero hay que tener en mente en que en la práctica no se podrá llegar a tener una estimación tan adecuada del ruido que disturba la imagen.

El resultado que arroja la deconvolución con el filtro de Wiener y un prepromediado antes de dividir en espectro es el siguiente:

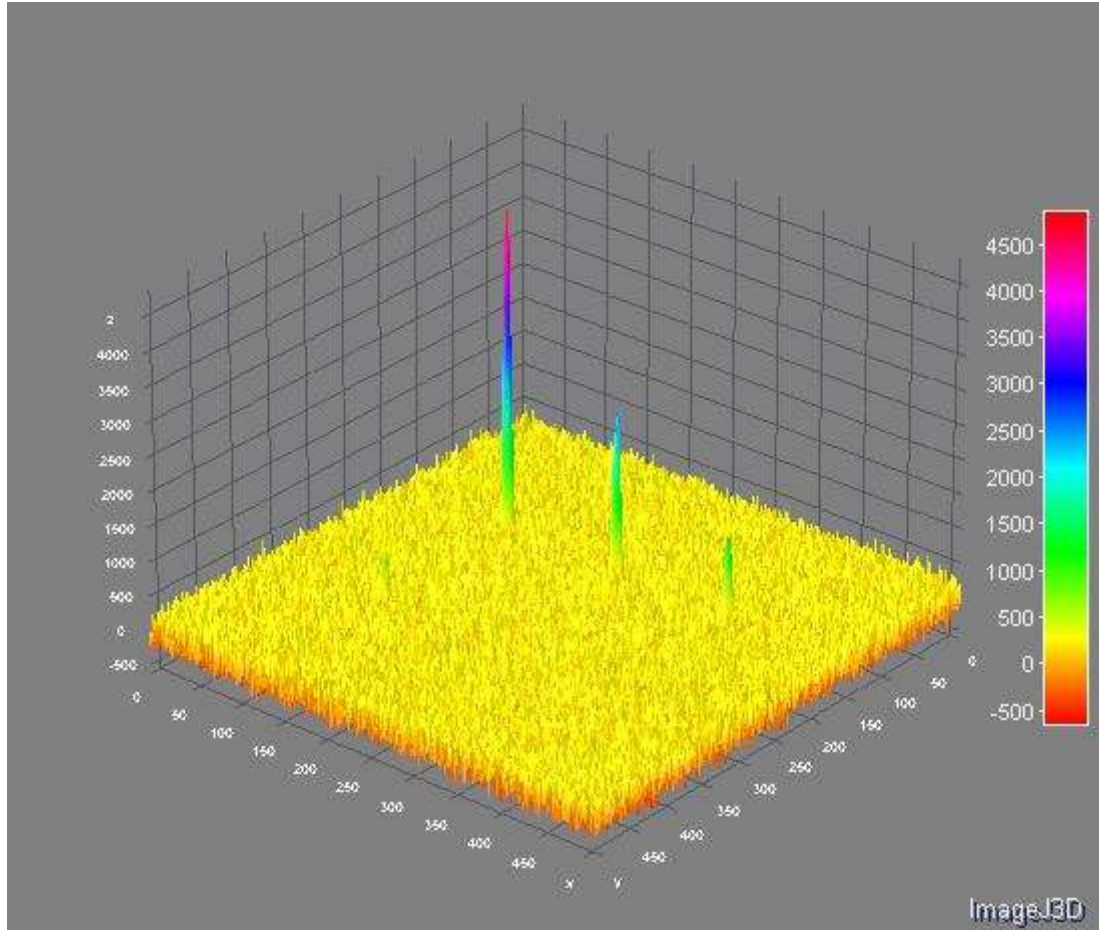


Ilustración 34. Resultado Wiener Prepromediado TipTilt ( $\gamma = 5.2 \cdot 10^6$ )

Como se puede observar en la imagen las estrellas aparecen de una manera más intensa, se puede observar la cuarta estrella e incluso intuir un poco de la última. Se

puede ver que el ruido es más granulado, de manera que obviamente se ha visto suavizado.

Merece la pena echar un vistazo a la representación en tres dimensiones de la imagen:



**Ilustración 35.** Representación 3D Wiener Prepromediado TipTilt ( $\gamma = 5.2 \cdot 10^6$ )

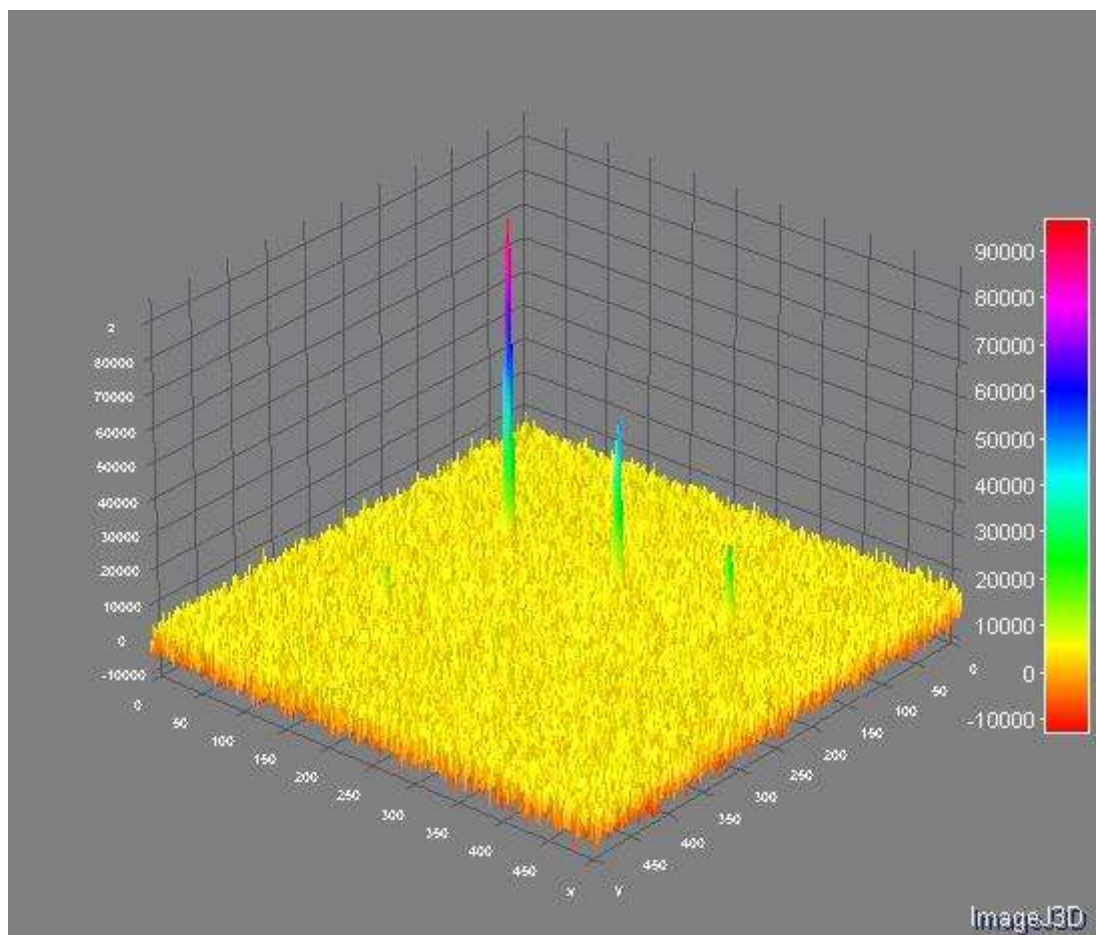
En el caso de usar un promediado posterior a la división en espectro se obtendrá:



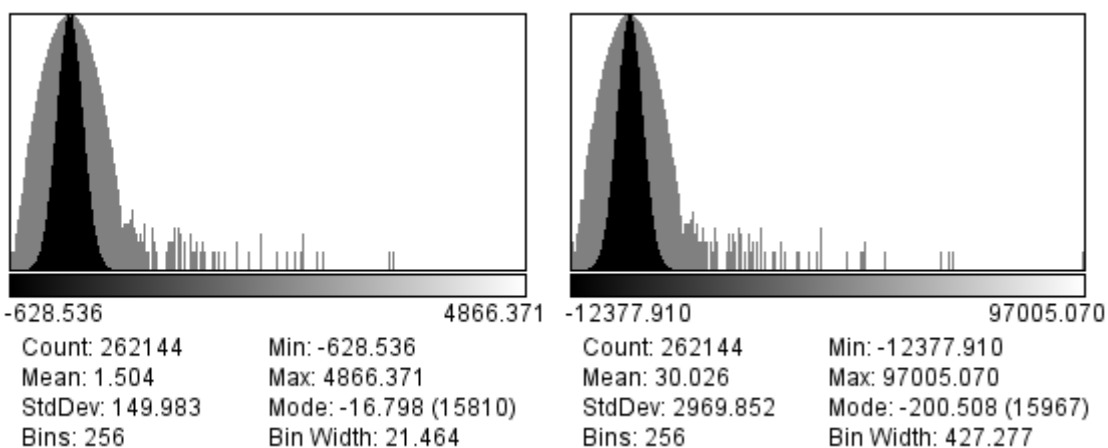
Ilustración 36. Wiener Postpromediado TipTilt ( $\gamma = 5.2 \cdot 10^6$ )

El parámetro de regularización (en este caso fijo en todos los valores del espectro) consigue unos buenos resultados gracias a que la distribución de ruido gaussiano no presenta una forma muy afilada. Según textos sobre regularización lineal en deconvolución una distribución distinta (poissoniana) podría haber dado un peor resultado con una regularización constante.

Aunque la diferencia no sea importante, solo para respetar la estructura se presenta la visualización 3D:



**Ilustración 37.** Representación 3D de Wiener Postpromediado TipTilt



**Ilustración 38.** Comparación de histogramas, a la izquierda prepromediada, a la derecha postpromediada.

En los histogramas se puede apreciar de nuevo un estrechamiento de la campana de ruido. Cabe reseñar que este método ha generado valores negativos, que de por si se sabe que no son posibles. Esto ocurre por la regularización, que hace que ciertos valores se hagan negativos. Uno de los problemas de la regularización lineal es la imposibilidad de incluir esta información a priori que de la que se dispone.

Con el objetivo de comprobar las posibles diferencias en cuanto a la variación del parámetro de regularización se presentan tres resultados en la regularización:



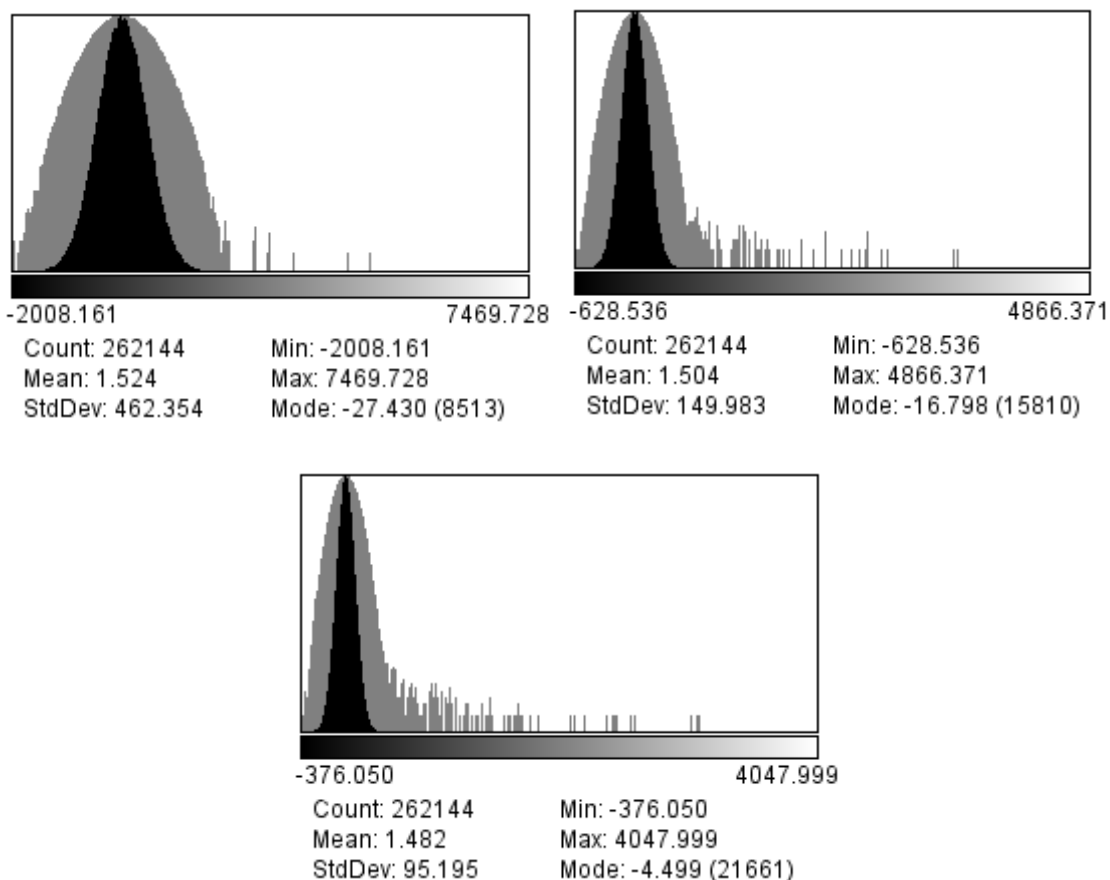
Ilustración 39. Arriba  $\gamma = 1 \cdot 10^6$ , medio  $\gamma = 5.2 \cdot 10^6$ , abajo  $\gamma = 10 \cdot 10^6$

Se puede observar como al aumentar el parámetro de regularización la imagen presenta menos ruido, pero por contrario también supone una reducción de contenidos de la señal haciendo que esta se difumine, incluso que dé lugar a la aparición de anillos y oscilaciones de Gibbs como se empieza a ver en la primera estrella en intensidad del caso  $\gamma = 5.2 \cdot 10^6$  y del posterior.



**Ilustración 40.** Comparación del efecto del parámetro de regularización, de izquierda a derecha se aumenta el parámetro

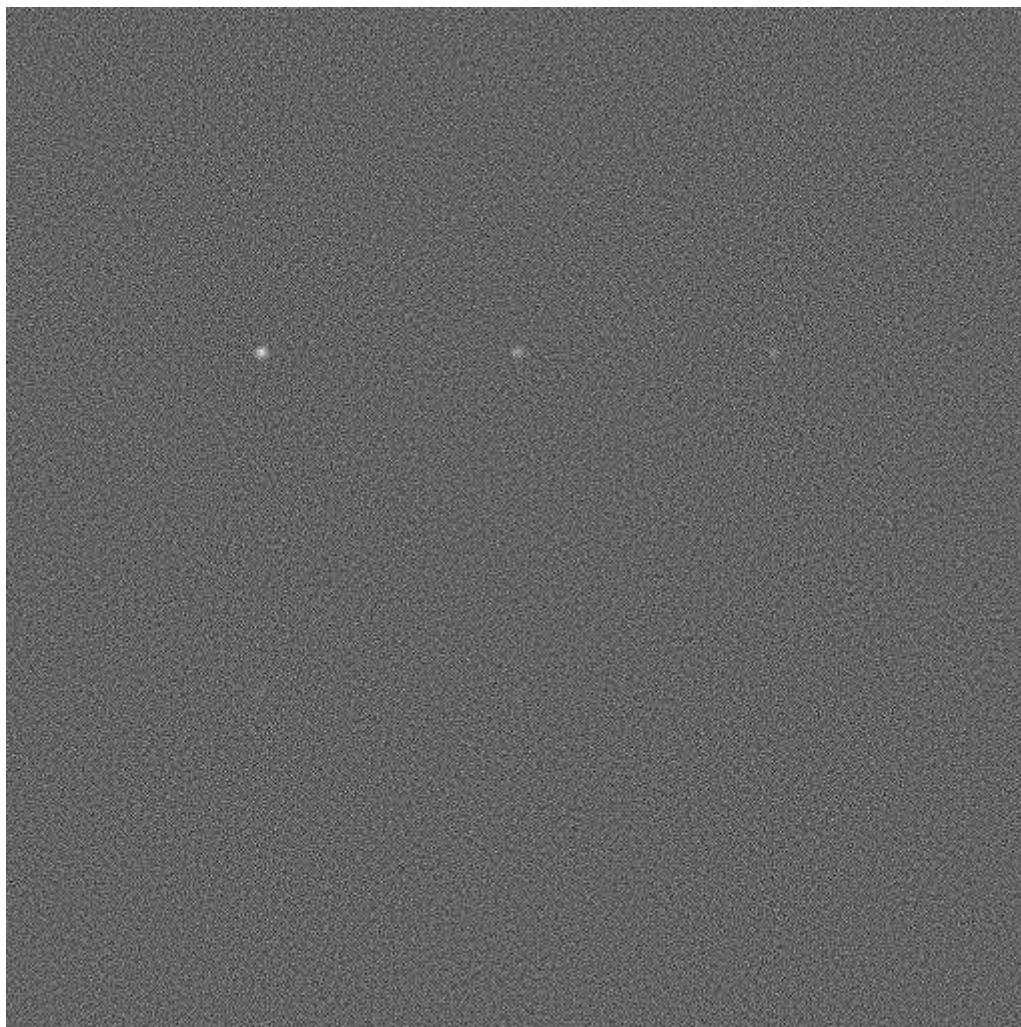
Así se podría decir que el parámetro de regularización supone un compromiso entre la fidelidad a los datos originales y el grado de cancelación de ruido. El efecto sobre el ruido también se observa en los histogramas:



**Ilustración 41.** Comparación de histogramas en variación del parámetro  $\gamma$

### 5.1.6 Resultado deconvolución de Wiener con regularización paso alto [8]

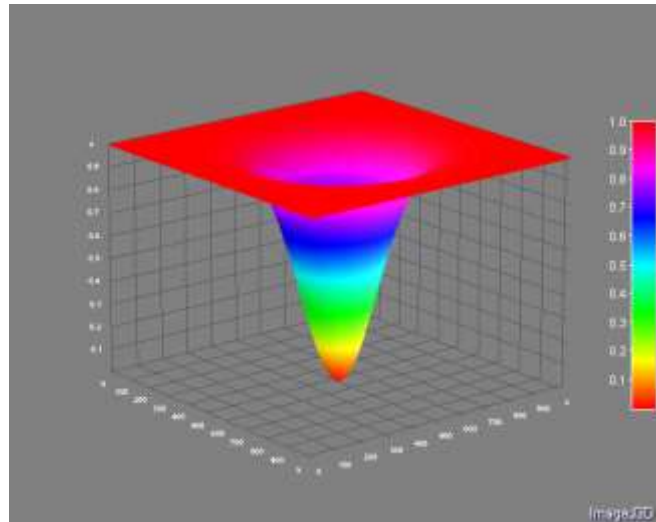
En este caso se va a usar un filtro paso alto como definidor del parámetro de regularización. La imagen que se obtiene es:



**Ilustración 42.** Resultado de regularización paso alto en el modelo tip tilt de 20 muestras

En este caso no se ha obtenido tan buen resultado como era de esperar. La regularización en este caso viene dada por un filtro de menos de 3 píxeles.

Conviene resaltar que este método no ha reseñado un buen desempeño en el caso de modelo tipilt, pero que este es algo superior en caso de usar modelos atmosféricos.



**Ilustración 43.** Imagen del filtro de 3píxeles aplicado

Otra posibilidad alternativa a usar la función de transferencia de un filtro paso alto es utilizar un filtrado paso alto de la imagen manchada para quedarnos con el contenido de alta frecuencia y usarlo de regulador. Como es sabido este contenido de alta frecuencia contiene las variaciones más bruscas, que con mayor frecuencia pertenecerán a cambios producidos por el ruido. De esta manera el método se postularía como la mejor manera de realizar un tratamiento con un solo ejemplar.



## 5.2 Comparativa visual de modelo atmosférico de 20 muestras

### 5.2.1 Niveles de ruido

Una vez que se convoluciona con las PSFs generadas por el modelo atmosférico se obtienen unos niveles:

Original	Convolucionada
10000	33455972
5000	16728460
2500	8364227
<b>1250</b>	<b>4182253,75</b>
625	2091280,75
312,5	1043987,563

Tabla 4. Valores de intensidad de la muestra típica en caso atmosférico

Conviene señalar que los niveles pueden verse sensiblemente afectados por los valores de la PSF generada mediante el modelo, es por esto que las medidas de la tabla están promediadas para el campo muestral de las veinte PSF disponibles.

En las siguientes imágenes se muestran lo que llegaría al telescopio en el caso de tener una deformación aleatoria según el nivel de ruido arriba establecido y el modelo estadístico de ruido atmosférico:

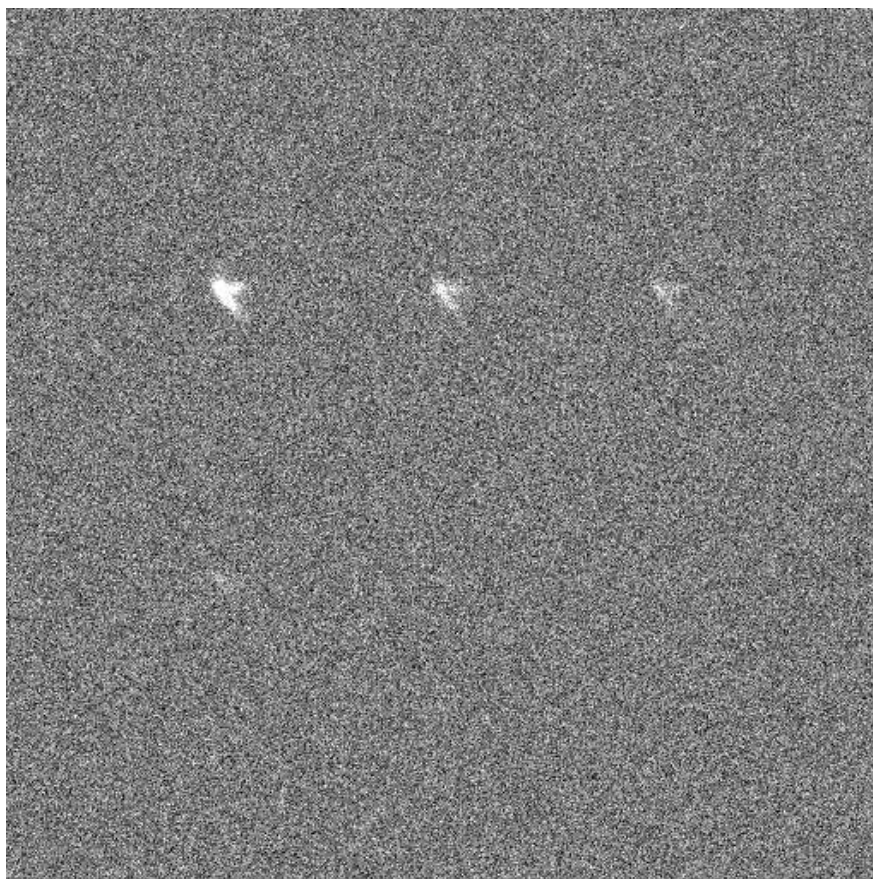


Ilustración 44. Imagen de modelo atmosférico con ruido de  $\sigma_N = 4.2e6$

### 5.2.2 Resultado algoritmo FASTCAM [1]

En este caso en el que las imágenes contienen aberraciones de mayor orden aparte de desplazamientos verticales u horizontales, cabe esperar que los resultados sean un tanto peores.



**Ilustración 45.** Solución FASTCAM 50% en modelo atmosférico

Queda patente que la imagen recuperada no dispone de definición suficiente para distinguir los cuerpos celestes, sin embargo si ha conseguido una difusa representación de las seis estrellas, pues se puede observar que todos los puntos están presentes.

Como ya se ha documentado en textos anteriores de *lucky imaging*, el ensanchado espacial puede imposibilitar la representación fiel de estrellas binarias (aquellas que están muy cerca y se puede confundir).

Esto se aprecia bien en la siguiente representación 3D:

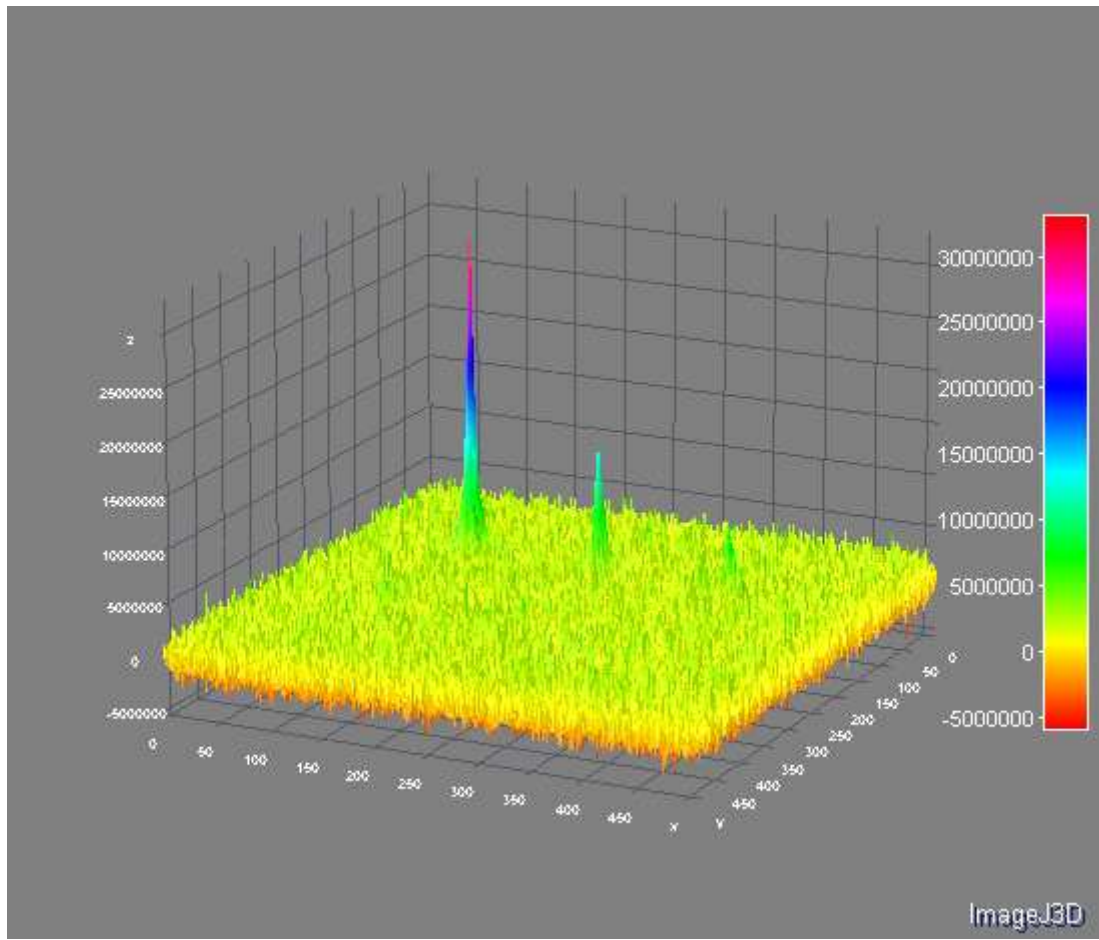


Ilustración 46. Surface plot de FASTCAM 50% en modelo atmosférico

A continuación se presenta el histograma.

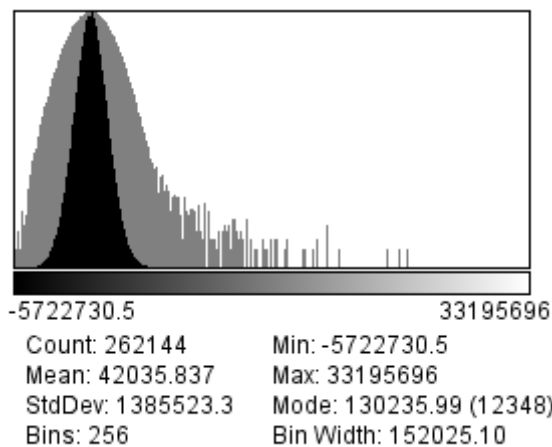
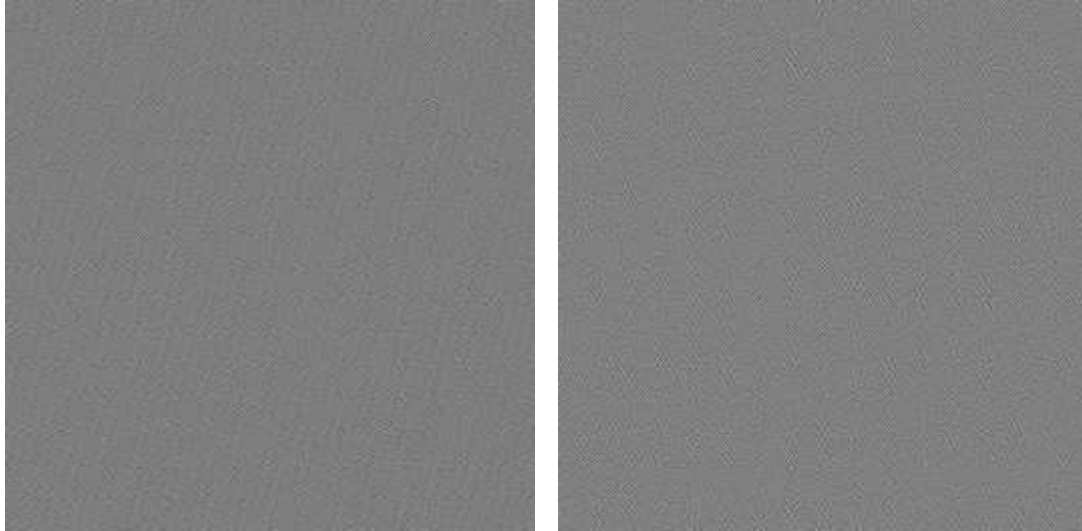


Ilustración 47. Histograma de FASTCAM 50% en modelo atmosférico

Si se compara esta imagen con la ilustración 26, en la que aparece el histograma del resultado de FASTCAM con el 100% de los fotogramas que podemos tomar al suponer solo Tip&Tilt, se puede apreciar una mayor anchura de la campana en este caso pues se han promediado menos imágenes.

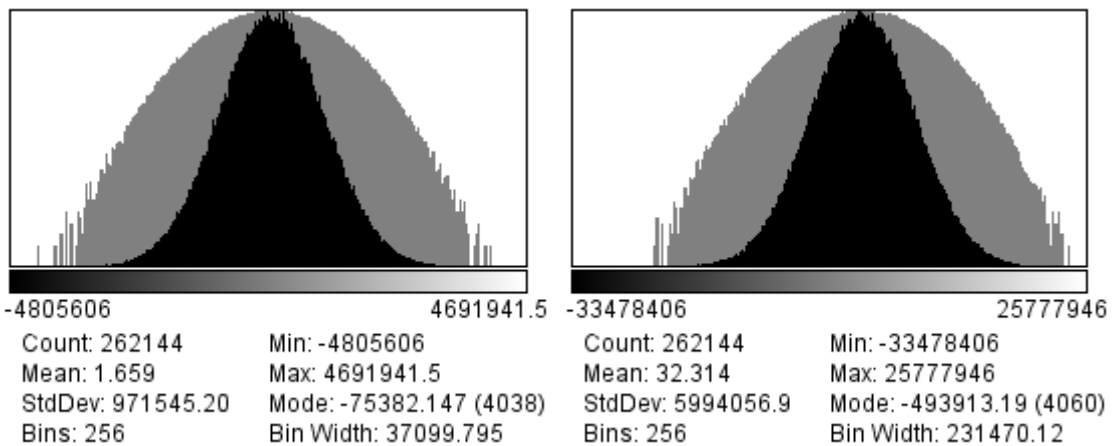
### 5.2.3 Resultado deconvolución estándar [2] y [5]

En este caso es lógico que con el nivel de ruido introducido no se pueda deconvolucionar el resultado sin caer en la saturación.



**Ilustración 48.** Caso de deconvolución estándar con modelo atmosférico. Izquierda: promediado anterior. Derecha: promediado posterior.

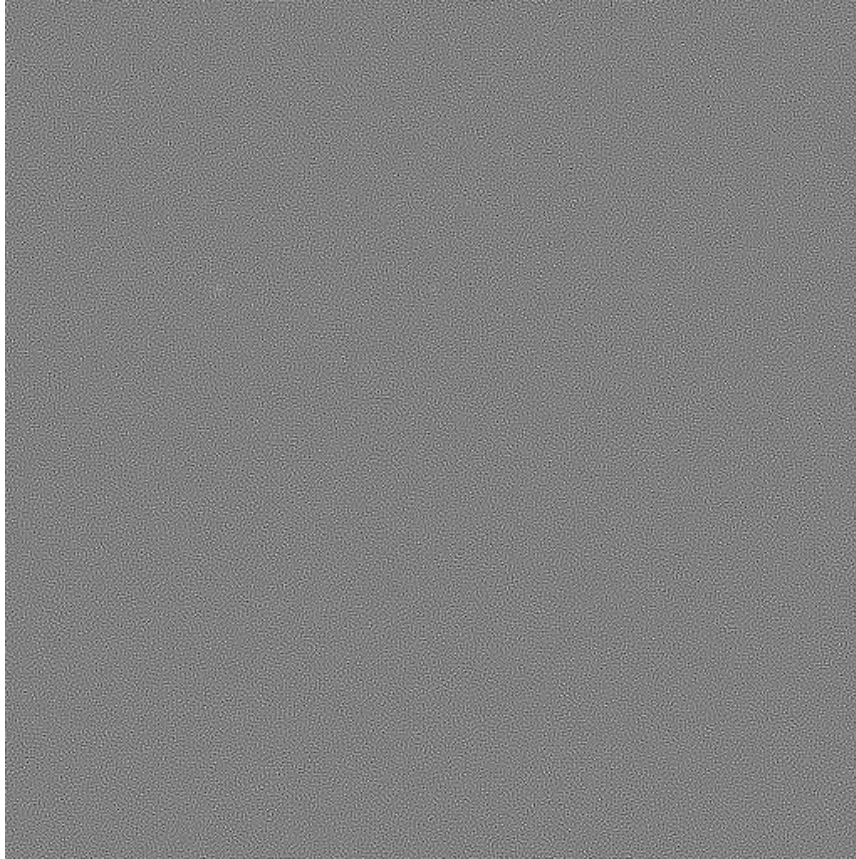
En este caso, para el nivel de ruido establecido, la convolución está saturada de ruido. En el caso de tener 1000 muestras podremos mejorar un poco este resultado, pero nunca lo suficiente como para que merezca la pena tomar 1000 muestras.



**Ilustración 49.** Comparación de histogramas de deconvolución estándar, a la izquierda prepromediada, a la derecha postpromediada

#### 5.2.4 Resultado deconvolución Primot [4] y [7]

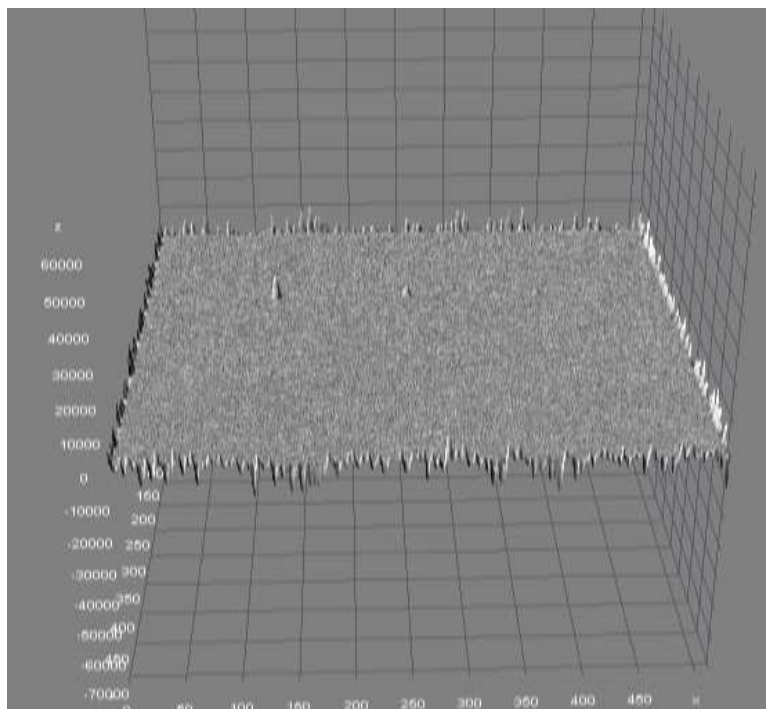
En el caso del algoritmo sin regularización propuesto por (Primot, Rousset, & Fontanella, 1990) se obtiene el siguiente resultado:



**Ilustración 50.** Caso Primot prepromediado en modelo atmosférico

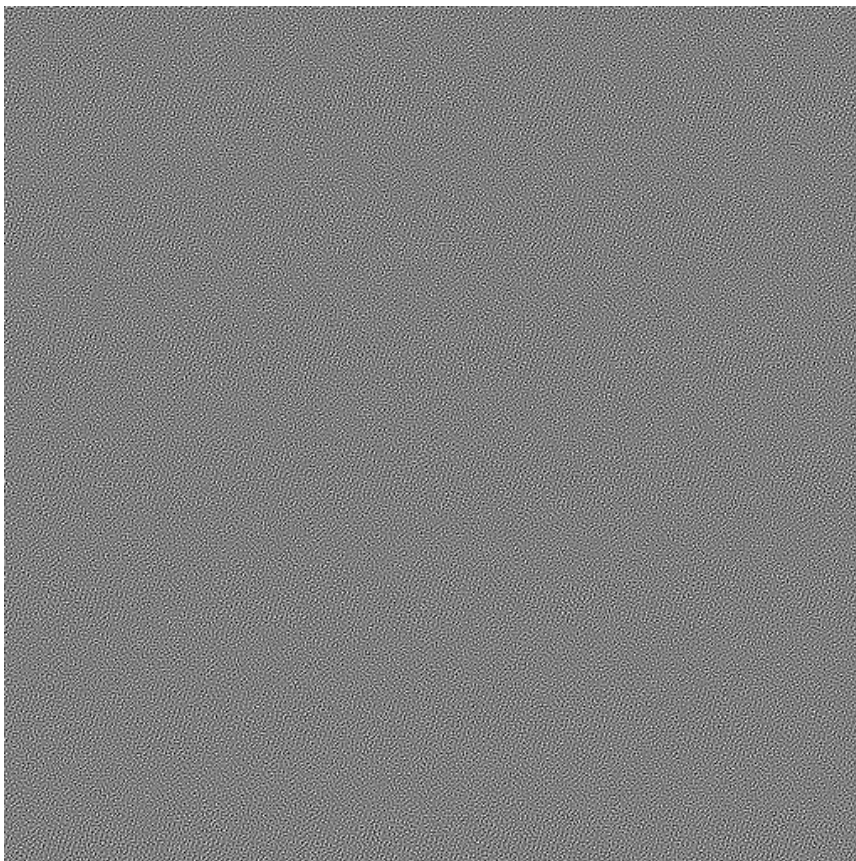
El resultado a priori no es demasiado positivo. Hay que tener en cuenta que el algoritmo se apoya en la media de muchas muestras para dar validez a la premisa de cancelación de ruido. Veinte muestras no es un número suficiente. Se espera que los resultados mejoren en el campo de 1000 muestras.

A simple vista no se puede decir que el resultado sea bueno, pues una visualización del trazado en 3D revela con más claridad, que se ven las dos primeras estrellas en intensidad y que se empiezan a intuir la siguiente:

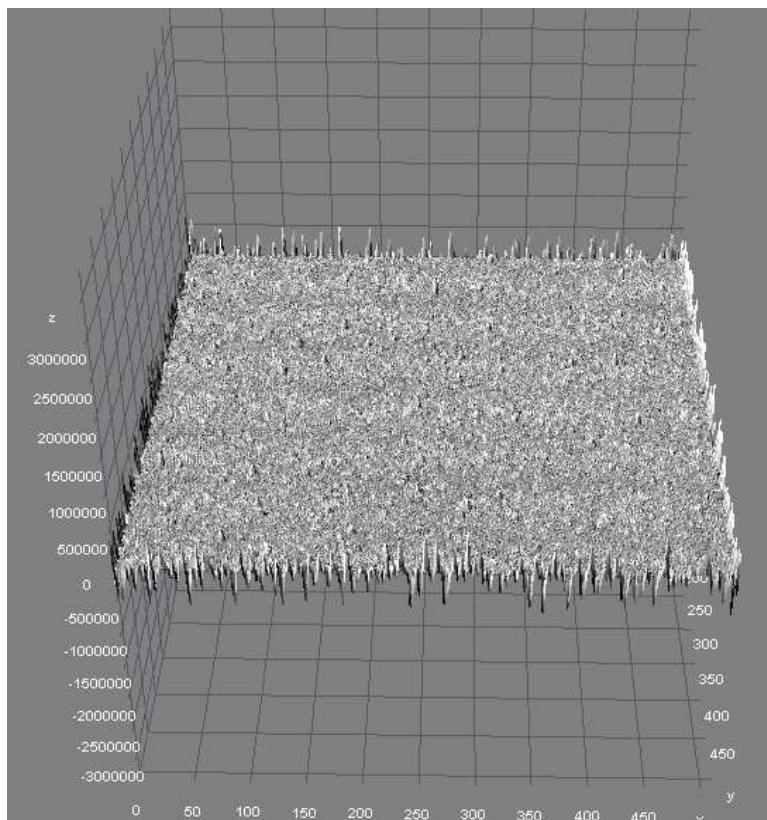


**Ilustración 51.** Surface plot de Primot con prepromediado en modelo atmosférico

Tal y como ya se ha realizado en el caso de los modelos simples de Tip&Tilt, se presenta el resultado del algoritmo con promediado posterior a la deconvolución. A modo de referencia, el resultado nos sirve para validar la teoría de (Primot, Rousset, & Fontanella, 1990) de que el promediado anterior arroja mejores resultados que el promediado posterior a la deconvolución.



**Ilustración 52.** Caso Primot Postpromediado con modelo atmosférico



**Ilustración 53.** Visualización 3D del caso Primot Postpromediado con modelo atmosférico

### 5.2.5 Resultado deconvolución de Wiener con regularización constante [3] y [6]

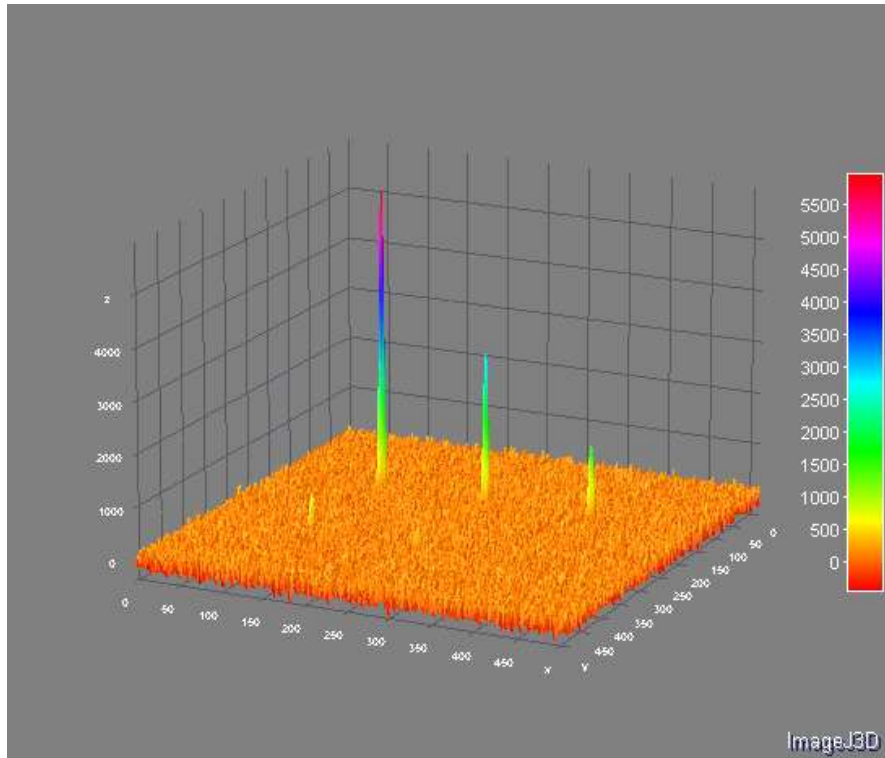
En este caso se tienen unos resultados de similar calidad a cuando se realizó el tratamiento con el modelo de Tip&Tilt, esto es algo que no ha podido mantener el resto de algoritmos sin regularización pues a pesar de que realizaban correcciones de las aberraciones del frente de onda no hacían nada al respecto del ruido gaussiano presente.



**Ilustración 54.** Caso Wiener con prepromediado en modelo atmosférico de 20 muestras. ( $\gamma = 4.2 \cdot 10^6$ )

Veamos la representación 3D:



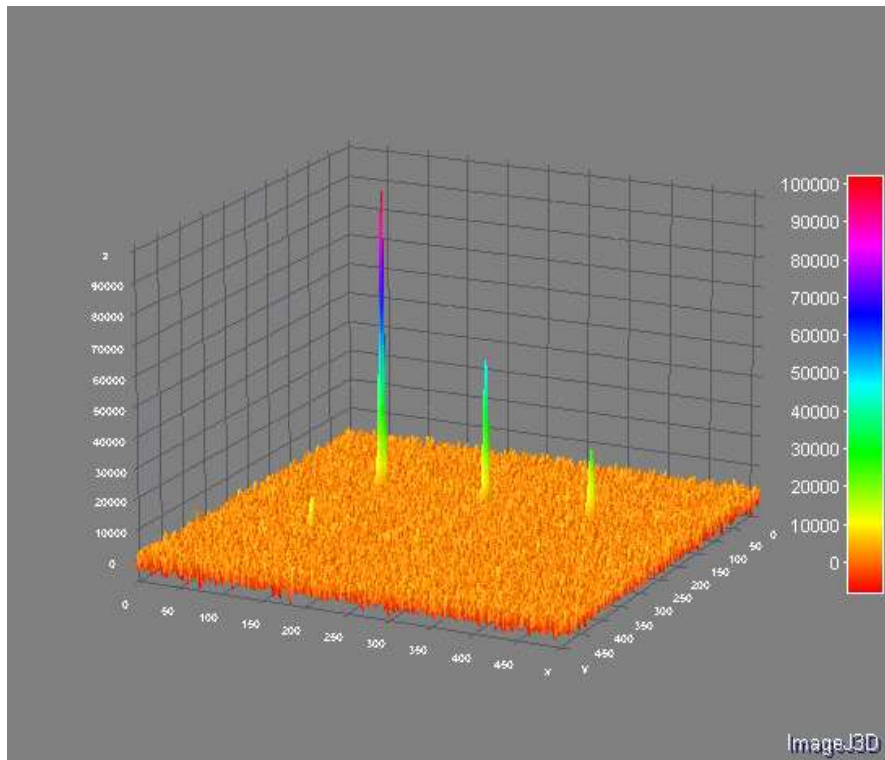


**Ilustración 55.** Trazado 3D del caso Wiener con promediado en el modelo atmosférico de 20 muestras ( $\gamma = 4.2 \cdot 10^6$ )

En este caso de regularización se puede observar visualmente que no existe gran diferencia entre el promediado y el postpromediado.

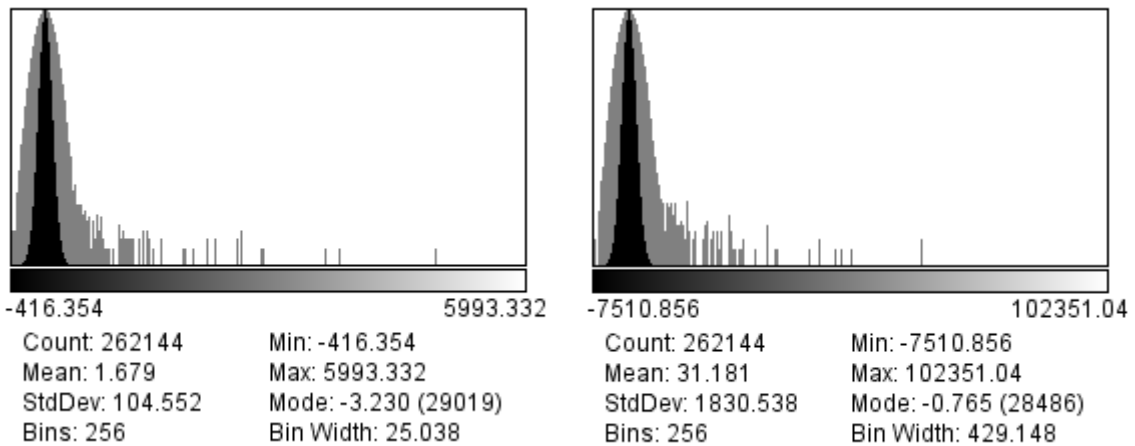


**Ilustración 56.** Caso Wiener con postpromediado en modelo atmosférico de 20 muestras. ( $\gamma = 4.2 \cdot 10^6$ )



**Ilustración 57.** Trazado 3D del caso Wiener con postpromediado en el modelo atmosférico de 20 muestras. ( $\gamma = 4.2 \cdot 10^6$ )

Como se podrá observar en los histogramas comparados a continuación, la mayor diferencia se encuentra en los rangos de las imágenes:



**Ilustración 58.** Comparación histogramas Pre y Post promediados de Wiener en el modelo atmosférico de 20 muestras. A la izquierda: Prepromediado, a la derecha Postpromediado. ( $\gamma = 4.2 \cdot 10^6$ )

En cuanto a la eliminación del ruido, se puede apreciar que esta no se ha visto afectada por la aparición de efectos distorsionadores de mayores órdenes a las traslaciones.

### 5.2.6 Resultado deconvolución de Wiener con regularización paso alto [8]



**Ilustración 59.** Resultado de regularización paso alto en el caso de modelo atmosférico de 20 muestras

En este caso se aprecia un mejor resultado que en el caso de usar este filtrado para el modelo de traslaciones. Como se verá posteriormente en el estudio estadístico, este algoritmo se encuentre muy cercano a los resultados arrojados por FASTCAM, pero hay que recordar que este método no necesitaría más que una imagen para obtener un resultado similar al obtenido con 20 imágenes.

## 5.3 Estudio estadístico de los modelos de 20 muestras

### 5.3.1 Figuras de mérito de los algoritmos

En cuanto a los métodos usados para comparar los resultados en nuestro caso hemos utilizado los siguientes parámetros estadísticos:

Nombre	Formula
<b>Pearson</b>	$p = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}}$
<b>Correlación reflectiva</b>	$r_{XY} = \frac{\sum_{i=1}^n (X_i)(Y_i)}{\sqrt{\sum_{i=1}^n (X_i)^2} \sqrt{\sum_{i=1}^n (Y_i)^2}}$
<b>Mean Square Error</b>	$MSE_{XY} = \frac{\sum_{i=1}^n (X_i - Y_i)^2}{\sum_{i=1}^n (X_i^2)}$
<b>Signal to Noise Relation</b>	$SNR(dB) = 10 \log_{10} \frac{1}{MSE}$
<b>Sensitivity</b>	$d = \frac{\max_i X_i}{\sigma_i}$

Tabla 5. Figuras de mérito del estudio comparativo

A continuación se presentan los datos obtenidos de las pruebas recolectados por una macro de ImageJ:

Metodo	Pearson	Ref. Corr	MSE	SNR(dB)	Sensitivity
<b>FASTCAM</b>	0,15000000	0,15100000	384409823,7	25,779	25,274
<b>Standrd PRE</b>	0,00007095	0,00007097	188842581,2	0,166	4,726
<b>Standrd POS</b>	0,00024950	0,00024960	7188125232,2	-0,006	4,329
<b>Primot PRE</b>	0,00100000	0,00100000	937039,0	0,093	4,339
<b>Primot POS</b>	0,00024020	0,00024020	19389116,4	0,098	4,419
<b>Wiener PRE</b>	0,48500000	0,48500000	1,8	46,393	66,697
<b>Wiener POS</b>	0,48200000	0,48200000	646,6	49,940	65,582
<b>Wiener PRE HP</b>	0,13000000	0,13000000	55,3	16,409	18,651

Tabla 6. Parámetros estadísticos de los algoritmos en modelo de Noll para 20 muestras

En cuanto a ambas correlaciones se esperaba que dieran resultados distintos en el caso de FASTCAM, pues la correlación reflectiva se supone sensible a desplazamientos de la imagen, sin embargo, la diferencia no ha sido muy importante.

Metodo	Pearson	Ref. Corr	MSE	SNR(dB)	Sensitivity
<b>FASTCAM</b>	0,32494010	0,32526585	325647098,5	28,31	36,48
<b>Standrd PRE</b>	0,00006699	0,00006705	1413770525,2	-0,53	4,32
<b>Standrd POS</b>	0,00596041	0,00596176	11485354,5	-1,48	3,89
<b>Primot PRE</b>	0,02461672	0,02462250	1638,1	0,59	4,73
<b>Primot POS</b>	0,00596041	0,00596176	28712,5	0,21	3,96
<b>Wiener PRE</b>	0,30802902	0,30816159	4,2	42,40	34,46
<b>Wiener POS</b>	0,31051399	0,31064740	1739,7	39,47	34,84
<b>Wiener PRE HP</b>	0,03841381	0,03840397	557,1	9,44	6,85

Tabla 7. Parámetros estadísticos de los algoritmos en modelo de TipTilp para 20 muestras

### 5.3.2 Gráficos del estudio

#### 5.3.2.1 Correlación

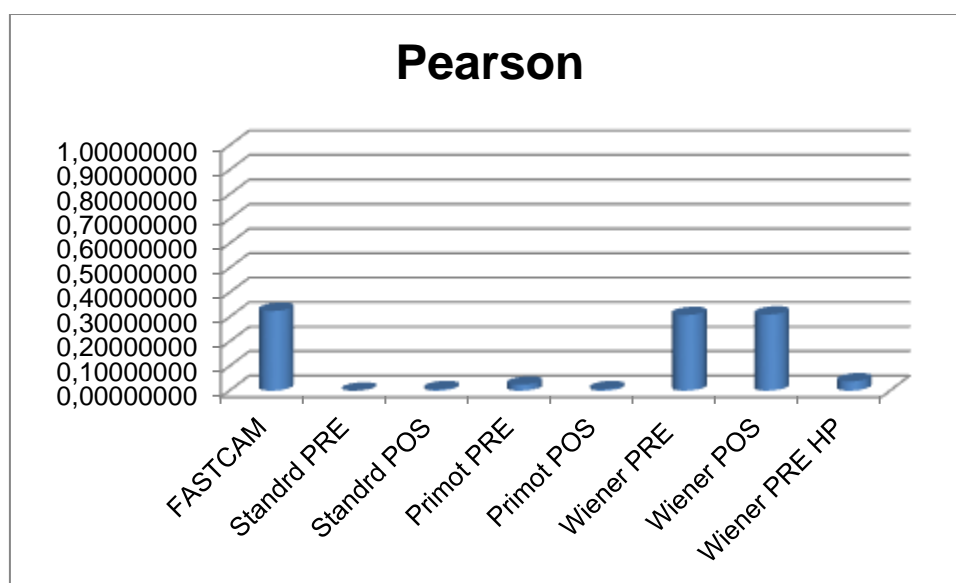


Ilustración 60. Gráfica correlación en modelo TipTilt de 20 muestras

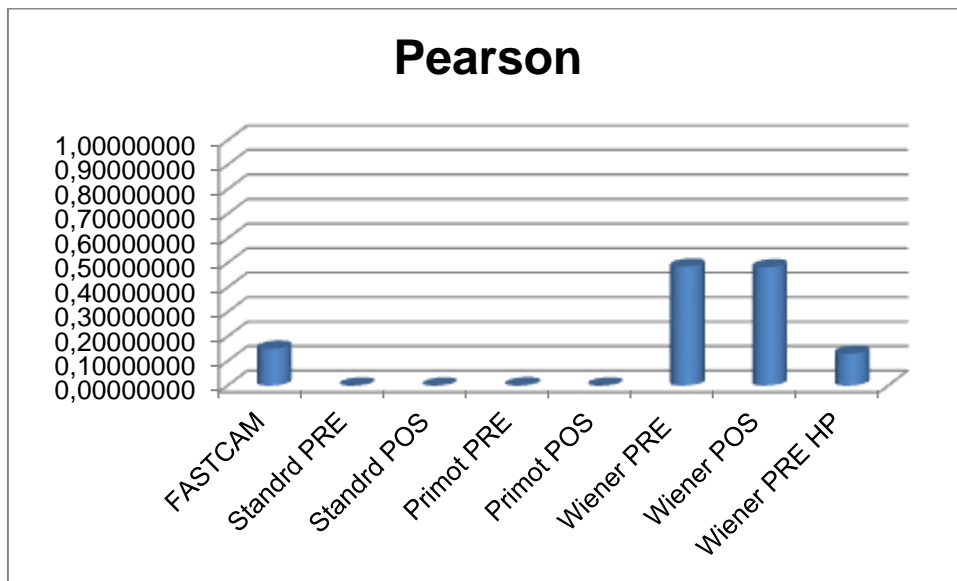


Ilustración 61. Gráfica correlación en modelo atmosférico de 20 muestras

Gráfica cruzada respecto de la correlación:

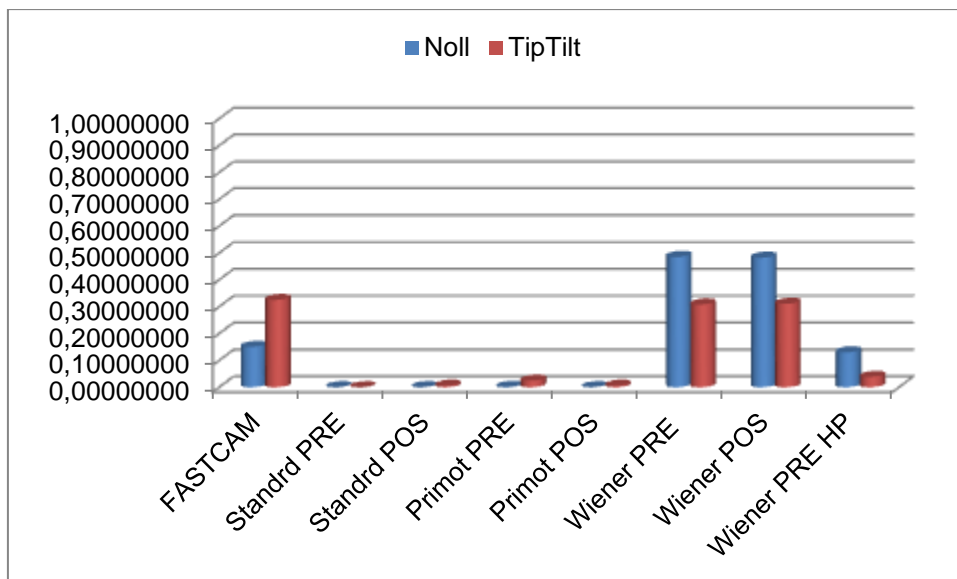


Ilustración 62. Comparativa correlación de modelo atmosférico versus Tip&Tilt

### 5.3.2.2 Mean Squared Error

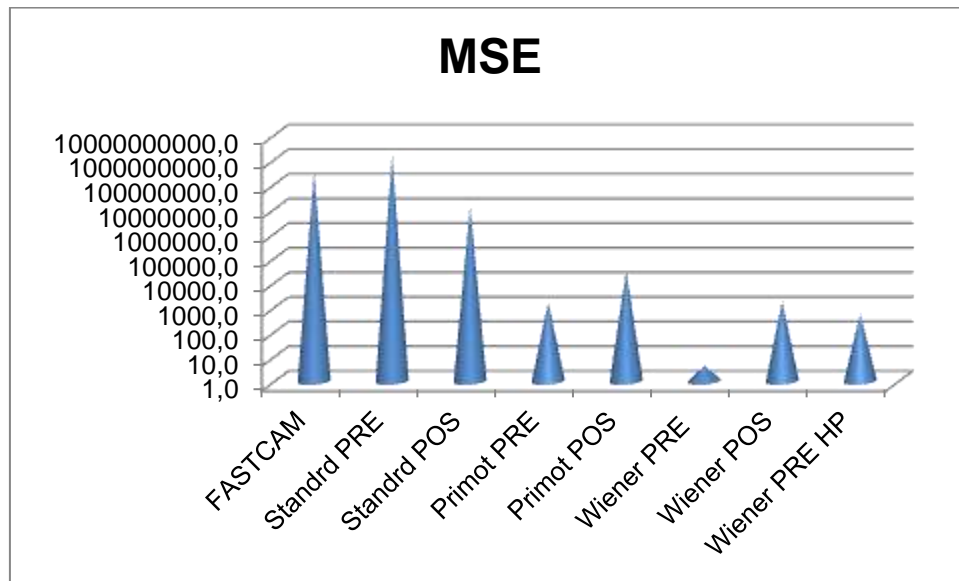


Ilustración 63. Gráfica MSE en modelo Tip&Tilt de 20 muestras

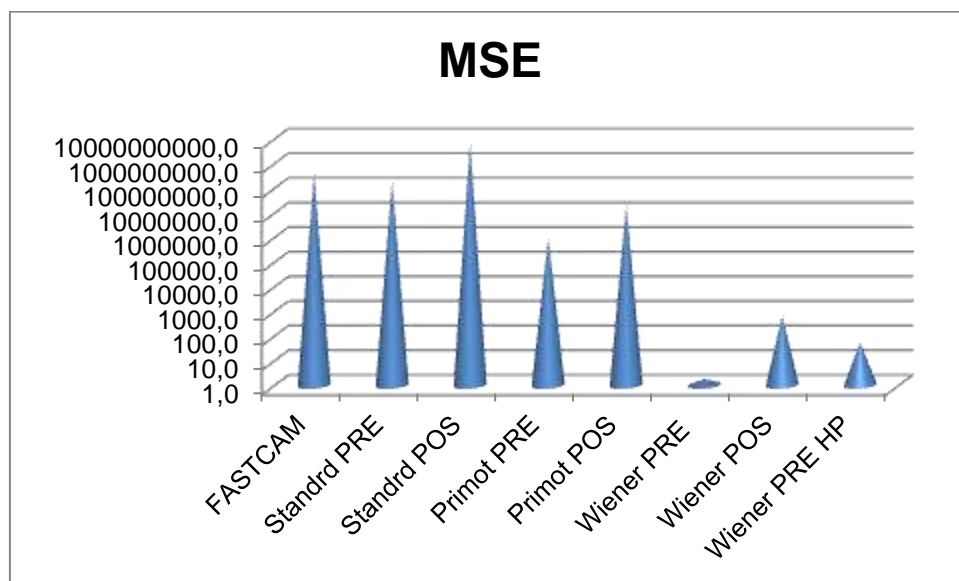


Ilustración 64. Gráfica MSE en modelo atmosférico de 20 muestras

### 5.3.2.3 *Signal to Noise Relation*

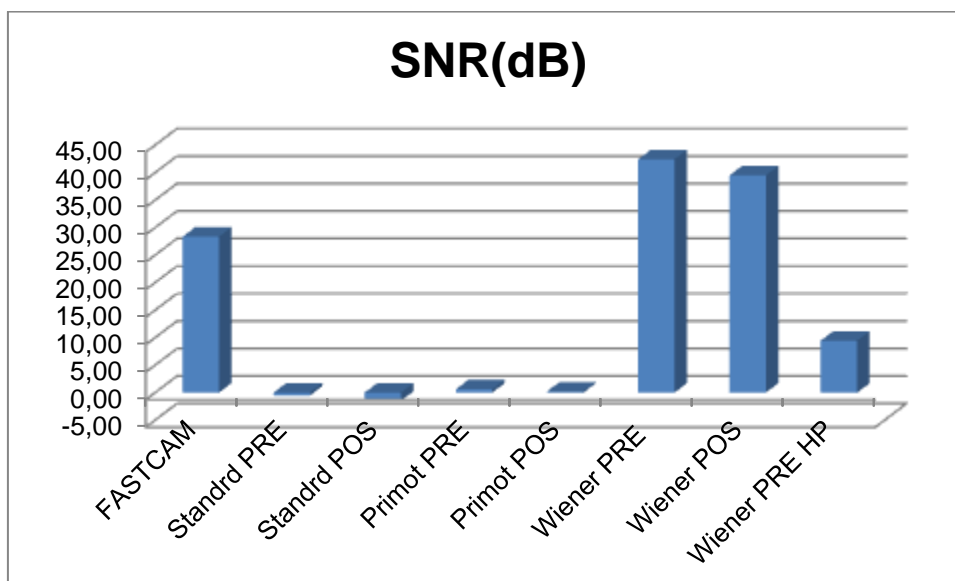


Ilustración 65. Gráfica SNR en modelo Tip&Tilt de 20 muestras

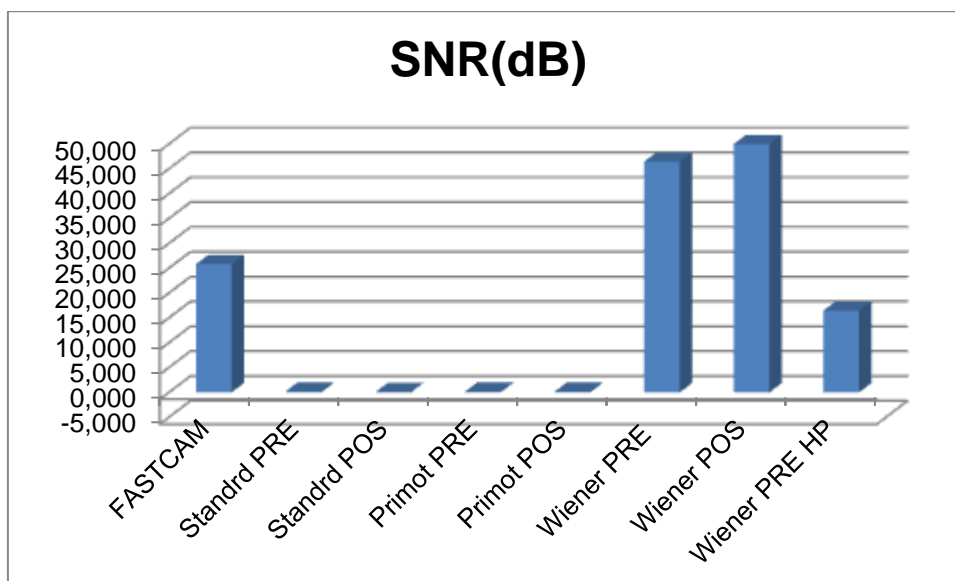


Ilustración 66. Gráfica SNR en modelo atmosférico de 20 muestras



### 5.3.2.4 Sensitividad

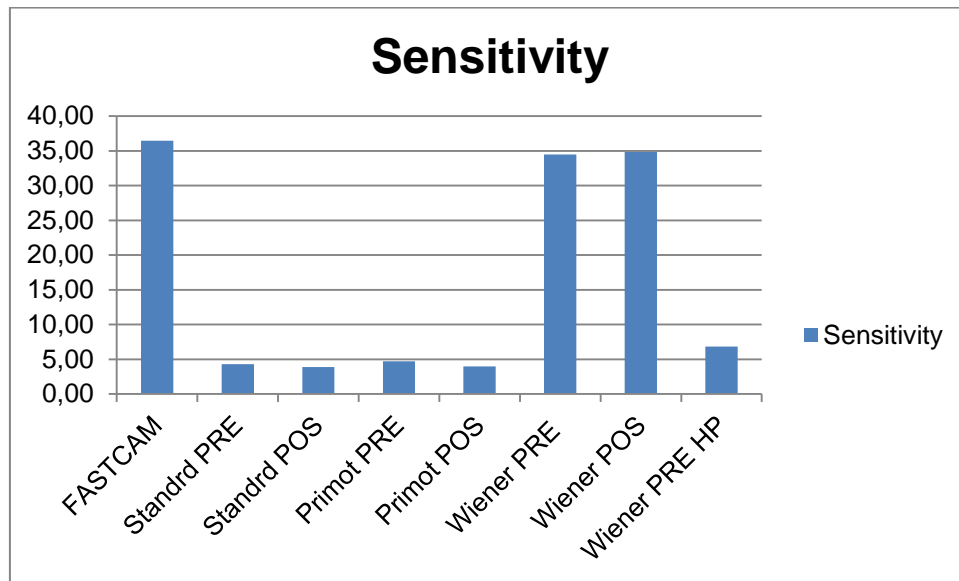


Ilustración 67. Gráfica sensibilidad en modelo Tip&Tilt de 20 muestras

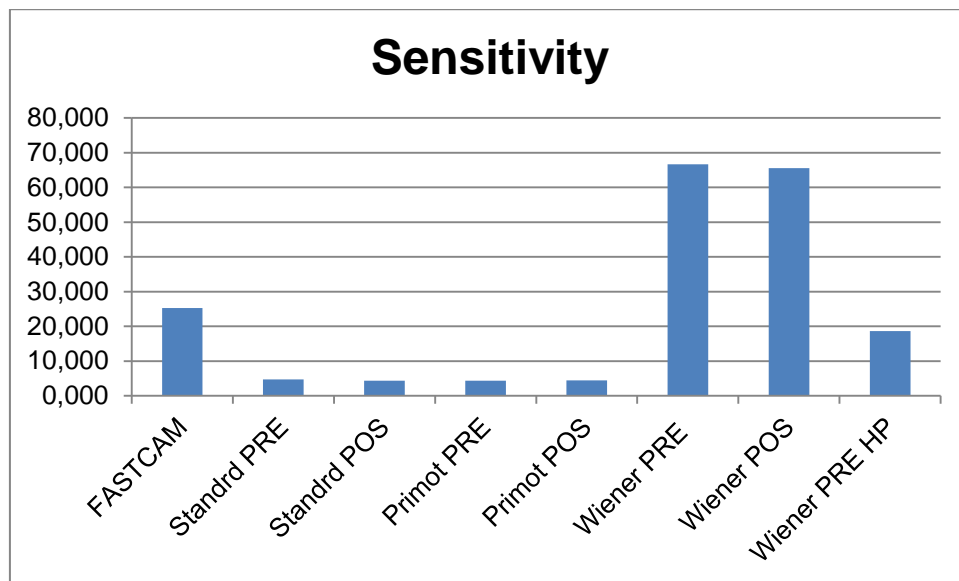


Ilustración 68. Gráfica sensibilidad en modelo atmosférico de 20 muestras

### 5.3.3 Estudio de coste computacional

El hardware en el que se han llevado a cabo las pruebas de tiempos es un doble Intel Xeon X5560 @280GHz 12GB RAM.

Algoritmo	Tiempo (ms)
std pre	110
std pos	1344
Primot Pre	2219
Primot Pos	7875
Wiener Pre	2188
Wiener Pos	2532
Wiener HP	14828

Tabla 8. Tabla de comparación de tiempos de computación

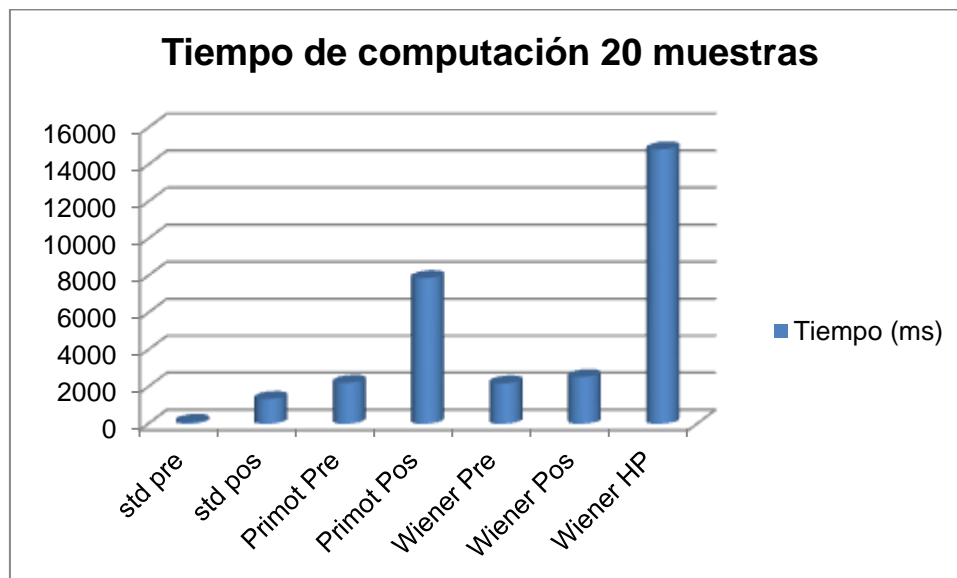


Ilustración 69. Gráfico de comparación de tiempos de computación en 20 muestras

## 6. ESTUDIO DE LARGA ESCALA

### 6.1 Comparativa visual de modelo atmosférico de 1000 muestras

#### 6.1.1 Niveles de ruido

Análogamente al apartado anterior se define un nivel de ruido en referencia al nivel medio de señal:

Original	Convolucionada
10000	34544604
5000	17272972
2500	8636489
<b>1250</b>	<b>4018412</b>
625	2.159.417
312,5	1.078.010

Tabla 9. Valores de intensidad de la muestra típica en modelo atmosférico de 1000 muestras

En las siguientes imágenes se muestran lo que llegaría al telescopio en el caso de tener una deformación de modelo atmosférico de Noll y un ruido gaussiano de **4000000 de desviación típica**.

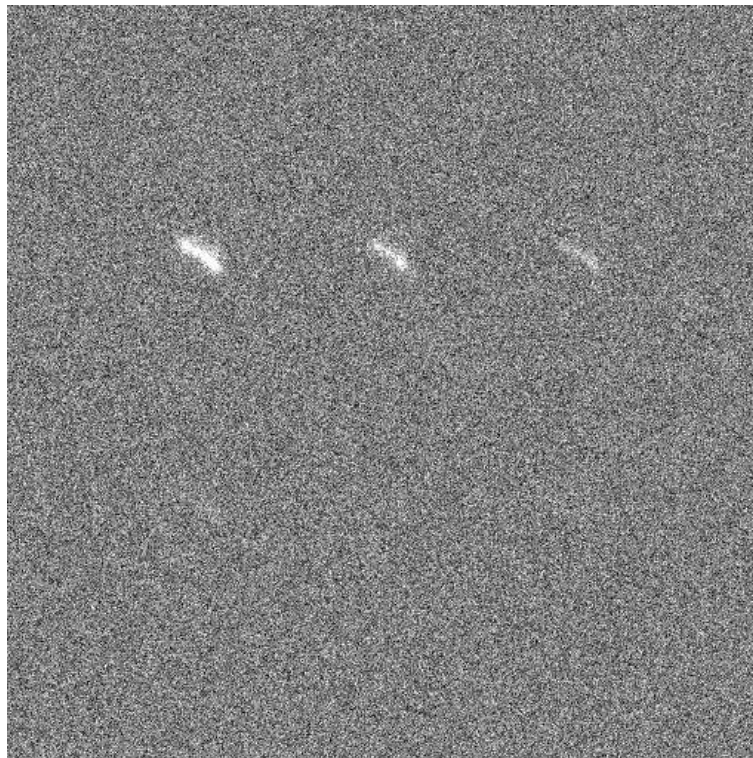


Ilustración 70. Imagen de modelo atmosférico con ruido de  $\sigma_N = 4e6$

### 6.1.2 Resultado algoritmo FASTCAM [1]

En este caso veremos como el algoritmo tendrá que elegir únicamente un pequeño porcentaje (10%) de las imágenes, considerándolas “afortunadas” frente al resto y usándolas con un recentrado para acumularlas, ya no podremos como en el caso de Tip&Tilt elegir un 100% de las imágenes, esto hará que no se pueda reducir tanto el ruido.

También hay que considerar que partimos de un campo de 1000 muestras y esto implica que se pasa de 20 muestras (100% del caso de baja escala) a 100 (10% del caso más extenso), suponiendo aun así un aumento de muestras.

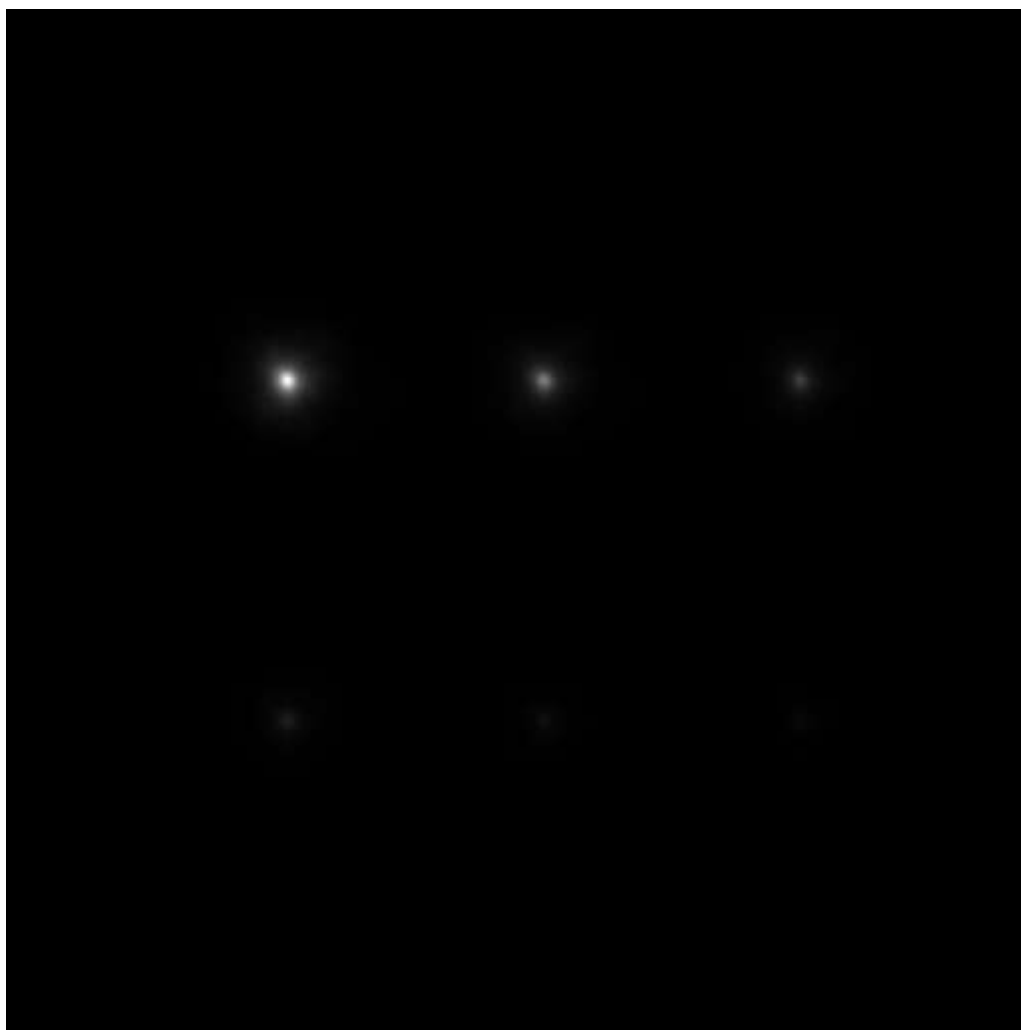


Ilustración 71. Resultado de FASTCAM 10% de larga escala en modelo atmosférico

Comparado con otros resultados, se puede apreciar que se ha obtenido una imagen bastante decente en la que se pueden intuir las seis estrellas y su escala original de intensidades. Sin embargo esto no serviría en un caso en el que se tengan muchas estrellas con una separación inferior a la actual, pues se confundirían unas con otras debido al difuminado que crea la atmosfera y que no se ha corregido mediante una deconvolución.

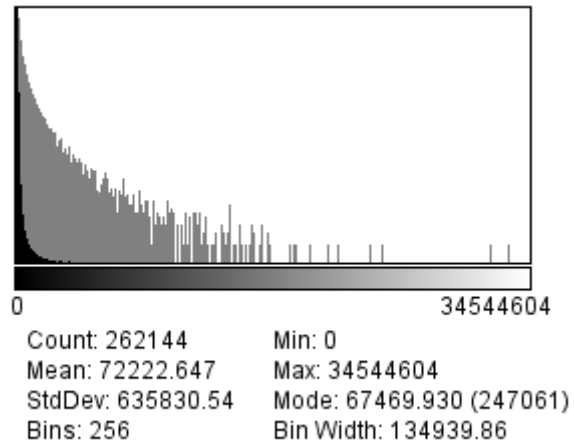


Ilustración 72. Histograma de FASTCAM 10% de larga escala en modelo atmosférico

Se puede apreciar que se ha reducido el ruido con respecto a la señal, esto da claridad a la imagen.

El efecto de la pérdida de definición se puede advertir fácilmente en la imagen 3D:

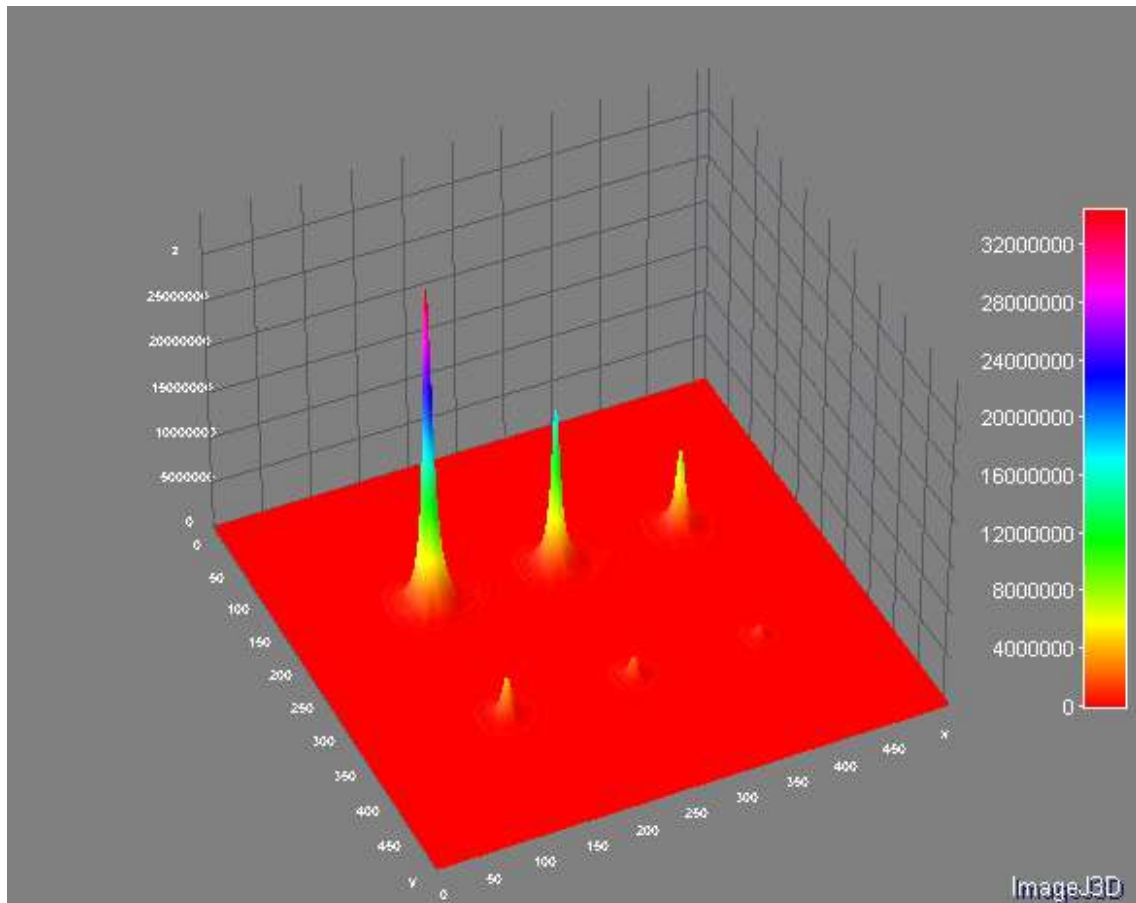
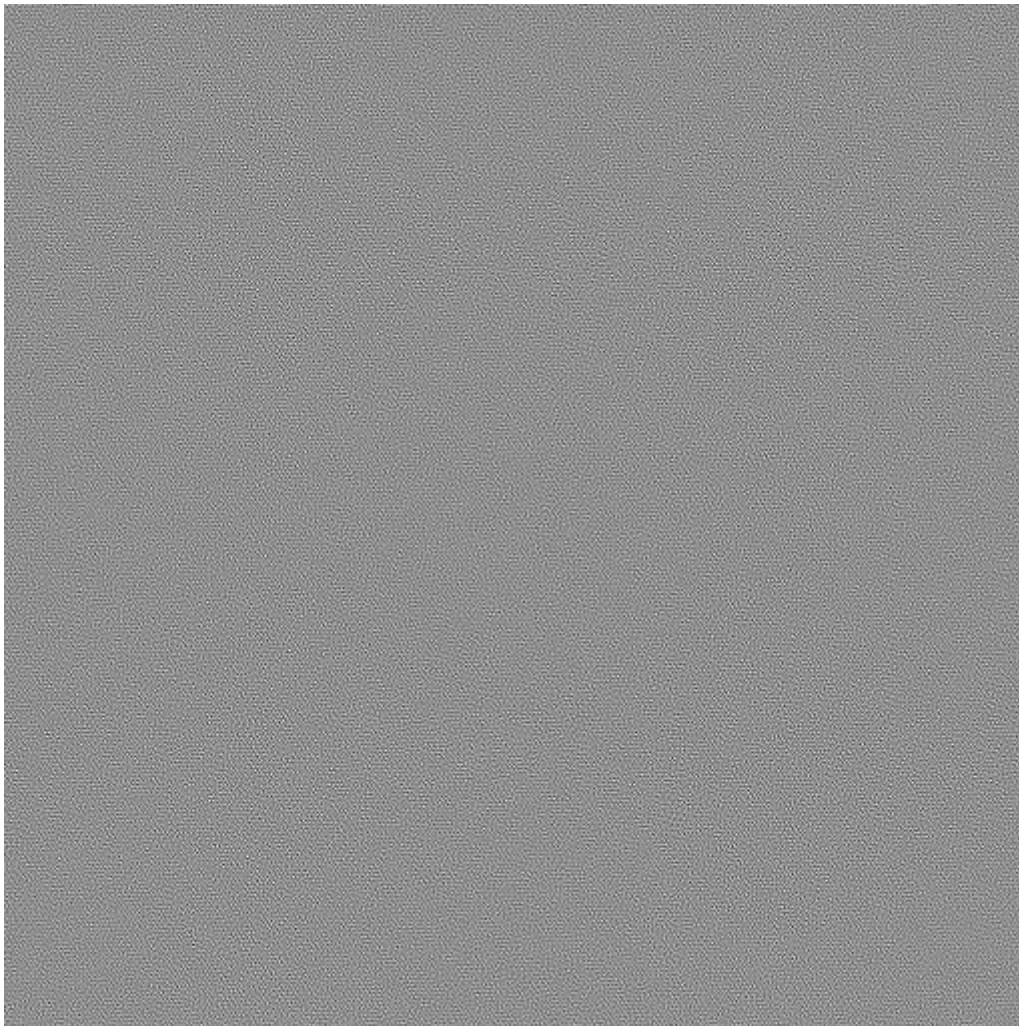


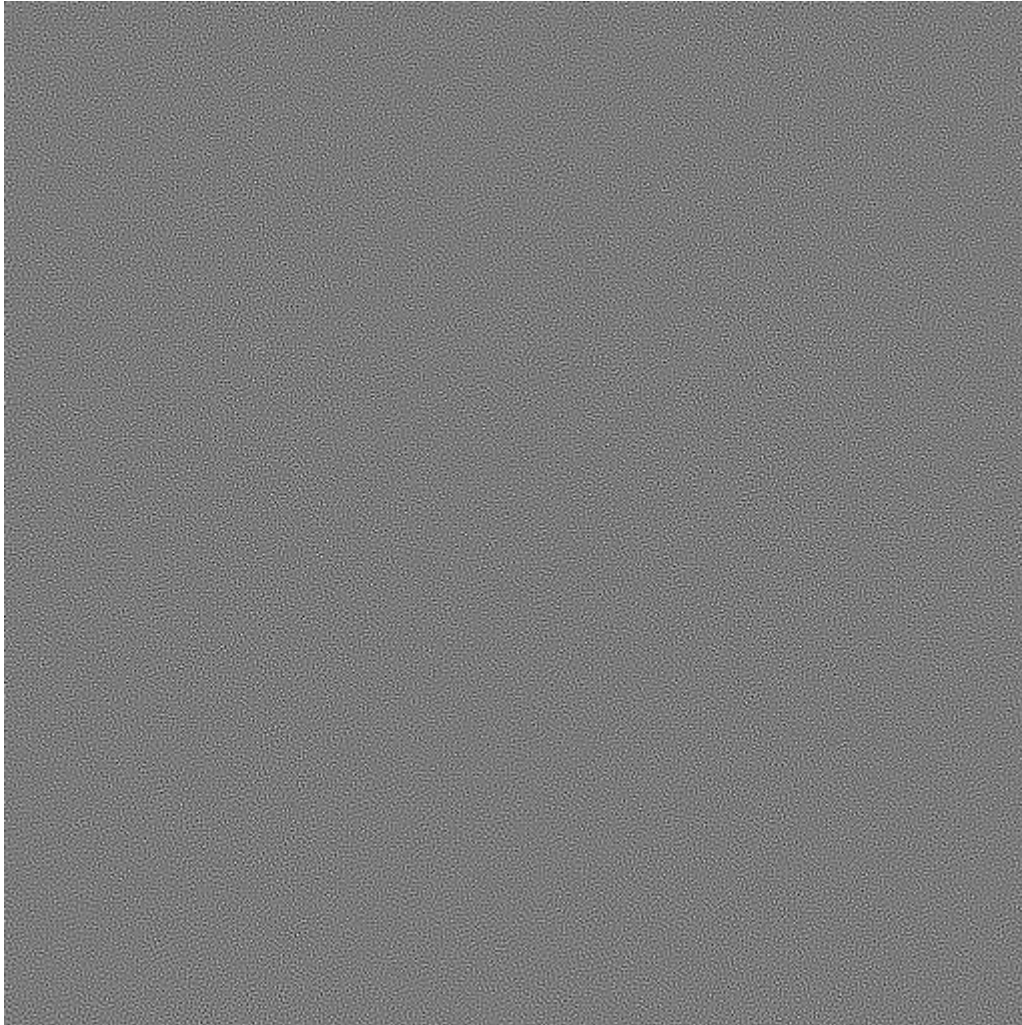
Ilustración 73. Representación 3D de FASTCAM 10% de larga escala en modelo atmosférico

### 6.1.3 Resultado deconvolución estándar [2] y [5]

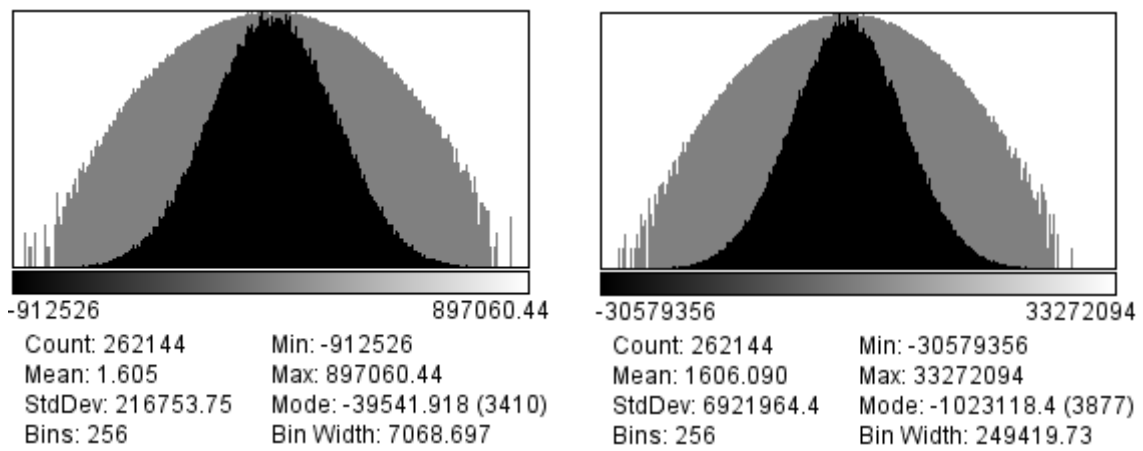
Se observa a continuación, como el resultado ha sido infructuoso, hasta el punto de que no se puede distinguir nada sobre el ruido.



**Ilustración 74.** Resultado de deconvolución estándar prepromediada en modelo atmosférico de larga escala



**Ilustración 75.** Resultado de deconvolución estándar postpromediada en modelo atmosférico de larga escala



**Ilustración 76.** Comparación de histogramas de deconvolución estándar en modelo atmosférico de larga escala, a la izquierda caso prepromediado, a la derecha caso postpromediado

#### 6.1.4 Resultado deconvolución Primot [4] y [7]

En este caso tenemos que tener en cuenta que la gran cantidad de imágenes va a dar lugar a una saturación de la imagen, superando en mucho la cantidad máxima que puede almacenar la imagen TIFF de 32 bits.

Para bordear este problema, se realiza una media aritmética en lugar de realizar una simple acumulación de *slíces*.

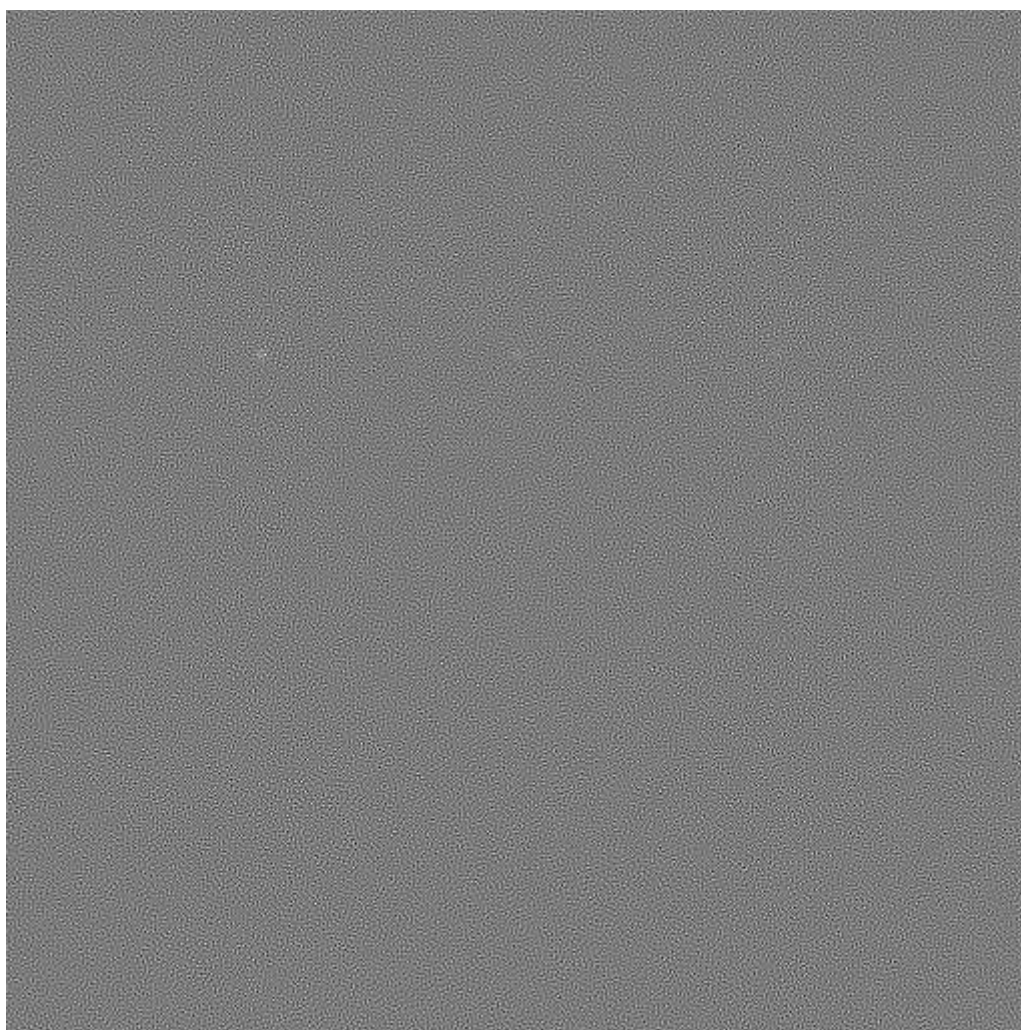


Ilustración 77. Caso Primot prepromediado en modelo atmosférico de 1000 muestras

Se puede colegir que no se ha conseguido mucho al aumentar el número de muestras. Esto nos sirve para descartar el método de Primot en casos en los que el ruido se encuentra en ordenes de magnitud que no sean excesivamente menores que los de señal.

Considerando esto y que debido a que por norma general los postpromediados dan lugar a imágenes con mayores intensidades, el caso de Primot postpromediado no se incluye en las pruebas de 1000 muestras.



### 6.1.5 Resultado deconvolución de Wiener con regularización constante [3] y [6]



Ilustración 78. Caso Wiener con premediado en modelo atmosférico de 1000 muestras. ( $\gamma = 4 \cdot 10^6$ )

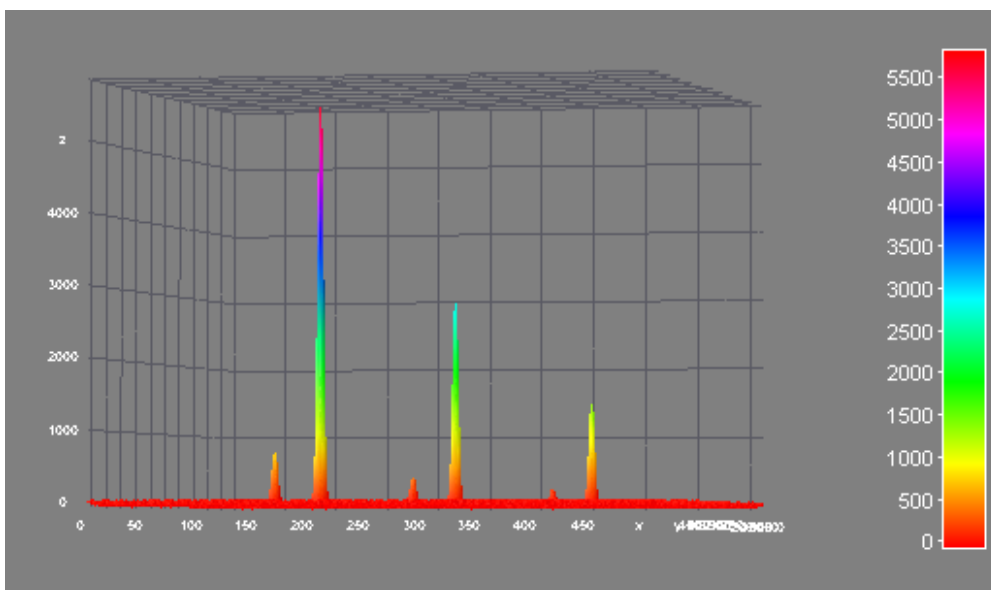
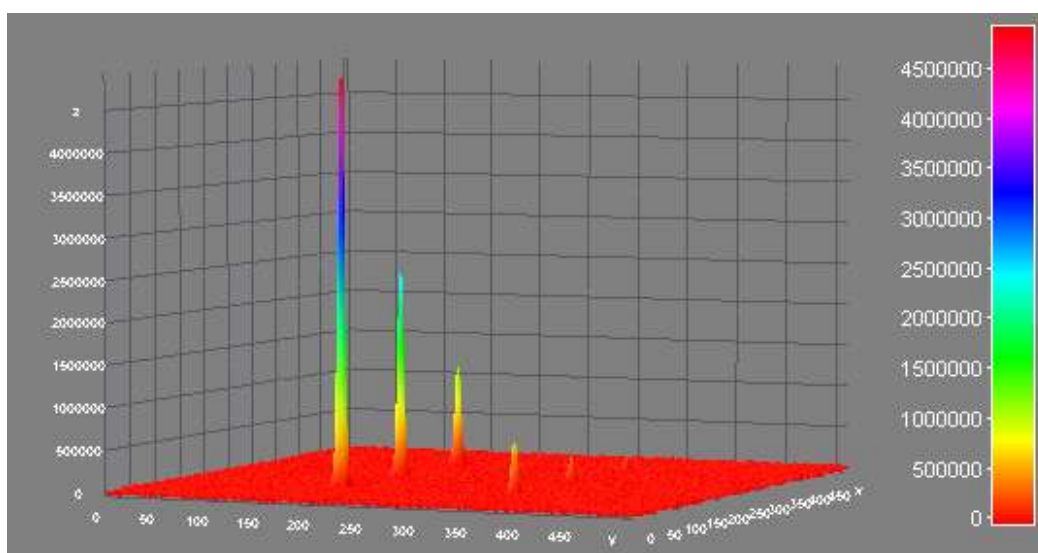


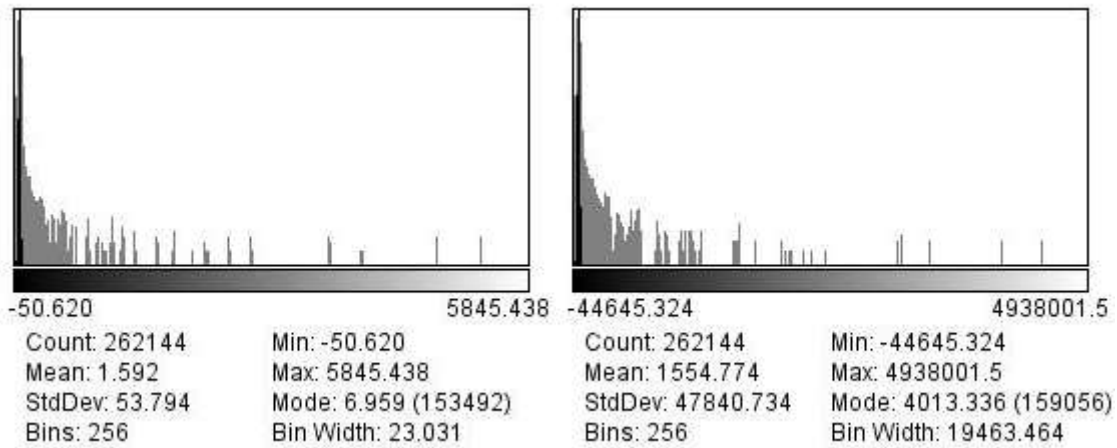
Ilustración 79. Representación 3D de caso Wiener con premediado en modelo atmosférico de 1000 muestras



**Ilustración 80.** Caso Wiener con postpromediado en modelo atmosférico de 1000 muestras. ( $\gamma = 4 \cdot 10^6$ )



**Ilustración 81.** Representación 3D de caso Wiener con postpromediado en modelo atmosférico de 1000 muestras



**Ilustración 82.** Comparación histogramas Pre y Post promediados de Wiener en el modelo atmosférico de 1000 muestras. A la izquierda: Prepromediado, a la derecha Postpromediado. ( $\gamma = 4 \cdot 10^6$ )

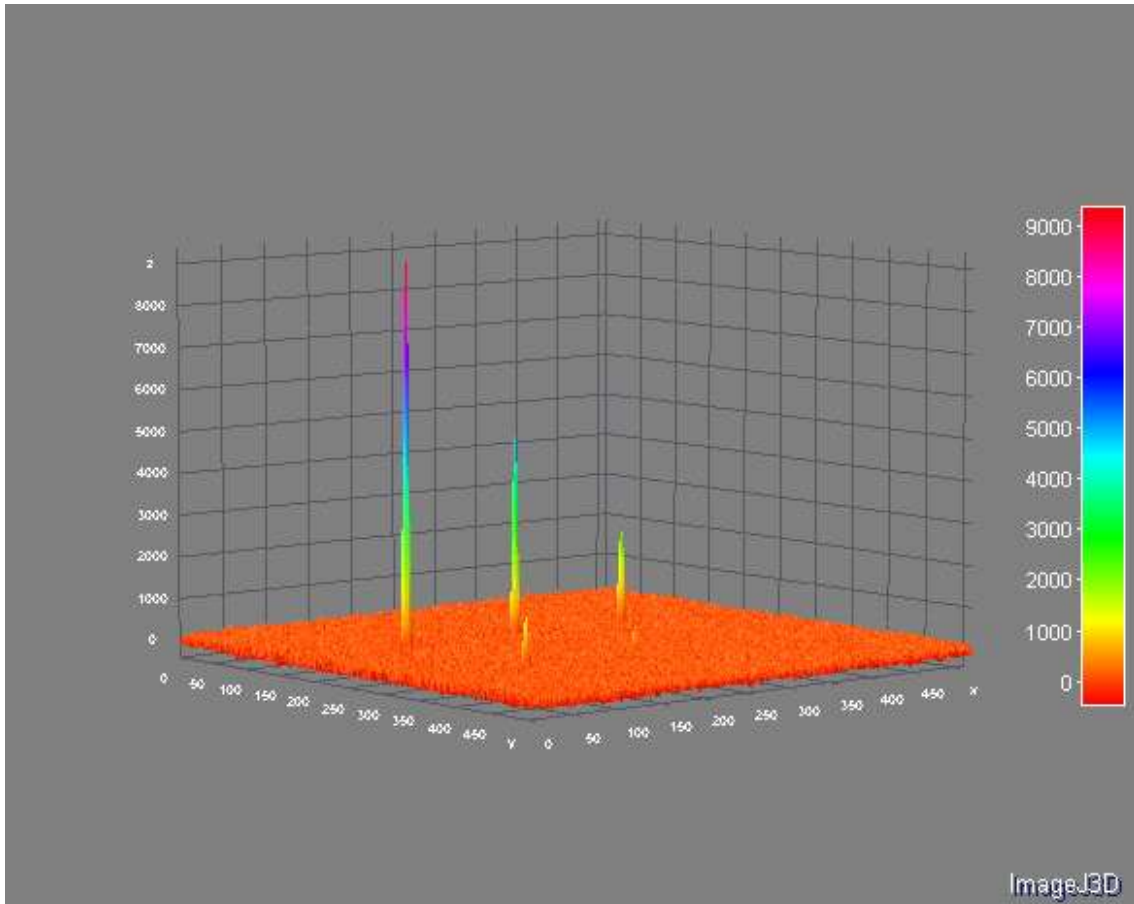
En este caso en el que disponemos de 1000 muestras se ha notado una mejora respecto al caso de 20 muestras, pero esta mejora no es suficiente para justificar el aumento en el número de muestras. Precisamente una de las fortalezas de este algoritmo va a ser evitar el requisito de disponer de un gran número de muestras para llegar a un buen resultado en términos de SNR y de definición, según acaezcan respectivamente problemas de ruido o de seeing.

### 6.1.6 Resultado deconvolución de Wiener con regularización paso alto [8]

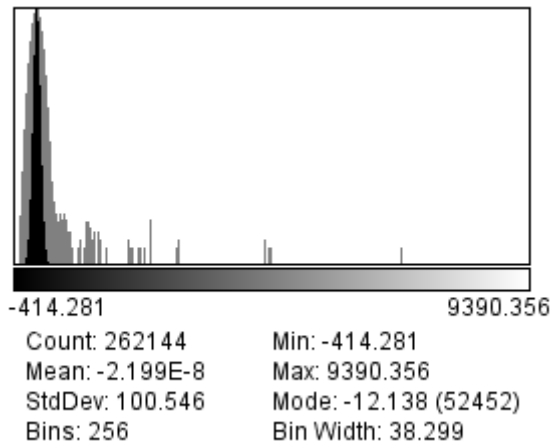


**Ilustración 83.** Resultado de regularización paso alto en modelo atmosférico de 1000 muestras.

Como es lógico siempre que se obtiene un resultado en el caso de 1000 muestras es superior al caso de menor promediado. Pero en este caso la mejora no es tan significativa recalcando un poco la menor dependencia del número de muestras con la calidad de este algoritmo.



**Ilustración 84.** Representación 3D de la regularización paso alto en modelo de 1000 imágenes



**Ilustración 85.** Histograma de la regularización paso alto en modelo de 1000 imágenes

## 6.2 Estudio estadístico de modelo atmosférico de 1000 muestras

### 6.2.1 Figuras de mérito del estudio

Metodo	Pearson	Ref. Corr	MSE	SNR(dB)	Sensitivity
<b>FASTCAM</b>	0,609000	0,608000	81915460,3	168,62	252,87
<b>Standrd PRE</b>	0,037820	0,037830	419394361,1	-0,06	4,10
<b>Standrd POS</b>	0,002000	0,002000	316829658602,9	-0,11	4,73
<b>Primot PRE</b>	0,008000	0,008000	16843,5	-0,10	5,08
<b>Primot POS</b>	0,007406	0,007408	18034166050,9	0,00	1,10
<b>Wiener PRE</b>	0,933000	0,933000	0,2	91,71	488,62
<b>Wiener POS</b>	0,913000	0,913000	457147,0	91,58	477,86
<b>Wiener PRE HP</b>	0,670000	0,670000	1116,0	68,40	124,93

Tabla 10. Parámetros estadísticos de los algoritmos en el modelo atmosférico para 1000 muestras

### 6.2.2 Gráficos del estudio

#### 6.2.2.1 Correlación

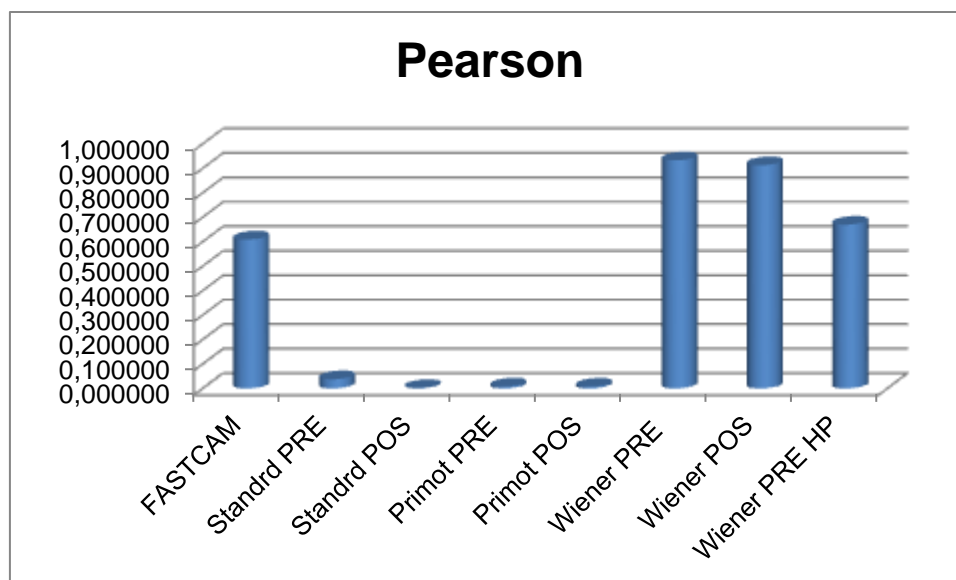


Ilustración 86. Gráfico de comparación de correlación en modelo atmosférico de 1000 muestras



Ilustración 87. Gráfico de comparativa de correlación según muestras

### 6.2.2.2 Mean Squared Error

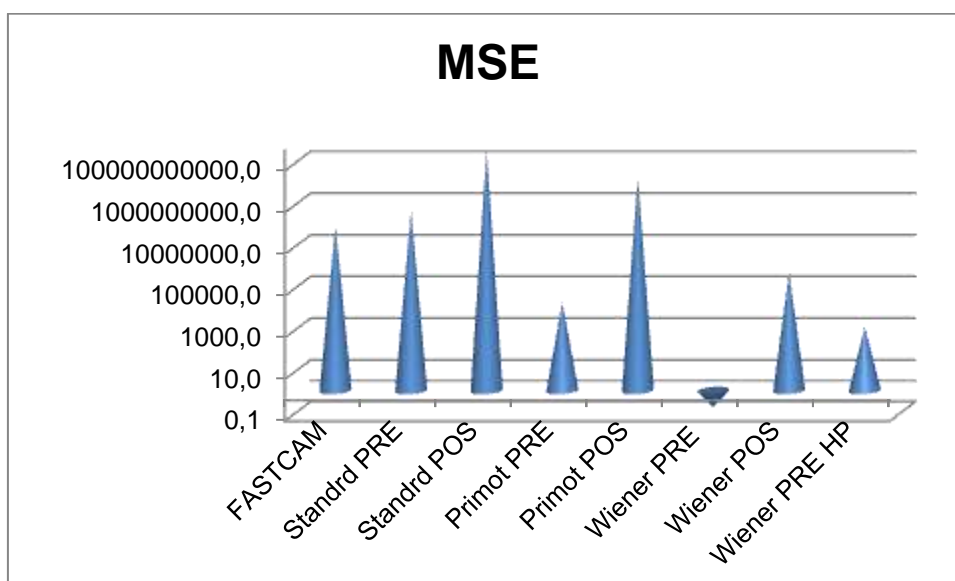


Ilustración 88. Gráfico de comparación del error cuadrático de los métodos en escala logarítmica

### 6.2.2.3 *Signal to Noise Relation*

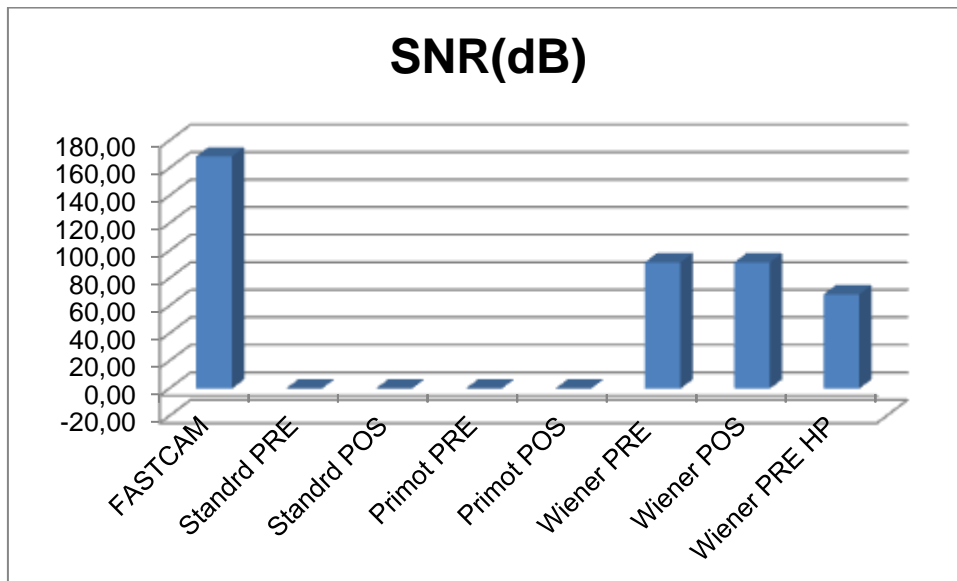


Ilustración 89. Gráfico comparativo de la SNR para el modelo atmosférico de 1000 muestras

### 6.2.2.4 *Sensitividad*

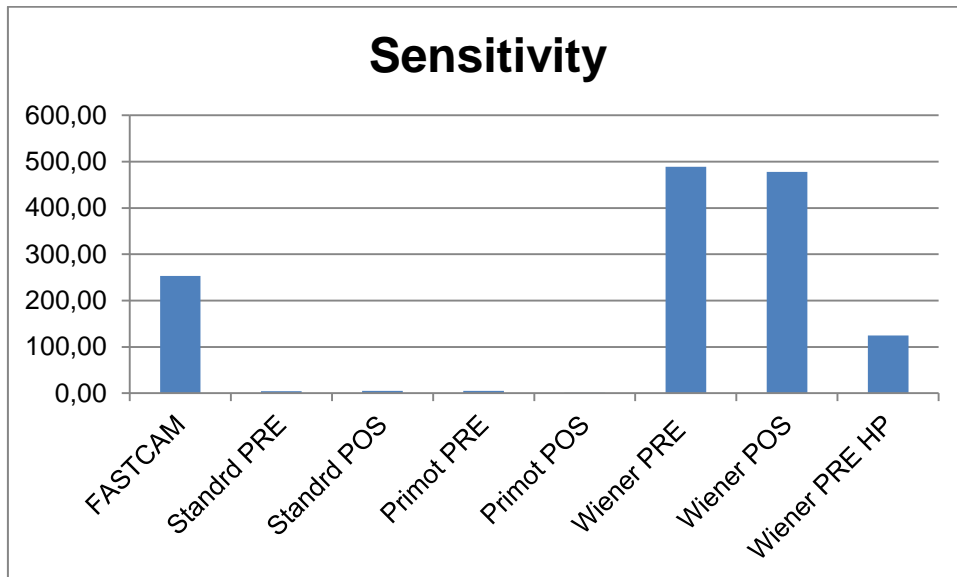


Ilustración 90. Gráfica comparativa de la sensibilidad para el modelo atmosférico de 1000 muestras



### 6.2.3 Estudio de coste computacional

Algoritmo	Tiempo (ms)
std pre	672
std pos	52093
Primot Pre	93110
Primot Pos	385125
Wiener Pre	84375
Wiener Pos	119328
Wiener HP	301094

Tabla 11. Tabla de comparación de tiempos de computación

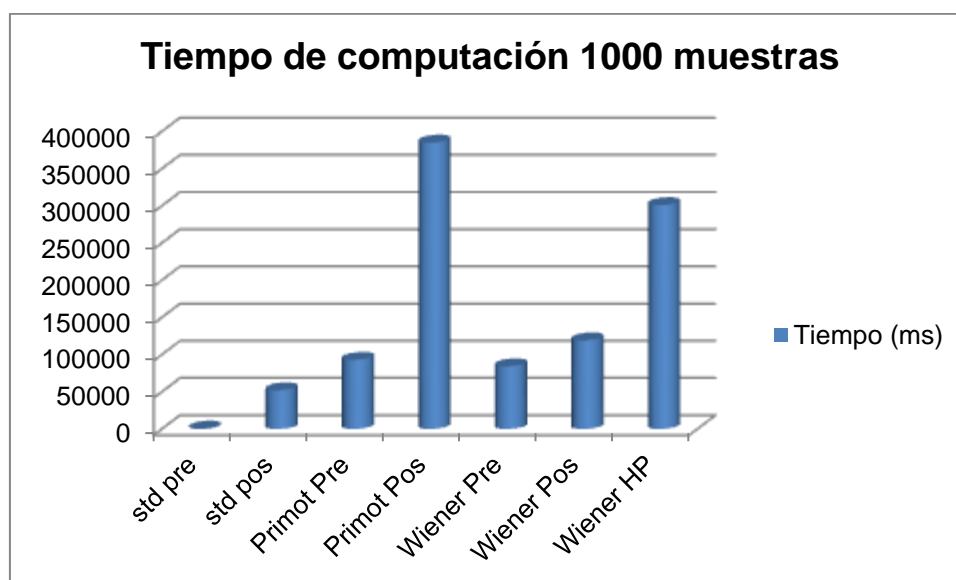


Ilustración 91. Gráfico de comparación de tiempos de computación en 1000 muestras

Cabe resaltar que los algoritmos de prepromedio siempre presentaran mejor rendimiento que los de pospromedio.

## 7. CONCLUSIONES

Como se ha podido observar los métodos de promediado (FASTCAM) ofrecen buenos resultados a costa de sacrificar la velocidad del algoritmo, presentando pues la necesidad de un compromiso entre el número de muestras a promediar, con su consecuente retardo, y el grado de calidad de la imagen resultante.

Estos algoritmos de promediado tendrán un resultado más pobre cuando se requiera una nitidez y un contraste alto, como en los casos de clasificación de estrellas binarias.

En ausencia de ruido se han obtenido mejores resultados en los algoritmos de promediado y deconvolución (prepromediado), tales como las estándares y el algoritmo de primot, pues se realizan deconvoluciones para contrarrestar los efectos del seeing. Hay que recordar que el seeing en los algoritmos de promediado supone, o bien una pérdida excesiva de imágenes si se desechan demasiadas conservándose muy pocas Lucky images o bien errores grandes si se toman demasiadas imágenes y se incluyen en el promedio imágenes difusas.

Por otro lado los algoritmos de regularización lineal aplicados a las imágenes dan lugar a buenos resultados en los casos en los que se tiene una buena y correcta concepción del ruido presente en las imágenes capturadas para poder hacer una buena regularización. Una regularización excesiva dará lugar a unas imágenes demasiado suavizadas y un ajuste por debajo del factor de regularización supondrá una menor eliminación de ruido. Esto exige de nuevo establecer un compromiso en el ajuste del factor  $\gamma$  o  $\lambda$  según se trate de Wiener o Tikhonov.

Otros algoritmos probados suponen una mezcla de estos dos tipos, los algoritmos híbridos probados arrojan muy buenos resultados en imagen pero hay que considerar que tienen debido a su naturaleza las debilidades de los algoritmos de promedio en cuanto al tiempo de retardo. En este caso el compromiso es doble, el número de imágenes de población para la estimación y el tiempo de adquisición y computación están sensiblemente ligados y por otra parte hay que considerar el ajuste del parámetro de regularización. Si se llega a un buen equilibrio en los parámetros de los algoritmos híbridos puede ser una cuestión meramente empírica y quizá específica al caso pero el resultado puede llegar a ser óptimo con un menor sacrificio de tiempo por un relativamente bajo coste en computación.

## 7.1 Líneas futuras de investigación

Como ya se comentó en la sección de introducción de los algoritmos y tal como se señala en (Starck & Pantin, 2002), los métodos de regularización lineal presentan un buen número de limitaciones o desventajas:

- *Aparición de oscilaciones de Gibbs*: en la vecindad de las discontinuidades de los datos. La calidad visual de la imagen se ve degradada.
- *No se puede usar información a priori*: por ejemplo, aunque sepamos que los datos negativos no son una solución posible generalmente, no podremos descartarlos.
- *Dado que la función ventana será un paso bajo, la resolución se verá degradada*. Existe un compromiso entre la resolución que queramos alcanzar y el nivel de ruido presente en la solución. Los métodos basados en wavelet no tendrán esta limitación.
- *Insensibilidad al error de estimación de la PSF*: no considera los posibles errores cometidos en la estimación y o captura de la PSF, lo que limita su interés en la práctica.

Algunas de estas debilidades de la regularización lineal se pueden subsanar totalmente o al menos en parte en algunas regularizaciones no lineales, como la regularización basada en wavelets. Este método es uno de los que mejor parados salen en comparaciones de otros trabajos como el ya mencionado (Starck & Pantin, 2002).

El siguiente paso a la hora de añadir más algoritmos a la comparación, será desarrollar ciertas variantes de wavelets. Las wavelets realizan una división espectral de toda la información de la imagen para poder procesar cada banda según convenga para conseguir un mejor resultado.

Cabe reseñar que todos los resultados presentados en este trabajo proceden de muestras sintéticas y por tanto el resultado en un entorno de imágenes reales puede verse alterado con respecto a lo observado. Debido a la obvia naturaleza de este trabajo, un proyecto fin de carrera, las limitaciones de tiempo han impedido una batería de pruebas con muestras reales para compensar los resultados teóricos.

## LA HERRAMIENTA IMAGEJ

### A.1 Introducción

ImageJ es un programa de procesamiento de imágenes basado en Java [11] creado por el NIH (*National Institutes of Health*). Está inspirado en el programa Image del NIH para Macintosh. Se puede ejecutar, bien como una applet online o como una aplicación descargable, en cualquier ordenador con una máquina virtual de Java 1.4 o posterior. Se pueden descargar distribuciones para Windows, Mac OS, Mac OS X y Linux.

Permite mostrar, editar, analizar, procesar, guardar e imprimir imágenes de 8, 16 y 32 bits. Puede leer muchos formatos de imagen incluyendo RIFF, GIF, JPEG, BMP, DICOM, FITS e imágenes crudas (“raw”), y soporta “stacks”, una serie de imágenes que comparten una única ventana. Es multihilo, por lo que operaciones que consumen mucho tiempo como leer archivos de imágenes pueden realizarse en paralelo con otras operaciones, aprovechando al máximo las ventajas de los actuales procesadores multicore.

Puede calcular estadísticas de los valores de los píxeles y el área de selecciones definidas por el usuario, medir distancias y ángulos, crear histogramas de densidad y perfiles de línea. Soporta funciones de procesamiento de imágenes estándares como manipulación de contrastes, sharpening, suavizado, detección de bordes y filtrado de la mediana.

Lleva a cabo transformaciones geométricas como escalado, rotación y giros. La imagen se puede ampliar hasta 32:1 y disminuir hasta 1:32. Todas las funciones de análisis y procesamiento están disponibles para cualquier factor de magnificación. El programa soporta cualquier número de ventanas (imágenes) de manera simultánea, estando éste limitado únicamente por la memoria disponible.

Dispone de calibración espacial para proporcionar medidas del mundo real en unidades como milímetros. También está disponible la calibración de densidad o de escala de grises.

ImageJ fue diseñado con una arquitectura abierta que proporciona extensibilidad mediante plugins y macros. Los plugins son escritos en Java y compilados a archivos .class. Las macros, escritas en el lenguaje de macros pseudo-Java de ImageJ, se guardan en ficheros .txt. Los plugins se ejecutan más rápido y son más flexibles, pero las macros son más sencillas de escribir y depurar. Los plugins y macros que se encuentren en la carpeta Plugins, con un guión bajo en su nombre, se muestran en la herramienta debajo de dicho menú. Los que se encuentren en subcarpetas de la carpeta Plugins se muestran en submenús.

Todos los algoritmos implementados en este TFM van a ser macros y se van a instalar dentro de una carpeta llamada “Reduce Images”. Esta carpeta ha de guardarse dentro de la carpeta Plugins, de forma que al abrir ImageJ aparezca un nuevo menú, “Reduce Images”.

## A.2 Conceptos básicos

### A.2.1 ImageJ es software libre

ImageJ es un software de código abierto de dominio público. Un usuario de ImageJ tiene las cuatro libertades esenciales definidas por Richard Stallman en 1986:

1. La libertad para ejecutar el programa, para cualquier propósito.
2. La libertad para estudiar cómo funciona el programa, y cambiarlo para que haga lo el usuario desee.
3. La libertad para redistribuir copias de manera que se pueda ayudar a la gente que le rodea.
4. La libertad para mejorar el programa, y hacer públicas las mejoras de forma que toda la comunidad se beneficie.

### A.2.2 Funcionalidad básica de la herramienta ImageJ

#### Ventanas

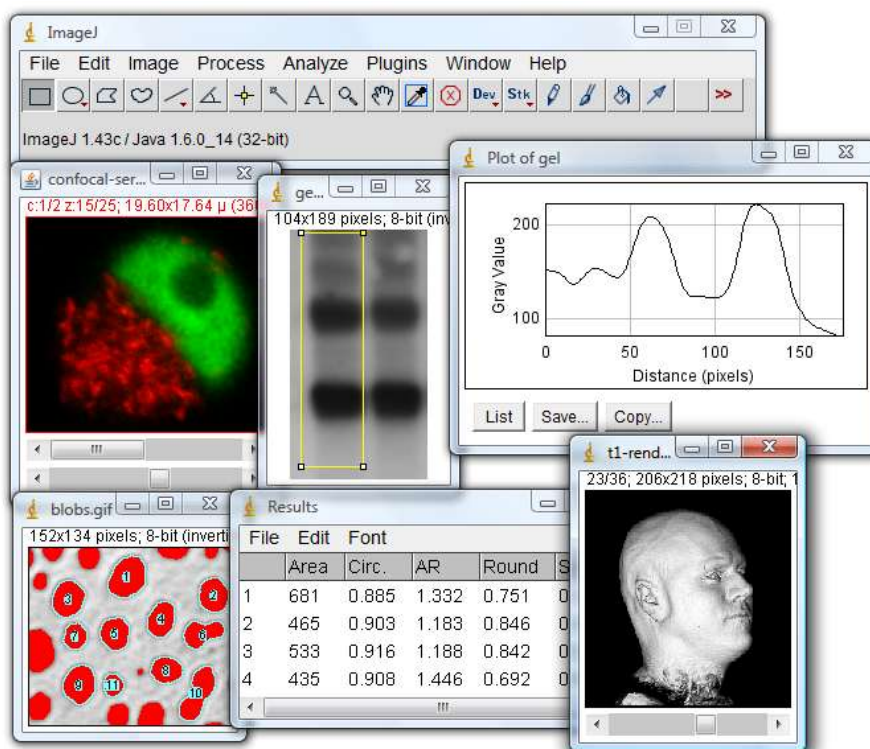


Ilustración 92: Diferentes tipos de ventanas de ImageJ

La ventana "ImageJ" contiene una barra de menú, una barra de herramientas, una barra de estado y una barra de progreso. Las imágenes, los histogramas, etc. se muestran en ventanas adicionales. Los resultados de las medidas se pueden ver en la ventana "Resultados".

Las ventanas pueden ser arrastradas por la pantalla y ser redimensionadas. Los histogramas y gráficos son ventanas de imagen ordinarias que pueden ser copiadas al portapapeles, editadas, imprimidas y guardadas.

### *Barra de herramientas*



Ilustración 93: Barra de herramientas de ImageJ

La barra de herramientas contiene herramientas para hacer selecciones, para hacer zoom o desplazarse en las imágenes y para cambiar el color con el que dibujar. Si se sitúa el puntero del ratón sobre una herramienta se muestra una descripción en la barra de estado. Las herramientas y menús de la parte derecha de la barra de herramientas se crean usando los macros definidos en el archivo ImageJ/macros/StartupMacros.txt.

### *Barra de estado*

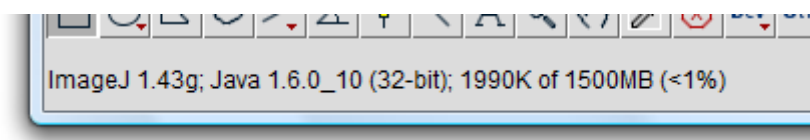


Ilustración 94: Barra de estado de ImageJ

La barra de estado, cuando el cursor está situado sobre una imagen, muestra las coordenadas y los valores del píxel. Tras ejecutar un filtro, se muestra el tiempo transcurrido y el ratio de procesado en píxeles/segundo.

Si se hace clic en el estado se mostrará, como se observa en la imagen, la versión de ImageJ, la versión de Java, la memoria en uso, la memoria disponible y el porcentaje de memoria utilizado.

### *Barra de progreso*



Ilustración 95: Barra de progreso de ImageJ

La barra de progreso, situada a la derecha de la barra de estado, muestra el progreso de las operaciones que necesitan un cierto tiempo para completarse. No aparece si la operación requiere menos de un segundo.

## Imágenes

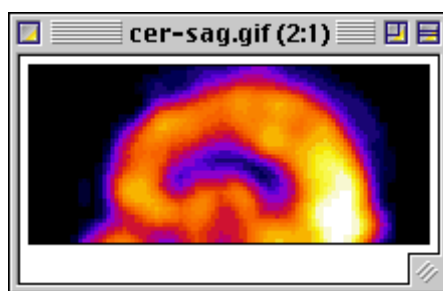


Ilustración 96: Imagen mostrada por ImageJ

ImageJ permite que múltiples imágenes se muestren en la pantalla al mismo tiempo. La ventana activa tiene la barra de título resaltada.

ImageJ soporta imágenes en escala de grises de 8, 16 y 32 bits e imágenes en color de 8 y 32 bits. Las imágenes de 8 bits se representan utilizando enteros sin signo en el rango 0-255. Las imágenes de 16 bits son enteros sin signo en el rango 0-65535 y las imágenes en escala de grises de 32 bits usan números en punto flotante.

Las imágenes en escala de grises de 16 y 32 bits no se pueden visualizar directamente en las pantallas de los ordenadores, que normalmente sólo pueden mostrar 256 niveles de gris. Por ello, los datos son mapeados a 8 bits mediante enventanado. La ventana define el rango de valores de gris que se muestran: los valores menores que los de la ventana se convierten a negro, mientras que los valores por encima de la ventana son blancos. La ventana se define por valores máximos y mínimos que pueden modificarse utilizando *Image>Adjust>Brightness/Contrast*.

## Stacks

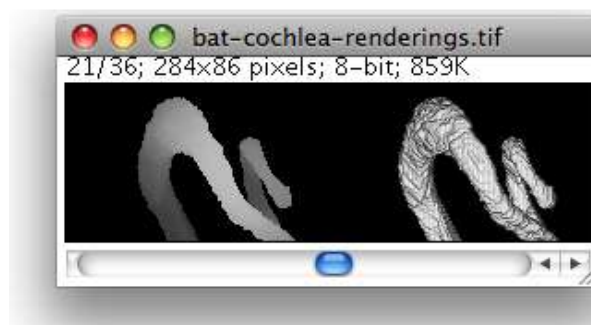


Ilustración 97: Stack mostrado por ImageJ

ImageJ puede mostrar múltiples imágenes relacionadas espacial o temporalmente en una única ventana. Estos conjuntos de imágenes se denominan *stacks*. Las imágenes que forman un *stack* se llaman *slices*. Todas las *slices* de un *stack* deben tener el mismo tamaño y profundidad de bit. Una barra de desplazamiento permite moverse por las *slices*. La mayoría de los filtros de ImageJ tienen como opción la de procesar todas las imágenes de un *stack*.

ImageJ abre ficheros TIFF multi-imágenes como un *stack* y guarda los stacks como TIFFs multi-imágenes. El comando *File>Import>Raw* abre ficheros multi-imágenes sin comprimir. *File>Import>Image Sequence* abre una carpeta de imágenes como un *stack*. Para crear un nuevo stack, simplemente se selecciona *File>New>Image* y en el campo “*Slices*” se introduce un valor mayor que uno. El submenú *Image>Stacks* contiene comandos para operaciones comunes con *stacks*.

## Selecciones



Ilustración 98: Distintos tipos de selecciones que pueden elegirse en ImageJ

Las selecciones son áreas o líneas definidas por el usuario en una imagen. Sólo puede estar activa una selección cada vez, salvo que se realice una selección múltiple. Las selecciones de área se crean utilizando las herramientas de selección rectangular, elíptica, poligonal y hecha a pulso. Las áreas seleccionadas pueden ser medidas (*Analyze>Measure*), filtradas, rellenadas (*Edit>Fill*) o dibujadas (*Edit>Draw*).

Las selecciones de líneas son creadas utilizando las herramientas de selección de línea recta, línea segmentada y línea dibujada a pulso. Se emplea *Edit>Draw* para dibujar la línea del color actual. La longitud de las líneas seleccionadas puede medirse utilizando *Analyze>Measure*.

Las selecciones se pueden mover haciendo clic y arrastrando. La barra de estado muestra las coordenadas de la esquina superior izquierda de la selección. El cursor cambia a una flecha cuando está dentro de la selección. Para mover los contenidos de una selección rectangular, en lugar de la selección en sí misma, se utiliza *Edit>Copy* (c), *Edit>Paste* (v), y entonces hacer clic en la selección y arrastrar. Se utilizan las teclas de dirección para mover las selecciones un píxel cada vez en cualquier dirección.

Las selecciones rectangulares y elípticas pueden ser redimensionadas. Cuando la selección se redimensiona, la anchura y la altura se muestran en la barra de estado. Se utilizan las teclas de dirección junto con la tecla Alt para estirar las selecciones píxel a píxel.

Para eliminar una selección, se elige cualquiera de las herramientas de selección y se hace clic fuera de la selección, o se usa *Edit>Selection>Select None* (shift-a). Se emplea *Edit>Selection>Restore Selection* (shift-e) para recuperar una selección después de haberla eliminado.

Una selección se puede traspasar de una ventana de imagen a otra activando la ventana destino y utilizando *Edit>Selection>Restore Selection*. Las selecciones se



pueden guardar en disco usando *File>Save As>Selection* y recuperadas mediante *File>Open*. Una herramienta especializada, el ROI Manager, se emplea para trabajar con selecciones múltiples.

### *Formatos de archivo*

El comando *File>Open* abre imágenes TIFF, GIF, JPEG, PNG, DICOM, BMP, PGM y FITS. También abre tablas de búsqueda y selecciones. Además, el submenú *File>Import* proporciona acceso a plugins para leer archivos “raw”, imágenes en formato ASCII, y para cargar imágenes en red utilizando una URL. Para importar un archivo “raw”, se debe conocer cierta información sobre su distribución, incluyendo el tamaño de la imagen.

Los archivos pueden guardarse en formato TIFF, GIF, JPEG, PNG, PGM, FITS, texto delimitado por tabulaciones y “raw”.

### *Plugins*

La funcionalidad de ImageJ puede expandirse mediante el uso de plugins escritos en Java. Los plugins permiten soportar nuevos formatos de archivo o filtrar o analizar imágenes.

Los plugins guardados en la carpeta “plugins” de ImageJ se instalan automáticamente en el menú Plugins. También pueden instalarse en otros menús utilizando *Plugins/Hot Keys/Install Plugin*. Los plugins pueden ser creados o modificados empleando *Plugins/Edit*.

### *Macros*

Una manera cómoda de trabajar en el entorno de ImageJ es mediante la automatización de procesos que ofrece la herramienta de Macros. ImageJ dispone de un lenguaje de comandos interpretados por el mismo programa para crear scripts.

Este lenguaje de alto nivel permite llevar un paso adelante la complejidad de algunos plugins y combinarlos para obtener complejos algoritmos sin necesidad de compilar mediante la máquina virtual de java un ejecutable.

Existen extensas guías de programación de macros de ImageJ, han sido citadas en la bibliografía de este trabajo.

## Tablas de búsqueda



**Ilustración 99:** Tabla de búsqueda en ImageJ

Las imágenes en escala de grises son mostradas utilizando una tabla de búsqueda de color que describe el color que debe usarse para cada uno de los 256 posibles valores de pixel. Se pueden seleccionar paletas de color alternativas en el submenú *Image/Lookup Tables*. Se usa *Image/Adjust/Brightness/Contrast* para mejorar imágenes cambiando dinámicamente el mapeado de la tabla de búsqueda y *Analyze/Show LUT* para mostrar la tabla de búsqueda de la imagen activa.

### A.3 ImageJ y la computación de la FFT

La herramienta ImageJ es un software muy enfocado al apartado gráfico. Esto hace que a menudo no sea muy específico o útil en cuanto a cuestiones matemáticas.

Cuando se trabaja en espectro es importante considerar en que espacios nos movemos. La transformada rápida de Fourier esta computada en ImageJ sin usar la información compleja del espectro (no biespectro) ya que se realiza mediante la transformada discreta de Hartley:

$$H(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) \text{cas}(\omega t) dt \quad [25]$$

Siendo la función cas:

$$\text{cas}(t) = \cos(t) + \sin(t) \quad [26]$$

Pudiendo obtenerse la transformada de Fourier como:

$$F(\omega) = \frac{H(\omega) + H(-\omega)}{2} - j \frac{H(\omega) - H(-\omega)}{2} \quad [27]$$

Una de las mayores ventajas es que los coeficientes de esta transformada son números reales (complejos en la transformada de Fourier). Las transformadas de Hartley y la de Fourier tienen propiedades equivalentes.

Elegir la transformada de Fourier o la HRT depende del número de operaciones que requiere una aplicación concreta, en nuestro caso trabajamos con imágenes con muchos píxeles y por tanto muchas operaciones son necesarias.

Uno de los pequeños contrapuntos de esto se puede encontrar por ejemplo cuando se quiere realizar una correlación.

$$g \otimes h = \int g(t)h(\tau - t) dt \quad [28]$$

$$g \otimes h = \mathcal{F}^{-1}(\mathcal{F}(G) \cdot \mathcal{F}(H)) \quad [29]$$

Sin embargo la correlación que se lleva a cabo en ImageJ no es rigurosamente una correlación pues se hace la suposición

$$\mathcal{F}^{-1}(\mathcal{F}(G) \cdot \mathcal{F}(H)) = \text{IFHT}(\text{FHT}(G) \cdot \text{FHT}(H)) \quad [30]$$

Siendo esta incorrecta. La comprobación se llevó a cabo en MATLAB reduciendo un caso de convolución de una imagen de 4 píxeles y a realizando esta prueba teóricamente a lápiz.

La diferencia entre unos resultados y otros reside en el tratamiento de colas durante las convoluciones. ImageJ realiza una extensión de la imagen en espejo desde los laterales de las imágenes y MATLAB hace un relleno de ceros. Esto no debería ser ningún tipo de cortapisa en los usos de ImageJ de la FFT, pero es conveniente reseñarlo.

## Bibliografía

- A. Glindemann, S. H. (1999, September). Adaptive Optics on Large Telescopes. 41. Germany.
- Burghera, L., Litvina, I., & Forbesa, A. (2008, Abril). Simulating atmospheric turbulence using a phase-only spatial light modulator. *South African Journal of Science*(104), 129 - 134.
- Dong feng Shi, C. F. (2011, Noviembre ). Reconstruction of spatially misaligned and turbulence degraded images. Heifei, China.
- Noll, R. J. (1976, Marzo). Zernike polynomials and atmospheric turbulence. *Optical Society of America*, 66(3), 207-210.
- Primot, J., Rousset, G., & Fontanella, J. (1990, Septiembre). Deconvolution from wave-front sensing: a new technique. *J. Opt. Soc. Am. A/Vol. 7, No. 9*, pages 1 - 11.
- Starck, J., & Pantin, E. (2002, October). Deconvolution in Astronomy: A review. *Astronomical Society of the Pacific*(114), 1051-1069.
- Team, F. D. (2012, April). *Introduction to macro programming*. Retrieved from [http://fiji.sc/wiki/index.php/Introduction\\_into\\_Macro\\_Programming](http://fiji.sc/wiki/index.php/Introduction_into_Macro_Programming)
- Werner, B. (2006, July 2). Writing ImageJ Plugins – A Tutorial. Hagenberg, Austria.