



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Instituto Universitario de Microelectrónica Aplicada
Sistemas de información y Comunicaciones

Máster en Tecnologías de Telecomunicación



Trabajo Fin de Máster

COMPRESIÓN DE IMÁGENES HIPERESPECTRALES MEDIANTE TÉCNICAS DE FUSIÓN Y DESMEZCLADO ESPECTRAL

Autor: José María Melián Álamo
Tutor(es): Sebastián López Suárez
Roberto Sarmiento Rodríguez
Fecha: Enero de 2016



t +34 928 451086 | iuma@iuma.ulpgc.es
f +34 928 451083 | www.iuma.ulpgc.es

Campus Universitario de Tafira
35017 Las Palmas de Gran Canaria



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Instituto Universitario de Microelectrónica Aplicada
Sistemas de información y Comunicaciones

Máster en Tecnologías de Telecomunicación



Trabajo Fin de Máster

COMPRESIÓN DE IMÁGENES HIPERESPECTRALES MEDIANTE TÉCNICAS DE FUSIÓN Y DESMEZCLADO ESPECTRAL

HOJA DE FIRMAS

Alumno/a: José María Melián Álamo Fdo.:

Tutor/a: Sebastián López Suarez Fdo.:

Tutor/a: Roberto Sarmiento Rodríguez Fdo.:

Fecha: Enero de 2016



t +34 928 451086 iuma@iuma.ulpgc.es
f +34 928 451083 www.iuma.ulpgc.es

Campus Universitario de Tafira
35017 Las Palmas de Gran Canaria



Máster en Tecnologías de Telecomunicación



Trabajo Fin de Máster

COMPRESIÓN DE IMÁGENES HIPERESPECTRALES MEDIANTE TÉCNICAS DE FUSIÓN Y DESMEZCLADO ESPECTRAL

HOJA DE EVALUACIÓN

Calificación:

Presidente Fdo.:

Secretario Fdo.:

Vocal Fdo.:

Fecha:



MEMORIA DESCRIPTIVA	1
CAPÍTULO 1. INTRODUCCIÓN Y JUSTIFICACIÓN.....	3
1.1.- <i>Introducción a las diferentes tipologías de imagen</i>	5
1.2.- <i>Motivación del Trabajo</i>	6
1.3.- <i>Propuestas planteadas</i>	7
1.4.- <i>Objetivo del trabajo fin de máster.</i>	8
CAPÍTULO 2. ESTRATEGIAS DE COMPRESIÓN PROPUESTAS	9
2.1 <i>Estrategia de compresión basada en Data Fusion</i>	11
2.2 <i>Estrategia de compresión basada en unmixing empleando el algoritmo FUN</i>	16
CAPÍTULO 3. SENSORES HIPERESPECTRALES, MÉTRICAS Y ESTADO DEL ARTE.....	23
3.1.- <i>Información general</i>	25
3.1.1.- Sensores hiperespectrales.....	25
3.1.1.1.- Airbone Visible / Infrared Imagine Spectrometer (AVIRIS).....	25
3.1.1.2.- Compact Reconnaissance Imaging Spectrometer for Mars (CRISM)	26
3.1.1.3.- Atmospheric Infrared Sounder (AIRS) e Infrared Atmospheric Sounding Interferometer (IASI)	27
3.1.2.- Métricas utilizadas.....	28
3.1.3.- Algoritmos de referencia	29
3.1.3.1.- Computationally Efficient algorithm for Fusing Multispectral and Hyperspectral Images (CoEF-MHI).....	29
3.1.3.2.- Generalized Gram-Schmidt Adaptive (GGSA) y Generalized Local Minimum Square Error (GLMMSE).....	30
3.1.3.3.- Principal Component Analysis (PCA).....	30
3.1.3.4.- Lossy Compression for Exomars (LCE)	31
3.1.3.5.- Consultative Committee for Space Data System (CCSDS).....	32
CAPÍTULO 4. RESULTADOS	33
4.1.- <i>Planteamiento de las simulaciones</i>	35
4.1.1.- Resultados obtenidos con la imagen del sensor AIRS	36
4.1.2.- Resultados obtenidos con la imagen del sensor AVIRIS	37
4.1.3.- Resultados obtenidos con la imagen del sensor CRISM	39
4.1.4.- Resultados obtenidos con la imagen del sensor IASI.....	40
4.2.- <i>Resumen de resultados</i>	41
4.3.- <i>Compresión adicional mediante algoritmos CCSDS</i>	42
4.3.1.- CCSDS-122	42
4.3.2.- CCSDS-123	44
4.4.- <i>Versatilidad del compresor CFUN</i>	47
4.4.1.- Posibilidad de realizar la compresión y descompresión de la imagen por macrobloques	47
4.4.2.- Posibilidad de realizar la compresión y descompresión de la imagen tanto por píxeles como por bandas	49
4.4.3.- Filtro de ruido de la imagen.....	50
CAPÍTULO 5. CONCLUSIONES.....	55
5.1.- <i>Conclusiones relacionadas con las propuestas de compresión de imágenes presentadas en este TFM</i>	57
REFERENCIAS.	61
ANEXOS	63
A.1.- <i>Anexo I</i>	65
A.1.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS. Métrica: SNR	65
A.1.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS. Métrica: SAM	68
A.2.- <i>Anexo II</i>	73
A.2.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS. Métrica: SNR	73
A.2.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS. Métrica: SAM	76
A.3.- <i>Anexo III</i>	81

A.3.1.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM. Métrica: SNR	81
A.3.2.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM. Métrica: SAM	84
A.4.- Anexo IV	89
A.4.1.- Gráficas de resultados de las simulaciones para la imagen del sensor IASI. Métrica: SNR	89
A.5.- Anexo V	93
A.5.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS CDF+CCSDS-122	93
A.5.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS CDF+CCSDS-122	95
A.5.3.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM CDF+CCSDS-122	97
A.5.4.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM CDF+CCSDS-122	99
A.6.- Anexo VI	101
A.6.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS CDF+CCSDS-123	101
A.6.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS. CDF+CCSDS-123	102
A.6.3.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM. CDF+CCSDS-123	103
A.6.4.- Gráficas de resultados de las simulaciones para la imagen del sensor IASI. CDF+CCSDS-123	104

MEMORIA DESCRIPTIVA

Capítulo 1. Introducción y justificación.

En este capítulo se introducen los conceptos de las diferentes tipologías de imágenes digitales: pancromáticas, multiespectrales, hiperespectrales y ultraespectrales que son la base de este trabajo. Se explican además, de forma simplificada, las nuevas propuestas de compresión de imágenes que se van a presentar en este trabajo de fin de máster, para terminar exponiendo los objetivos del mismo.

1.1.- Introducción a las diferentes tipologías de imagen

En la actualidad existen sensores capaces de medir la intensidad de la radiación en diferentes longitudes de onda. Con el empleo de estos sensores, se generaliza el concepto de imagen digital, y se obtienen imágenes en las que cada pixel pasa a ser un vector N_b -dimensional, en el que cada componente representa la radiación captada por dicho sensor en unas longitudes de onda determinadas. Atendiendo a la resolución espectral del sensor que captura la imagen digital, ésta tendrá un número determinado de bandas, N_b , pudiendo ser así clasificadas como imágenes pancromáticas, imágenes multispectrales, imágenes hiperespectrales e imágenes ultraspectrales. Las imágenes pancromáticas son aquellas que tienen una sola banda, que contiene, normalmente, la intensidad de la radiación captada por el sensor en el rango del espectro visible. Las imágenes multispectrales son aquellas que contienen un número reducido, típicamente menos de 100 bandas. Un caso particular de las imágenes multispectrales son las imágenes RGB, que contienen tres bandas comprendidas en los rangos espectrales correspondientes al rojo, al verde y al azul. No obstante, las imágenes multispectrales pueden contener bandas fuera de estos rangos espectrales, e incluso en los rangos infrarrojo y ultravioleta. Por último, las imágenes hiperespectrales y ultraspectrales son aquellas que tienen cientos y miles de bandas respectivamente, y son capaces de cubrir grandes zonas del espectro visible, del infrarrojo y del ultravioleta de forma casi continua.

Las imágenes descritas pueden ser representadas como un cubo de datos, en el cual dos dimensiones representan la dimensión espacial de la imagen, y la tercera dimensión se emplea para representar la dimensión espectral de la misma, tal y como se muestra en la Figura 1.

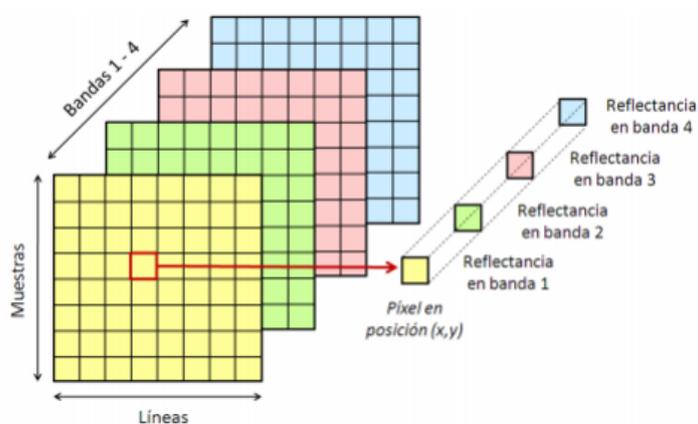


Figura 1. Representación de imagen multi-dimensional de 4 bandas

1.2.- Motivación del Trabajo

En la actualidad, el desarrollo tecnológico ha permitido incorporar sensores hiperespectrales y ultraespectrales a bordo de satélites con el objetivo de tomar datos de la Tierra de forma remota. Si bien esta tecnología ha avanzado en gran medida, sigue habiendo dificultades no solo a la hora de analizar la información obtenida sino también al enviarla desde el satélite, donde se sitúa el sensor, hasta la Tierra donde será almacenada, analizada y estudiada. Esto se debe a que actualmente la capacidad de adquisición de datos de los sensores hiperespectrales y ultraespectrales es mucho mayor que la capacidad de transmisión de información de los satélites en los que se encuentran. Lo que se explica sabiendo que el ancho de banda de la conexión con la Tierra es limitado y solamente se pueden transferir los datos a lo largo de ciertas horas del día. Por este motivo, ha adquirido gran importancia el desarrollo de técnicas de compresión de imágenes hiperespectrales en los últimos años. Además, este proceso resulta ser altamente complejo. En primer lugar, se desea alcanzar grandes ratios de compresión, ya que cuanto más grande son los ratios de compresión alcanzados, mayor es la cantidad de información de la que se podría disponer en tierra. Es importante también que el proceso de compresión y descompresión pierda poca información, o ninguna, ya que de no ser así, la información con la que se contaría en tierra no sería de gran utilidad. Además, los algoritmos empleados para llevar a cabo el proceso de compresión deben ser lo más simple posibles por diversos motivos: en primer lugar, por tratarse del espacio, los algoritmos son ejecutados por hardware específico, el cual tiene menos prestaciones, en términos de velocidad, que el hardware del que es posible disponer en tierra, ya que deben ser tolerantes a la radiación y deben tener mecanismos para soportar fallos y poderse recuperar automáticamente. Además, en el espacio, un menor consumo de potencia, de área o de tiempo supone una ventaja.

Las técnicas de compresión existentes se clasifican en dos categorías: sin pérdidas (lossless) y con pérdidas (lossy). En el primer caso, los algoritmos tratan de comprimir la información para que su coste de almacenamiento sea menor que el de la imagen original, sin embargo, a pesar de que con ellos se obtiene la imagen de partida al descomprimir la información, el ratio de compresión no es muy elevado pudiéndose obtener en algunos casos ratios de compresión inferiores a 1. Por ello, para mejorar los ratios respecto a los que se consiguen con la compresión lossless, se tiene un proceso equivalente pero con pérdidas de información. Estos últimos algoritmos consiguen mejores ratios de compresión que los

algoritmos lossless a costa de introducir pérdidas en el proceso de compresión y descompresión de la imagen. Actualmente, la investigación se centra sobre todo en métodos lossy [1].

1.3.- Propuestas planteadas

En este Trabajo de Fin de Máster (TFM) se pretenden desarrollar técnicas de compresión y descompresión alternativas, específicas para imágenes hiperespectrales y ultraespectrales, desarrolladas desde su concepción con el objetivo de que sean lo más simples y paralelizables posible, facilitando así su implementación en el hardware paralelizable para aplicaciones que requieran de tiempo real. Con este objetivo se proponen dos técnicas de compresión:

1. Desarrollar un proceso de compresión con pérdidas sencillo que se base en obtener a partir de una imagen hiperespectral o ultraespectral: una imagen multiespectral con el mismo número de píxeles que la original pero con un menor número de bandas, y una imagen hiperespectral con menos información espacial pero con la misma información espectral. Se plantea así de forma que la imagen original pueda ser reconstruida posteriormente utilizando algoritmos de Data Fusion, los cuales ya han demostrado ser capaces de fusionar imágenes multiespectrales e hiperespectrales con muy buenos resultados [2][3][4][5]. Pensando en aplicaciones de remote sensing, esto permitiría realizar un proceso sencillo de compresión on-board dejando que el proceso complejo sea el de reconstrucción de la imagen, ya que éste se desarrollaría en tierra, suponiendo esto una ventaja respecto a la manera de proceder actual en la cual la parte más complicada, la compresión, se realiza on-board.
2. Utilizar una modificación del algoritmo Fast algorithm for linearly UNmixing hyperspectral images (algoritmo FUN) [6] concebido para aplicaciones de desmezclado espectral, de forma que pueda ser aplicable con fines de compresión de imagen. El proceso llevado a cabo por este algoritmo tiene dos importantes ventajas. En primer lugar, no se realizan operaciones matriciales complejas y es altamente paralelizable, y en segundo lugar, permite especificar a priori el ratio de compresión y/o el SNR que se desea obtener en la reconstrucción. Además, la descompresión de la imagen se reduce a un simple producto matricial.

1.4.- Objetivo del trabajo fin de máster.

Como se ha podido ver en el apartado anterior, los objetivos del proyecto que engloba a este TFM se pueden resumir en desarrollar dos nuevas formas de realizar la compresión de imágenes hiperespectrales y ultraespectrales. La primera línea estaría basada en el desarrollo de un compresor sencillo que obtenga dos imágenes: una hiperespectral y otra multiespectral, ambas a partir de la imagen original de forma que al unir las, haciendo uso de algoritmos de Data Fusion, se obtenga una imagen prácticamente igual a la de partida, resultado que dependerá de la calidad del algoritmo de Data Fusion utilizado. La segunda línea de trabajo hará uso de una modificación del algoritmo FUN que permita llevar a cabo la compresión de las imágenes empleando únicamente operaciones matriciales sencillas y altamente paralelizables, así como introducir el ratio de compresión y/o el SNR que se desea obtener en la reconstrucción de la imagen.

Una vez desarrolladas las propuestas anteriores, para verificar la bondad de los procedimientos formulados se compararán los resultados de las simulaciones realizadas con las nuevas metodologías con los que se obtienen con los algoritmos que conforman el *estado del arte* en lo que a compresión hiperespectral y ultraespectral se refiere. Para ello, se hará uso de una serie de imágenes hiperespectrales reales obtenidas por diferentes sensores hiperespectrales usados en la actualidad (AIRS, AVIRIS, CRISM e IASI).

Capítulo 2. Estrategias de compresión propuestas

En este trabajo fin de máster se van a presentar dos nuevas formas de realizar procesos de compresión y descompresión de imágenes hiperespectrales: una que hará uso de un compresor lossy sencillo que necesitará algoritmos de Data Fusion durante el proceso de descompresión para la obtención de la imagen original, y la otra que utilizará una modificación realizada sobre el algoritmo FUN de desmezclado espectral. Para explicar con claridad ambas estrategias se procederá al desarrollo por separado de las mismas.

2.1 Estrategia de compresión basada en Data Fusion

Existe en la actualidad un compromiso entre la resolución espacial y la espectral de los satélites empleados para la obtención de imágenes desde el espacio. De este modo, se tienen sensores pancromáticos, capaces obtener imágenes con una gran resolución espacial pero poca resolución espectral y sensores hiperespectrales y ultraespectrales, capaces de conseguir imágenes con una gran resolución espectral pero poca resolución espacial. Por este motivo, y con el fin de aprovechar la existencia de ambos sensores, se han propuesto a lo largo de los últimos años una serie de algoritmos que permiten llevar a cabo la fusión de imágenes hiperespectrales y multiespectrales obteniendo imágenes hiperespectrales de gran resolución espacial.

Hasta ahora, los algoritmos desarrollados para fusionar imágenes multiespectrales e hiperespectrales se habían pensado con el objetivo de mejorar los resultados conseguidos en las aplicaciones tradicionales de la tecnología hiperespectral, como son las de unmixing o clasificación, entre otras. No obstante, en este TFM se van a utilizar con el propósito de reconstruir una imagen tras haber sido comprimida previamente gracias al compresor que se propondrá a continuación, el cual va a permitir obtener partiendo de una imagen hiperespectral o ultraespectral, dos imágenes distintas: una imagen hiperespectral con menos resolución espacial que la original pero con el mismo número de bandas, y una imagen multiespectral con menos resolución espectral que la imagen inicial pero con la misma resolución espacial. De esta manera, se conseguirían dos imágenes que en conjunto ocupan menos que la imagen hiperespectral de la que proceden, pudiendo ésta ser obtenida nuevamente empleando para ello algoritmos de Data Fusion.

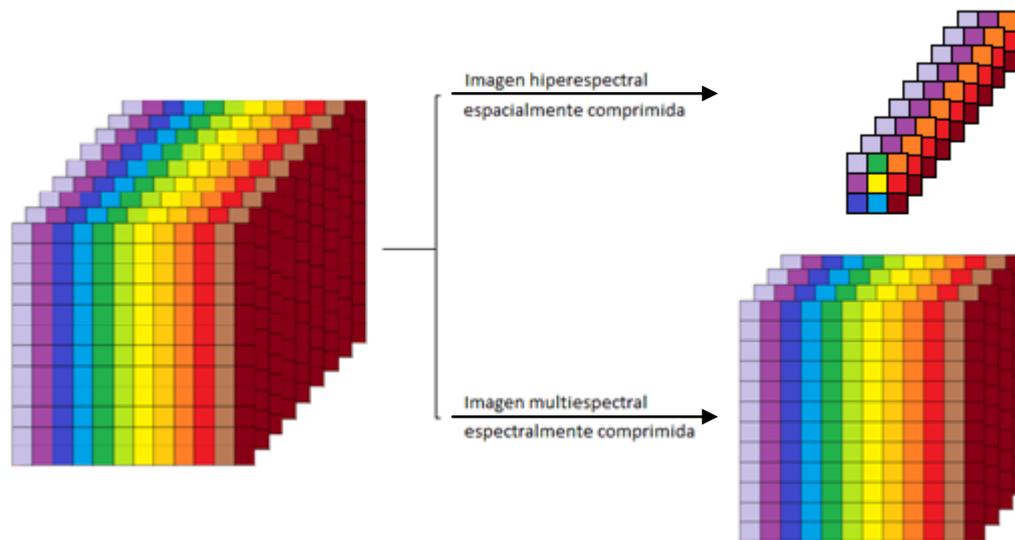


Figura 2. Representación del funcionamiento del compresor (CDF)

2.1.1.1. Compresor (CDF)

El compresor propuesto, que será referido a partir de ahora como CDF al necesitar algoritmos de Data Fusion para descomprimir la imagen, realiza dos procesos de compresión independientes que son los que permiten obtener: por un lado la imagen hiperspectral espacialmente reducida, y por otro lado la imagen multispectral, espectralmente comprimida. Los parámetros de entrada para el compresor CDF son los siguientes:

H_y: Imagen hiperspectral original

P_p: Relación lineal entre el número de píxeles de la imagen a comprimir y la imagen hiperspectral de baja resolución espacial

P_b: Relación entre el número de bandas de la imagen a comprimir y la imagen multispectral.

Siendo a su vez los parámetros de salida:

H_{yR}: Imagen hiperspectral con reducción de su resolución espacial

MR: Imagen multispectral tras la realización de una reducción espectral

rc: Ratio de compresión conseguido

A continuación se van a exponer las diferentes maneras de realizar la compresión espacial y espectral en las imágenes hiperespectrales de entrada al compresor CDF:

a. Realización de la compresión espacial:

a.1.) Sin promedio en píxeles: En este caso se procederá a comprimir la imagen seleccionando el primer píxel de cada cuadrado de P_p píxeles de lado de la imagen (Figura 3).

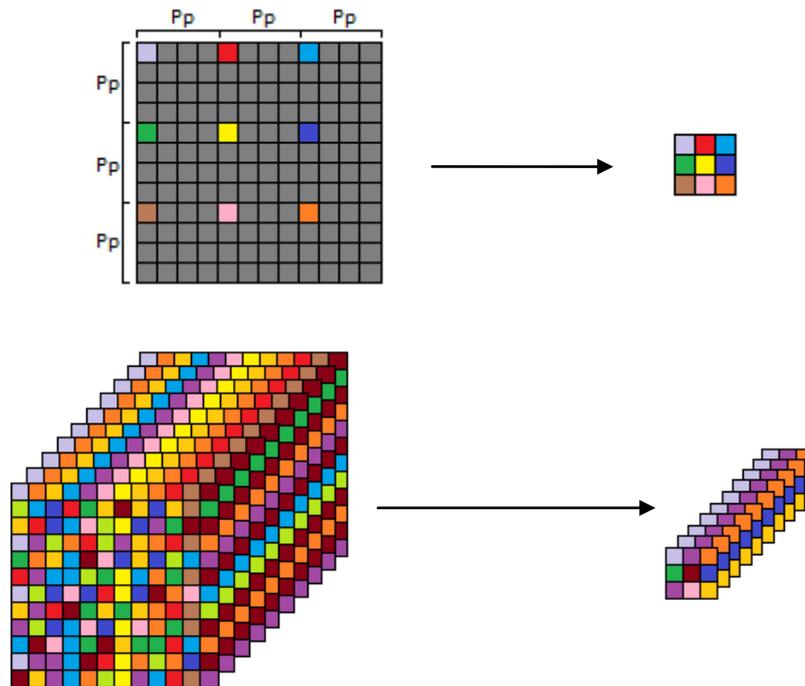


Figura 3. Reducción de la componente espacial de la imagen hiperespectral sin realizar promedio en píxeles

a.2.) Promediando en píxeles: En esta configuración, se pasaría a obtener un promedio con la información de cada P_p^2 píxeles (Figura 4).

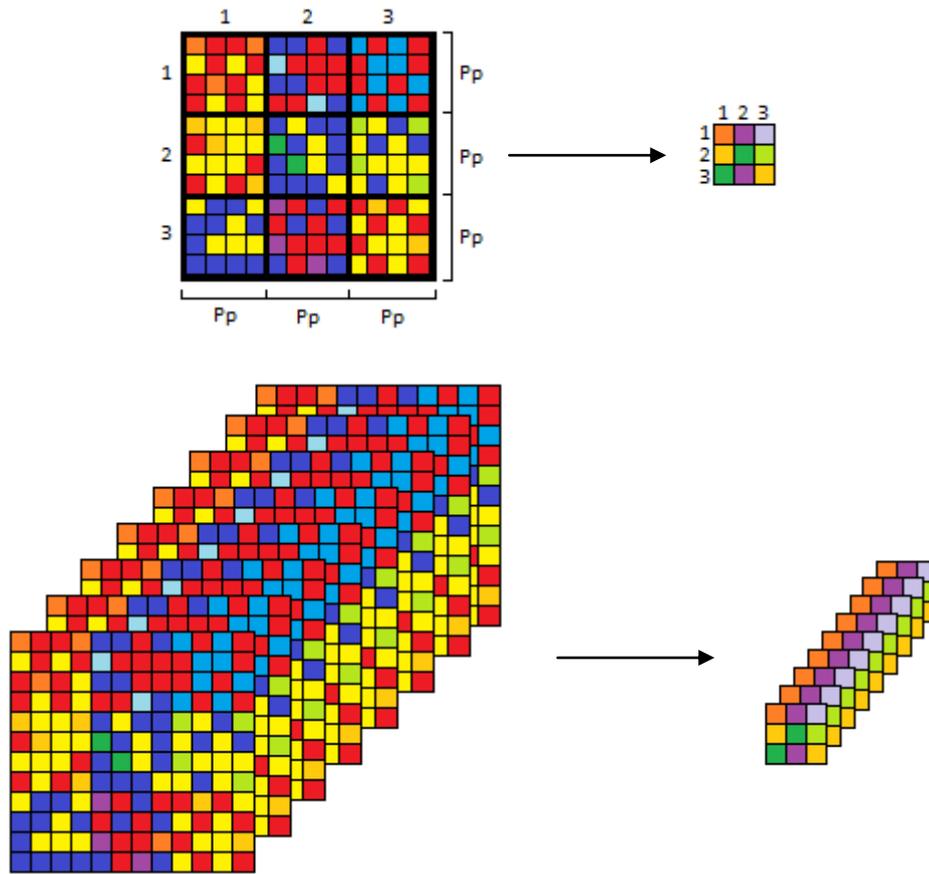


Figura 4. Reducción de la componente espacial de la imagen hiperespectral realizando promedio en píxeles ($P_p = 4$)

b. Realización de la compresión espectral

b.1.) Sin promedio en bandas: En este caso, se escogerá una banda de la imagen hiperespectral de cada P_b bandas, tal y como se observa en la Figura 5:

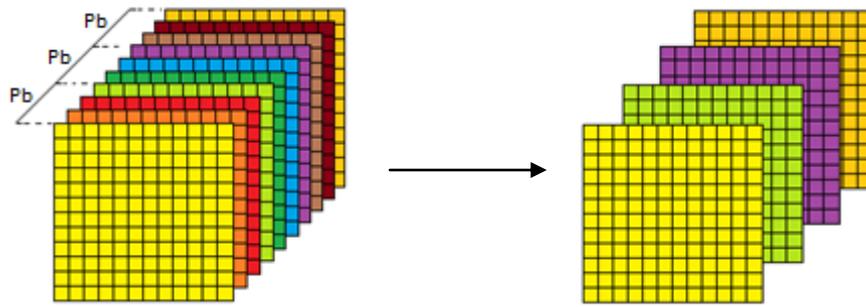


Figura 5. Reducción de la componente espectral de la imagen hiperespectral sin realizar promedio en bandas ($P_b = 3$)

b.2.) Con promedio en bandas: Según este criterio, se comprimirá la imagen espectralmente realizando un promedio de cada P_b bandas de la imagen hiperespectral (Figura 6).

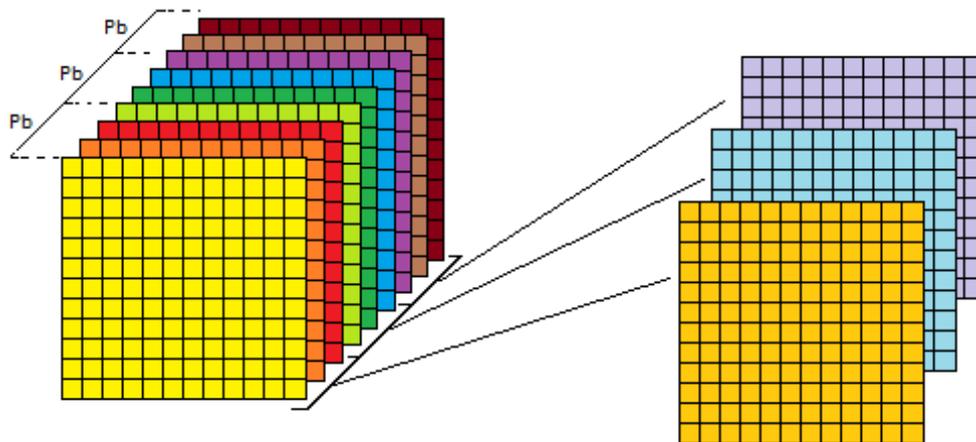


Figura 6. Reducción de la componente espectral de la imagen hiperespectral realizando promedio en bandas ($P_b = 3$)

Para la realización de las simulaciones se usarán cuatro combinaciones diferentes de los cuatro criterios anteriores a la hora de comprimir la imagen con el objetivo de decidir cual da mejores resultados:

1. Sin promedio, ni en bandas ni en píxeles (a.1. + b.1.).
2. Promediando en píxeles pero no en bandas (a.2. + b.1.).
3. Promediando en bandas y no en píxeles (a.1. + b.2.).
4. Promediando tanto en píxeles como en bandas (a.2. + b.2.).

Una de las ventajas que proporciona el compresor CDF es que es muy sencillo y rápido, altamente paralelizable, y que no realiza en ningún momento operaciones computacionalmente complejas, ya que el proceso complejo será el realizado por los algoritmos de Data Fusion, proceso que tiene lugar en tierra y es utilizado para reconstruir nuevamente la imagen.

Con este compresor, el ratio de compresión que se alcanza sigue la siguiente fórmula:

$$rc = \frac{1}{\frac{1}{Pp^2} + \frac{1}{Pb}} \quad (2.1)$$

Atendiendo a la expresión 2.1, se observa que se puede conseguir un amplio abanico de ratios de compresión, para lo cual solamente habría que realizar combinaciones con los ratios en píxeles y en bandas vistos con anterioridad. Otra ventaja importante que presenta el compresor CDF es el hecho de que permite actuar tanto en la resolución espacial (variando Pp) como en la espectral (modificando Pb), de forma que en imágenes hiperespectrales con una gran número de bandas y que estén muy correlacionadas entre sí, Pb puede ser muy elevado ya que posteriormente se podrá reconstruir de nuevo la imagen con buenos resultados, tal y como se observa en el “Capítulo 4: Resultados” de esta memoria.

Por último, hay que tener presente el hecho de que el compresor CDF se sirve del estado del arte en lo que algoritmos de Data Fusion se refiere para realizar la descompresión de las imágenes hiperespectrales con las que se trabaja, de forma que si en el futuro dichos algoritmos mejoran, esto se traducirá a su vez en un avance de los resultados obtenidos con los algoritmos de Data Fusion usados al realizar las simulaciones desarrolladas en este TFM.

Un aspecto importante a tener en cuenta es que con el compresor CDF no se busca mejorar el ratio de compresión conseguido en las imágenes hiperespectrales, sino aumentar la simplicidad de las compresiones a realizar, garantizando que la calidad de los resultados siga siendo adecuada.

2.2 Estrategia de compresión basada en unmixing empleando el algoritmo FUN

La segunda propuesta de compresión de imágenes hiperespectrales que se presenta en este TFM se basa en el uso de una adaptación del algoritmo FUN de unmixing, por lo que se referenciará a dicha propuesta a partir de ahora como ‘compresor CFUN’. Por ello, en primer

lugar se presenta el algoritmo original y posteriormente se indica las modificaciones realizadas para alcanzarlo.

Entre los usos más comunes que se le dan a las imágenes hiperespectrales está el de determinar la composición de materiales presentes en una escena. Para realizar esta labor, se utilizan algoritmos de desmezclado espectral que se basan en el concepto de que cada uno de los píxeles que componen una imagen de este tipo puede ser representado como una combinación lineal de los p píxeles puros, o endmembers, que ésta contiene, dando un parámetro conocido como abundancia la proporción de los diferentes endmembers en cada uno de los píxeles de la imagen.

$$r_i = \sum_{j=1}^p e_j \cdot a_{i,j} + n_i \quad (2.2)$$

Siendo:

e_j : la firma espectral del endmember j

p : número de endmembers de la imagen

$a_{i,j}$: la abundancia del endmember j para el píxel i

n_i : el ruido presente en el píxel i

El proceso de unmixing como se ha llevado a cabo tradicionalmente se divide en tres pasos: estimación del número de endmembers, extracción de los endmembers y cálculo de las abundancias. Cada uno de dichos pasos se desarrolla mediante algoritmos distintos los cuales proporcionan buenos resultados. Sin embargo, al aplicarse estos de forma secuencial, los resultados de unos afectan a los siguientes algoritmos, de forma que un error en el cálculo del número de endmembers se irá arrastrando en todo el proceso de unmixing, lo que se traducirá en una pérdida en la exactitud de los resultados, sin contar con que estos algoritmos suelen realizar operaciones matricialmente complejas como el cálculo de inversas o estimación de autovalores y autovectores.

Con el objetivo de evitar los inconvenientes indicados anteriormente, se ha presentado el algoritmo FUN que estima el número de endmembers y lleva a cabo su extracción de la imagen hiperespectral de forma simultánea, haciendo para ello uso de operaciones matricialmente sencillas. Por otra parte, permite incluso llevar a cabo la obtención de las abundancias realizando unos pocos cálculos adicionales.

Para realizar el proceso de desmezclado espectral, el algoritmo FUN tiene que ser inicializado con un endmember. Para ello se emplea el píxel medio o centroide de la imagen. En primer lugar, el algoritmo FUN obtiene la proyección ortogonal de cada píxel de la imagen respecto del centroide, haciendo uso del método modificado de Gram-Schmidt. Finalmente, el algoritmo FUN selecciona como endmember el píxel con mayor proyección ortogonal respecto al centroide. Una vez que el primer endmember se ha seleccionado, el algoritmo FUN comienza a extraer los endmembers restantes de forma análoga a cómo obtuvo el primero de ellos, con la diferencia de que utiliza el último píxel puro obtenido para realizar la proyección ortogonal, en lugar del píxel medio o centroide.

Cabe destacar que para que un píxel sea considerado como puro es necesario que cumpla con un criterio de selección que permite detener el proceso desde que haya un candidato que no lo cumpla, lo cual indica que ya han sido encontrados todos los endmembers de la imagen. Ésta es la forma en la que el algoritmo consigue estimar tanto el número como cuáles son los endmembers de forma simultánea.

Para decidir si un píxel es o no un endmember, el algoritmo FUN calcula el porcentaje de información que se perdería si éste no fuera considerado como tal, lo cual sirve como criterio de parada para el algoritmo. Si el porcentaje de información es inferior a un determinado valor dado como parámetro de entrada al algoritmo, éste se detiene.

Hasta ahora se ha definido en este apartado el algoritmo FUN en su concepción para realizar desmezclado espectral, pero tal y como se ha indicado, se van a realizar sencillas modificaciones del mismo para que pueda ser aplicado a compresión de imágenes hiperespectrales. Para ello:

- Se modifica el criterio de parada. Tal y como se ha expresado previamente, el criterio de parada en el algoritmo FUN usado en labores de desmezclado espectral se basa en comparar el porcentaje de información que se perdería si no fuera considerado un píxel candidato como endmember. Si este porcentaje es inferior a un determinado valor, dicho píxel no se consideraría como puro y el algoritmo se detendría. Por un lado, el porcentaje de información perdida está directamente relacionado con el Signal-to-noise ratio (SNR) que se obtendrá en la reconstrucción de la imagen, y esto permite que algoritmo acepte como parámetro de entrada el SNR que se desea obtener en la reconstrucción cuando es empleado para

compresión de imágenes hiperespectrales o ultraespectrales. Por otra parte, el número de píxeles seleccionados como endmembers está directamente relacionado con el ratio de compresión alcanzado, con lo que el algoritmo también puede aceptar como parámetro de entrada el ratio de compresión que se desea obtener. De esta forma, el número de píxeles seleccionados como endmembers sería el indicado por la expresión 2.3 que se muestra a continuación:

$$p_{max} = \frac{N \cdot N_b}{rc_{min} \cdot (N + N_b)} \quad (2.3)$$

Siendo:

- p_{max} : el número de píxeles puros a extraer habiendo fijado el ratio de compresión
 - N : Número de píxeles de la imagen
 - N_b : Número de bandas espectrales de la imagen
- Otra modificación importante se encuentra en los parámetros de salida del algoritmo FUN. En su versión original enfocada a procesos de unmixing, la salida del algoritmo proporciona los endmembers de la imagen hiperespectral y las correspondientes abundancias para cada uno de ellos en cada píxel de la imagen. En el caso que ocupa a este TFM centrado en la compresión de imágenes, la salida del compresor CFUN tras la modificación del algoritmo FUN sería una matriz U que contiene a aquellos vectores de proyección calculados durante el desarrollo del algoritmo hasta que éste se detenga por el criterio de parada, y una versión reducida de la imagen hiperespectral original (M_{red}).

El proceso descrito en este apartado se presenta en el pseudocódigo posterior (Algorithm 1), en el cuál se comprueba que la entrada al algoritmo sería la imagen hiperespectral M , junto con el criterio de parada: SNR mínimo (SNR_{min}) y ratio de compresión mínimo (RC_{min}). En la línea 2 se escoge el primer vector de proyección de acuerdo con el criterio seleccionado de mayor proyección ortogonal respecto al centroide de la imagen, y en la 8 se establece el número mínimo de vectores ortogonales a conseguir para alcanzar el ratio de compresión mínimo exigido. En el bucle incluido entre los pasos 12-23 se calculan las proyecciones ortogonales de los píxeles con respecto al último endmember obtenido, y se almacena aquél que presenta la mayor proyección, repitiéndose el proceso hasta que no se

cumpla con el criterio de parada del inicio del bucle (línea 12). Finalmente se obtiene la imagen reducida con una simple multiplicación matricial, línea 24.

Algorithm 1: Hyperspectral compression based on orthogonal vectors

Inputs: $M = [r_1, r_2, \dots, r_{N_p}]$, SNR_{\min} , RC_{\min}

- 1 $X := [x_1, x_2, \dots, x_{N_p}] = M$ {Auxiliary copy of the hyperspectral image}
- 2 $u_1 := x_j / \|x_j\|$ where x_j is the pixel selected as the first projection vector according to the initialization criteria.
- 3 $U := [u_1]$;
- 4 $\text{SNR}_{\text{num}} = 0$, {Quadratic addition of the hyperspectral image};
- 5 **for** $j = 1$ **to** N_p **do**
- 6 | $\text{SNR}_{\text{num}} = \text{SNR}_{\text{num}} + r_j \circ r_j$;
- 7 **end**
- 8 $p_{\max} := (N_p \cdot N_b) / (\text{RC}_{\min} \cdot (N_p + N_b))$, {Maximum number of orthogonal vectors to be extracted};
- 9 $p := 1$; {Number of projection vectors already extracted}
- 10 $\text{exit} := 0$;
- 11 $\text{SNR}_{\text{est}} := 0$, {Estimated SNR};
- 12 **while** $p \leq p_{\max}$ **and** $\text{SNR}_{\text{est}} \leq \text{SNR}_{\min}$ **do**
- 13 | $\text{SNR}_{\text{den}} = 0$;
- 14 | **for** $j = 1$ **to** N_p **do**
- 15 | | $x_j = x_j - (x_j \circ u_p) \cdot u_p$;
- 16 | | $\text{error}_j = x_j \circ x_j$;
- 17 | | $\text{SNR}_{\text{den}} = \text{SNR}_{\text{den}} + \text{error}_j$, {Quadratic addition of the remaining hyperspectral image};
- 18 | | $\text{SNR}_{\text{est}} = 10 \cdot \log_{10}(\text{SNR}_{\text{num}} / \text{SNR}_{\text{den}})$;
- 19 | **end**
- 20 | $j_{\max} := \arg \max(\text{error}_j)$;
- 21 | $u_p := x_{j_{\max}} / \|x_{j_{\max}}\|$;
- 22 | $p := p + 1$;
- 23 **end**
- 24 $M_{\text{red}} = U^T \cdot M$;

Outputs:
 $M_{\text{red}} = [r_{1\text{red}}, r_{2\text{red}}, \dots, r_{N_p\text{red}}]$, {Reduced image};
 $U = [u_1, u_2, \dots, u_p]$, {Projection vectors}

En resumen, haciendo uso del compresor CFUN, que surge de modificar ligeramente el algoritmo FUN original de desmezclado espectral, se conseguiría la compresión de la imagen hiperespectral original obteniendo los vectores de proyección sucesivos que se calculan antes de alcanzar el criterio de parada y ordenándolos en una matriz de proyección, U, así como con el cálculo de la imagen reducida que resulta al multiplicar la traspuesta de dicha matriz U por la que contiene la información de la imagen hiperespectral original, M. Tal y como se muestra en la expresión:

$$M_{\text{red}} = U^T \cdot M \quad (2.4)$$

El proceso de recuperación de la imagen es sencillo, basta con premultiplicar la imagen reducida por la traspuesta de los vectores de proyección:

$$M = U^T \cdot M_{Red} \quad (2.5)$$

Capítulo 3. Sensores hiperspectrales, métricas y estado del arte.

En este capítulo se presenta información general que es necesaria conocer antes de presentar los resultados obtenidos en el conjunto de simulaciones realizadas con las metodologías propuestas. De esta forma, se van a presentar los sensores que han tomado las imágenes hiperspectrales y ultraespectrales que se utilizan en las simulaciones, las métricas que permiten evaluar los resultados y los algoritmos que componen el estado del arte en compresión de imágenes hiperspectrales actualmente, así como los de Data Fusion utilizados para descomprimir las imágenes comprimidas con el compresor CDF.

3.1.- Información general

En este apartado se muestran en primer lugar las características de los sensores hiperespectrales que han captado las imágenes utilizadas para evaluar los métodos de compresión propuestos; en segundo lugar, se explican las métricas usadas para comparar los resultados obtenidos por las diferentes metodologías de compresión; en tercer lugar, se realiza una explicación breve de los algoritmos que componen el estado del arte en compresión de imágenes hiperespectrales y que se han empleado para evaluar los resultados obtenidos con las metodologías propuestas, además, también se presentarán los que conforman el estado del arte de algoritmos de Data Fusion y que se utilizarán para descomprimir las imágenes comprimidas mediante el compresor CDF.

Los algoritmos explicados en la última sección del presente capítulo son: el algoritmo Generalized Gram-Schmidt Adaptive (GGSA), el algoritmo Generalized Local Minimum Square Error (GLMMSE), el algoritmo Computationally Efficient algorithm for Fusing Multispectral and Hyperspectral Images (CoEF – MHI), el algoritmo Principal Component Analysis (PCA), el algoritmo Lossy Compression for Exomars (LCE) y los algoritmos Consultative Committee for Space Data System (CCSDS), tanto su versión CCSDS – 122 como su versión CCSDS – 123.

3.1.1.- Sensores hiperespectrales

3.1.1.1.- Airbone Visible / Infrared Image Spectrometer (AVIRIS)

El sensor hiperespectral AVIRIS trabaja en el rango espectral del infrarrojo cercano (400 – 2500 nm), presentando cada canal un ancho de 10 nm, lo que se traduce en 224 bandas aproximadamente. En función de la altitud a la que se tomen las imágenes hiperespectrales, cada píxel puede cubrir una superficie de 20 m² cuando se captan a altas altitudes o de 4 m² cuando la imagen se toma desde alturas más cercanas al nivel del mar.

La imagen que se ha utilizado para realizar las simulaciones tiene unas dimensiones de 512x677x224 (píxeles x píxeles x bandas), siendo la Figura 7 una representación en falso color de la imagen utilizada:

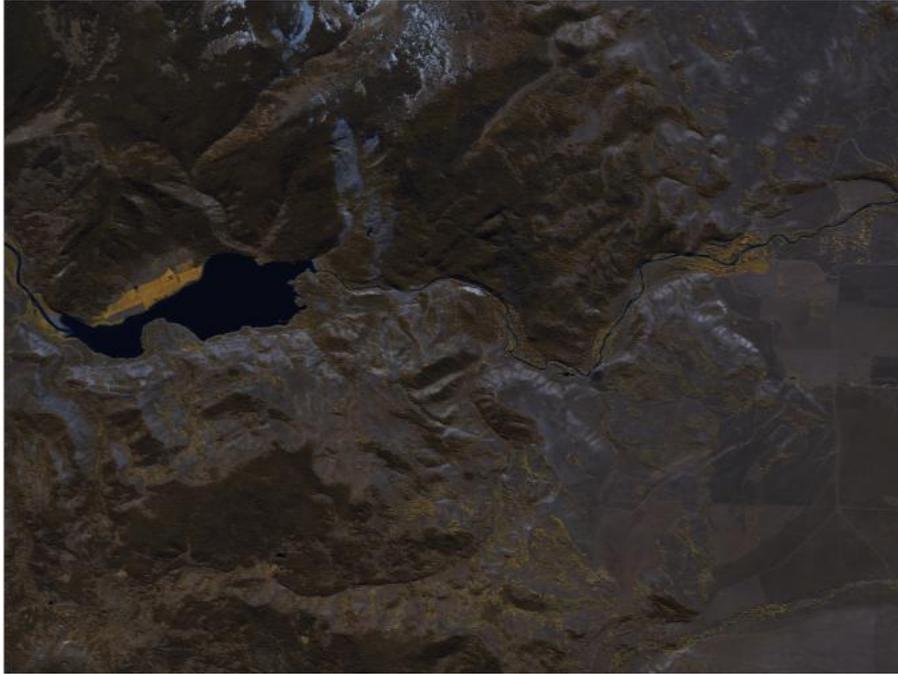


Figura 7. Imagen en falso color de la escena captada por el sensor AVIRIS

3.1.1.2.- Compact Reconnaissance Imaging Spectrometer for Mars (CRISM)

Éste es un sensor que trabaja tanto en el infrarrojo cercano (362 – 1053 nm) como en el infrarrojo (1002-3920 nm). Presenta un total de 545 bandas con una resolución espacial de 18.4 m² por píxel de la imagen a 300 km de altitud.

La imagen utilizada para realizar las simulaciones tiene unas dimensiones de 480x320x545 (píxeles x píxeles x bandas) siendo la representación de tres bandas de la imagen hiperespectral captada por el sensor la que se presenta en la Figura 8:



Figura 8. Representación en falso color de la imagen obtenida por el sensor CRISM

3.1.1.3.- Atmospheric Infrared Sounder (AIRS) e Infrared Atmospheric Sounding Interferometer (IASI)

Estos dos sensores, el AIRS y el IASI, trabajan en las bandas del infrarrojo térmico. En ambos casos, el rango espectral que son capaces de captar van desde los $3.62 \mu\text{m}$ hasta los $15.5 \mu\text{m}$, con la diferencia de que el AIRS tiene solamente 2378 canales los cuales distribuye por todo el rango, mientras que el IASI es capaz de tomar información espectral haciendo uso de 8461 bandas y a lo largo de todo el espectro dentro del rango de frecuencias sobre el que actúa, hecho que se puede observar en la Figura 9:

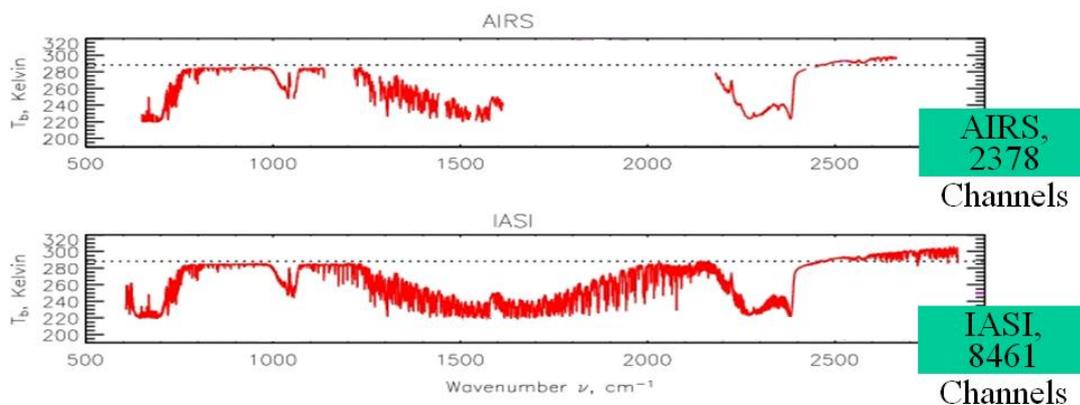


Figura 9. Comparativa del espectro obtenido por los sensores AIRS e IASI

La imagen procedente del sensor AIRS que se ha empleado para realizar las simulaciones tiene una dimensión de 135x90x1501 (píxeles x píxeles x bandas) siendo la Figura 10 una representación en falso color de la misma. Análogamente se tiene que la Figura 11 corresponde con una imagen captada por el IASI, teniendo la empleada en las simulaciones unas dimensiones de 60x66x8461 (píxeles x píxeles x bandas).



Figura 10. Imagen en falso color de la escena tomada por el sensor AIRS

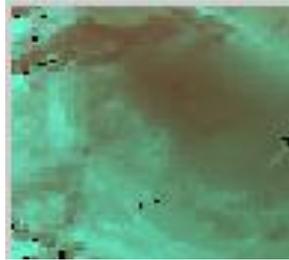


Figura 11. Representación en falso color de la imagen conseguida por el sensor IASI

3.1.2.- Métricas utilizadas

Para la evaluación de la calidad del proceso de compresión y descompresión se ha empleado las métricas Signal-to-Noise Ratio (SNR) y Spectral Angle Mapper (SAM).

- Signal-to-noise Ratio (SNR): Es una medida de la relación entre la información y el ruido presente en la imagen reconstruida (ImR) respecto de la imagen original (ImO). Entre mayor sea el valor de esta métrica, mejor es el resultado obtenido:

$$SNR = 20 \log_{10} \frac{\mu_{ImO}}{MSE}, MSE = \frac{1}{N_b \cdot N_p} \sum_{i=1}^{N_b} \sum_{j=1}^{N_p} (ImR - ImO)_{i,j}^2 \quad (3.1)$$

- Spectral Angle Mapper (SAM): Indica una medida de la calidad espectral de la imagen resultante (ImR) respecto de la imagen original (ImO), siendo el valor nulo el ideal para esta métrica, lo que indicaría que la información obtenida al descomprimir la imagen es exactamente igual a la que se tenía antes de iniciar el proceso.

$$SAM(ImR, ImO) = \frac{1}{N_p} \cdot \sum_{j=1}^{N_b} SA(ImR_j, ImO_j), \text{ para } SA(ImR_j, ImO_j) = \arccos \frac{ImR_j \cdot ImO_j}{\|ImR_j\| \cdot \|ImO_j\|} \quad (3.2)$$

Donde ImR_j y ImO_j se refieren al píxel j -ésimo de la imagen fusionada y de la hiperespectral original respectivamente; SA representa el ángulo espectral entre los píxeles hiperespectrales, representando $ImR_j \cdot ImO_j$ el producto escalar de esos dos vectores y $\| \cdot \|$ su correspondiente norma l_2 .

3.1.3.- Algoritmos de referencia

3.1.3.1.- Computationally Efficient algorithm for Fusing Multispectral and Hyperspectral Images (CoEF-MHI).

El CoEF-MHI es un algoritmo que permite obtener una imagen tras fusionar dos independientes: una multiespectral y otra hiperespectral de una misma escena, haciendo uso de un proceso que requiere de una baja carga computacional y obteniendo como resultado una imagen de gran calidad. De este modo, utilizando la información espacial que proporciona la imagen multiespectral de baja resolución espectral y alta espacial (HMSI, High resolution MultiSpectral Image), se obtiene partiendo de la imagen hiperespectral de baja resolución espacial y alta espectral (LHSI, Low resolution HyperSpectral Image), una nueva imagen hiperespectral con alta resolución espacial (HHSI, High resolution HyperSpectral Image), llegando a conseguir de esta forma una imagen que mejora a las dos originales de las que proceden.

Para ello, este algoritmo primero amplía espacialmente la imagen LHSI a partir de una interpolación bilineal hasta alcanzar el tamaño en píxeles de la HMSI. Posteriormente, mejora cada píxel de la imagen resultante combinándolos linealmente con sus vecinos más próximos de forma que alcance una calidad espacial similar a la multiespectral inicial.

3.1.3.2.- Generalized Gram-Schmidt Adaptive (GGSA) y Generalized Local Minimum Square Error (GLMMSE).

Los algoritmos GGSA y el GLMMSE son versiones generalizadas del Gram-Schmidt Adaptive (GSA) y del Local Minimum Square Error (LMMSE) respectivamente. Dichos algoritmos originales desarrollan la fusión de imágenes entre una imagen pancromática de alta resolución espacial y una multiespectral de baja resolución de este tipo.

Los algoritmos GGSA y GLMMSE dividen, en primer lugar, el espectro recogido por la imagen hiperespectral en varias regiones. De esta forma, se relaciona cada banda de la imagen multiespectral con una de dichas regiones y se producen artificialmente otras bandas para cubrir aquéllas que queden sin representación, haciendo uso del proceso Ratio Image-Based Spectral Resampling (RIBSR).

Tras realizar la relación entre bandas y regiones, la fusión se desarrolla de forma independiente por cada grupo haciendo uso de los algoritmos GSA o LMMSE.

3.1.3.3.- Principal Component Analysis (PCA)

El PCA [7] es un procedimiento matemático que facilita la simplificación de grandes conjuntos de datos, transformando un determinado número de variables correlacionadas en un número más pequeño y que no presentan correlación entre ellas, conocidas como componentes principales. La principal aplicación de este procedimiento matemático en remote sensing es la compresión de imágenes hiperespectrales, lo cual se realiza reduciendo la dimensionalidad del conjunto de datos de la imagen. Para ello se ha de encontrar un nuevo conjunto de ejes ortogonales que tendrán como origen el centro del conjunto de datos y se rotará de forma que la nueva disposición presente una variabilidad espectral máxima (máxima varianza). De esta forma, en este nuevo espacio, la matriz de covarianza solamente tendrá valores no nulos en la diagonal principal.

El PCA se apoya en el hecho de que las bandas próximas de imágenes hiperespectrales están altamente correlacionadas, y en general expresan prácticamente la misma información. Este método transforma los datos originales de forma que elimina la correlación entre las bandas. Para ello, se hace uso del principio matemático que descompone por autovalores a la matriz de covarianza de las bandas de la imagen hiperespectral. De esta forma, la compresión se habría realizado en el dominio espectral de la imagen, si bien la resolución espacial seguiría siendo la misma. Por este motivo, y debido a que las bandas de la imagen comprimida con el

PCA son ortogonales entre sí, puede llevarse a cabo posteriormente una compresión espacial de las mismas, empleando algoritmos como CCSDS-122 (lossless o lossy) o el JPEG. Si bien este proceso lleva a la obtención de muy buenos resultados, la complejidad del mismo hace que no sea posible su aplicación para compresión de imágenes hiperespectrales y ultraespectrales en satélites. El empleo del algoritmo PCA en las simulaciones se debe a que, la cadena compuesta por los algoritmos PCA y CCSDS-122 o JPEG podría ser sustituida por la cadena compuesta por los algoritmos FUN y CCSDS-122 o JPEG, ya que al igual que el algoritmo PCA, el algoritmo FUN proporciona una imagen comprimida espectralmente con bandas ortogonales entre sí.

3.1.3.4.- Lossy Compression for Exomars (LCE)

El algoritmo LCE [8] fue concebido con el fin de satisfacer la necesidad de realizar la compresión de datos durante la misión Exomars. Tiene su procedencia en un algoritmo lossless que fue resultado de un proyecto anterior de la ESA, el cual ha derivado en este LCE tras realizar una serie de cambios como pasar a ser un algoritmo con pérdidas, lossy. Ello ha llevado a que se alcance una gran calidad de resultados con ratios de compresión importantes, y al tener en consideración durante su desarrollo el que iba a trabajar en una misión espacial, fue diseñado para tener una alta eficiencia de compresión a la vez que cumplía con los requerimientos propios de la compresión on-board al mismo tiempo:

- Baja complejidad: El algoritmo está concebido para que sea ejecutado en una implementación física, por lo que se ha conseguido que necesite pocos recursos hardware para funcionar de forma eficiente, incrementando así la cantidad de información que es comprimida por unidad de tiempo. Esto da la posibilidad de obtener una implementación hardware que pueda operar en tiempo real.
- Resistencia a errores. Para evitar que haya errores provocados por efectos de la radiación o por el propio proceso de transmisión de datos, los cuales harían que se perdiera gran cantidad de información al no poder descomprimir los datos posteriores al error, el algoritmo LCE opera bajo bloques independientes de $N \times N$ muestras de píxeles con todas sus correspondientes bandas, de forma que si se produjera un error durante la compresión, éste solo afectaría al bloque en el que éste se encuentra.
- Hardware-friendliness. Debido a su baja complejidad, a que opera usando aritmética entera y a que puede ser paralelizable, es un algoritmo muy bueno a la hora de plantear una implementación en una ASIC o en una FPGA.

3.1.3.5.- Consultative Committee for Space Data System (CCSDS)

El CCSDS [9] es la unión de las agencias espaciales más importantes del mundo, las cuales han desarrollado tres estándares para compresión de datos en el espacio: un compresor sin pérdidas de carácter universal (CCSDS-121), un estándar de compresión con o sin pérdidas de información para imágenes bidimensionales a escala de grises basado en la transformada Wavelet (CCSDS-122), y un algoritmo para la compresión de imágenes multiespectrales e hiperspectrales lossless (CCSDS-123).

Al término de la exposición de resultados, se va a hacer uso tanto del CCSDS-122 como del CCSDS-123, ambos en configuración lossless, con el fin de probar qué mejoras se podrían llegar a producir en lo referente a los ratios de compresión conseguidos. El CCSDS-122 se aplicará a los resultados obtenidos con compresor CFUN, mientras que, por su parte, el CCSDS-123 a las imágenes hiperspectrales y multiespectrales que resultan tras aplicar el compresor CDF.

Capítulo 4. Resultados

En este capítulo se exponen los resultados globales conseguidos en las simulaciones realizadas para comprobar la calidad de los compresores propuestos, comparándolos para poder evaluarlos con el estado del arte en lo que se refiere a compresión de imágenes hiperespectrales. A su vez, al término del capítulo se explicarán características propias del compresor CFUN haciendo uso de los resultados para apoyar esas afirmaciones.

4.1.- Planteamiento de las simulaciones

Los resultados obtenidos se van a exponer en cuatro bloques diferentes, los cuales se dividen en función de la imagen hiperespectral estudiada. De esta forma, se tendrá un bloque dedicado a la imagen captada por el sensor AIRS, otro bloque para la del sensor AVIRIS, y otros dos para las imágenes del CRISM y del IASI.

Dentro de cada uno de los bloques, se van a presentar los resultados conseguidos con las diferentes pruebas realizadas. Así se tendrán:

- Estudio de la compresión usando el compresor CFUN:
 - o Fijando un ratio de compresión objetivo
 - Aplicando el algoritmo a la imagen completa
 - Aplicando el algoritmo a secciones de las imágenes de 25x25 píxeles, 50x50 píxeles y 100x100 píxeles, de forma independiente.
 - o Fijando un SNR objetivo. Análogamente al caso anterior:
 - Se aplicará el algoritmo a la imagen completa
 - Se aplicará el algoritmo a secciones de las imágenes de 25x25 píxeles, 50x50 píxeles y 100x100 píxeles, de forma independiente.
- Estudio de la compresión usando el compresor CDF y recuperando la imagen por medio de los siguientes algoritmos de Data Fusión:
 - o GGSA
 - o GLMMSE
 - o CoEF – MHI
- Comparación de los mejores resultados globales con los que proporcionan los algoritmos que componen el estado del arte en lo que se refiere a compresión de imágenes hiperespectrales (PCA y LCE).

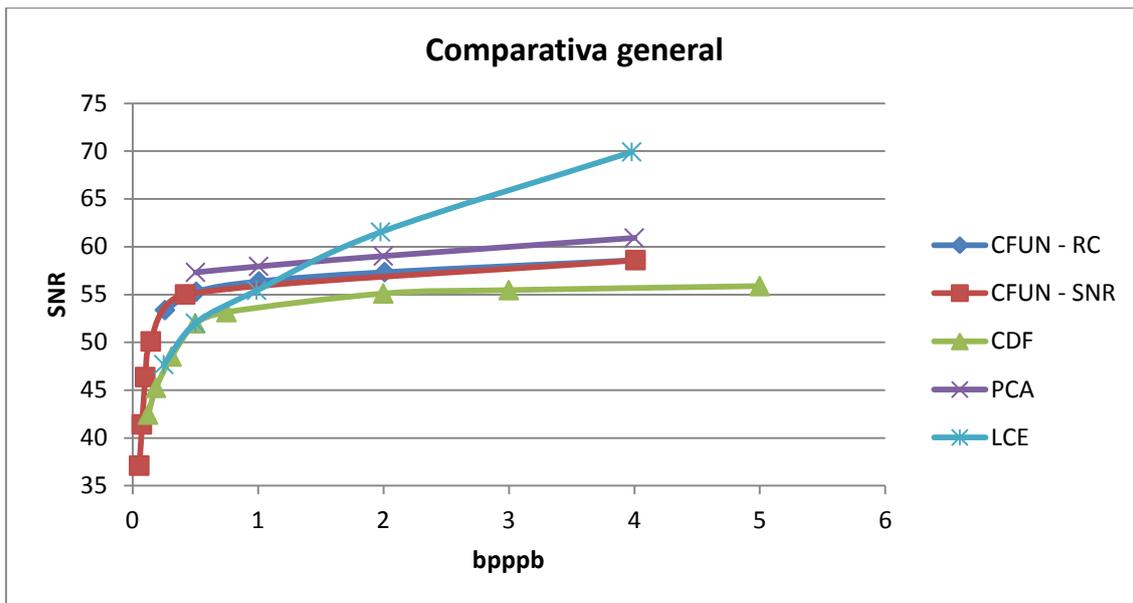
Una vez conseguidos los resultados anteriores, se van a seleccionar aquellos casos de las metodologías propuestas que otorguen mejores resultados con el objetivo de comprimirlos aún más, haciendo uso para ello de dos algoritmos en su configuración sin pérdidas: el CCSDS 122 y el CCSDS 123, los cuales generarán la misma calidad de resultados en lo que se refiere a las métricas SNR y SAM, a la vez que intentan conseguir una mejora adicional en los ratios de compresión.

Se han llevado a cabo simulaciones empleando distintas configuraciones para cada uno de los dos métodos de compresión propuestos. Las simulaciones realizadas sobre las diferentes imágenes hiperespectrales reales conseguidas gracias a los distintos sensores hiperespectrales indicados anteriormente, dieron lugar a un conjunto de datos a partir de los cuales se realizaron las gráficas recogidas en los distintos anexos de esta memoria. Tras comparar las diferentes configuraciones de los métodos propuestos, se ha seleccionado para cada caso la configuración que proporciona mejores resultados, y éstos se han comparado con los resultados proporcionados por los algoritmos PCA y LCE, dando lugar a las gráficas resumen que se van a presentar en esta sección de resultados. De esta manera se expondrán dos gráficas por cada imagen que se corresponderán con la representación de las métricas escogidas (SNR y SAM) respecto al número de bit por píxel y por banda (bpppb) que tiene cada una de las imágenes comprimidas, parámetro que resulta de dividir el número de bits que utiliza la imagen hiperespectral original, antes de la compresión, para almacenar la información de cada una de las bandas de cada píxel que compone la imagen por el ratio de compresión alcanzado durante el proceso.

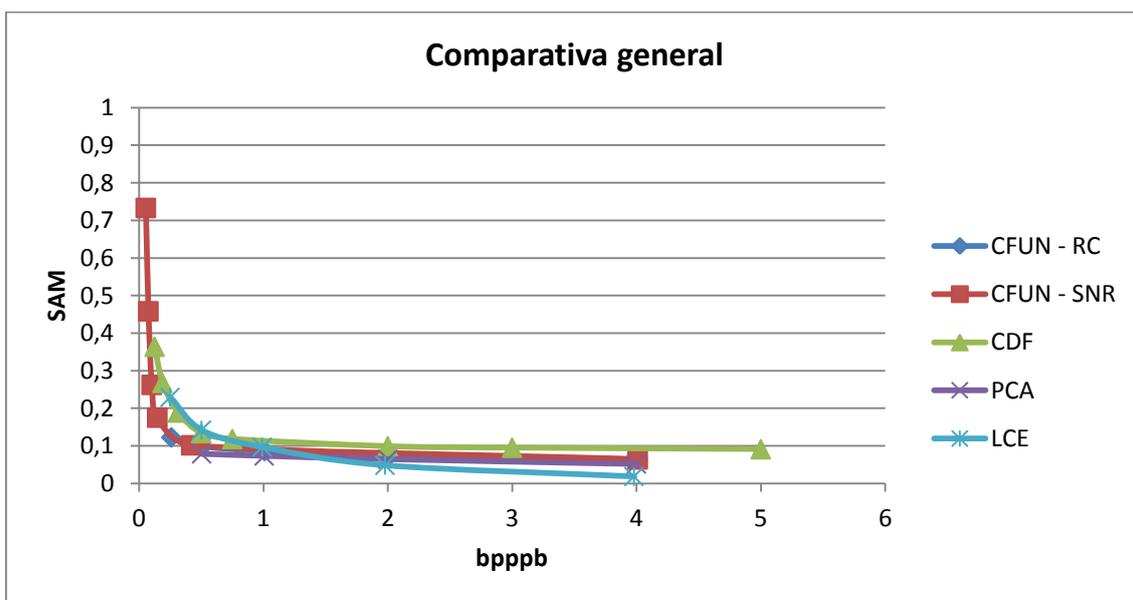
Cabe destacar que si bien se ha trabajado con dos métricas diferentes, las conclusiones que proporcionan ambas son equivalentes, por lo que se explicarán los resultados obtenidos haciendo referencia solamente a una de ellas, el SNR, al verse los resultados con mayor claridad. Por otro lado, en la imagen conseguida por el sensor IASI, hay varios píxeles saturados que no permiten calcular el SAM de las simulaciones sin alterar los resultados. Como consecuencia de ello, en estos casos solamente resulta una única gráfica, la correspondiente al SNR.

4.1.1.- Resultados obtenidos con la imagen del sensor AIRS

De acuerdo con los resultados mostrados en las *Gráficas 1 y 2*, se puede extraer como conclusión que en el caso de que los ratios de compresión aplicados a la imagen sean bajos, los cuales provocan que la información ocupe más de 1,5 bpppb, el algoritmo que mejor resultado proporciona es el LCE. Por el contrario, éste pasa a descomprimir las imágenes con menos exactitud que otros de los algoritmos testados cuando se utilizan altos ratios de compresión. Así, para ratios de compresión elevados, bpppb menor de 1.5, el algoritmo que mejores resultados proporciona es el PCA. Puede observarse también que para altos ratios de compresión el algoritmo FUN consigue claramente mejores resultados que el LCE, y que el compresor basado en Data Fusión genera resultados similares al LCE.



Gráfica 1 – Comparativa general de los SNR de las simulaciones usando la imagen del sensor AIRS

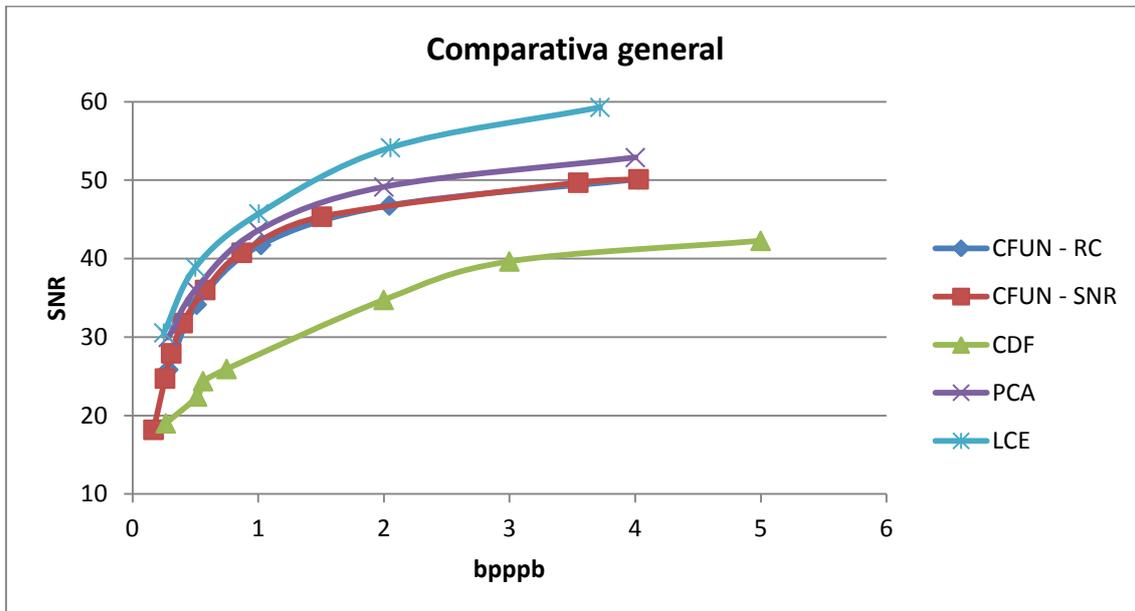


Gráfica 2 – Comparativa general de los SAM de las simulaciones usando la imagen del sensor AIRS

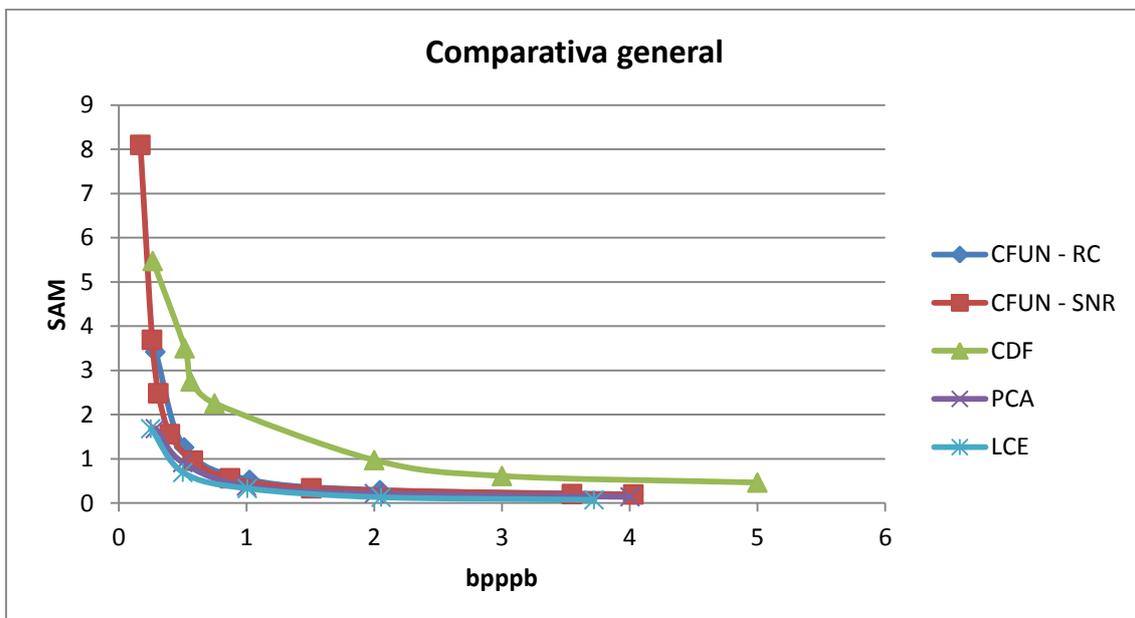
4.1.2.- Resultados obtenidos con la imagen del sensor AVIRIS

En el estudio de la imagen del AVIRIS, atendiendo a las Gráficas 3 y 4, se comprueba que al contrario de lo que ocurría con la imagen del AIRS, sea cual sea el ratio de compresión adquirido el que mejor resultado proporciona es el algoritmo LCE seguido del algoritmo PCA.

No obstante, éste prácticamente presenta los mismos resultados que los conseguidos haciendo uso del algoritmo FUN cuando se trabaja con ratios de compresión de 16 aproximadamente, lo que se traduce en 1 bpppb en la imagen comprimida. Por otro lado, se observa que el compresor CDF que descomprime la imagen con algoritmos de Data Fusion, en este caso el CoEF – MHI, es con el que se consigue peor calidad en los resultados.



Gráfica 3 – Comparativa general de los SNR de las simulaciones usando la imagen del sensor AVIRIS

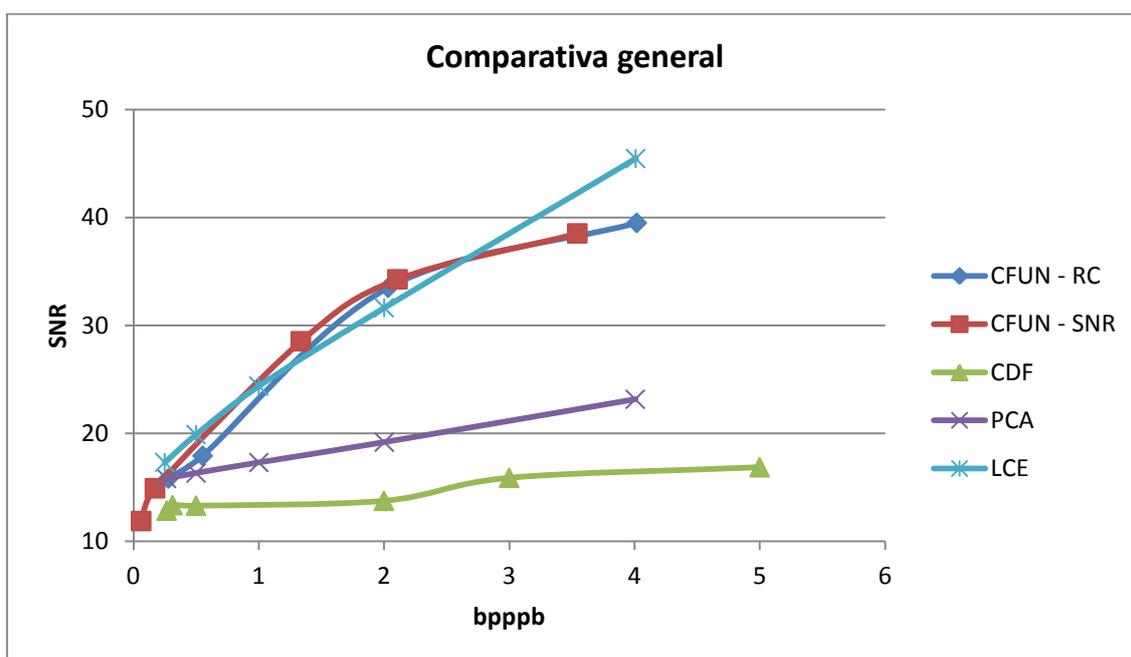


Gráfica 4 – Comparativa general de los SAM de las simulaciones usando la imagen del sensor AVIRIS

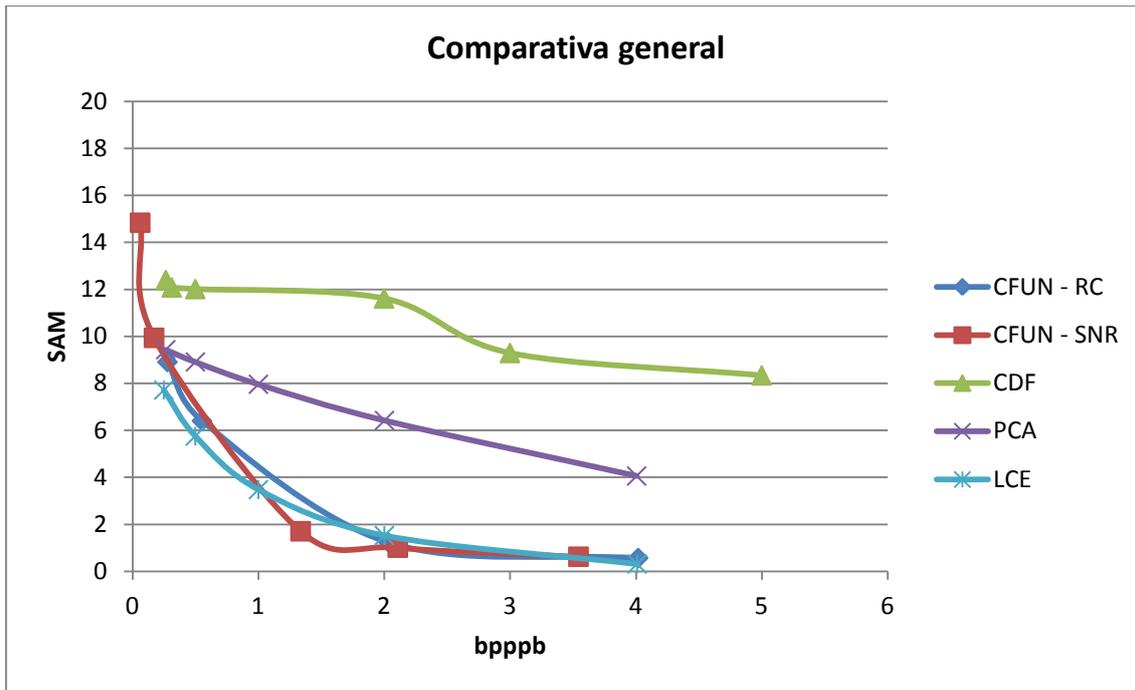
4.1.3.- Resultados obtenidos con la imagen del sensor CRISM

La imagen captada por el sensor CRISM tiene la peculiaridad de poseer una gran cantidad de ruido. Como consecuencia, si se analizan los resultados conseguidos por todos los algoritmos (ya sean los propuestos o los que compongan el estado del arte), éstos no llegan a tener valores que destaquen por altos SNR o bajos SAM, valores deseables en dichas métricas.

Sabiendo pues que esta imagen contiene mucho ruido, ha de dársele mayor importancia a los resultados conseguidos por la nueva metodología presentada basada en unmixing. Esto es así ya que si bien es el LCE el que otorga mejores resultados, el algoritmo FUN consigue unos bastante destacables, llegando incluso a mejorar los del LCE para compresiones que consigan que la imagen presente entre 1 y 2,5 bpppb. Otro punto destacable es el hecho de que esta metodología propuesta supera en mucho a los resultados proporcionados por el algoritmo PCA, si bien siguen siendo mejores los de éste último que los conseguidos con el compresor CDF y que utiliza el estado del arte de Data Fusion para la descompresión de la información.



Gráfica 5 – Comparativa general de los SNR de las simulaciones usando la imagen del sensor CRISM



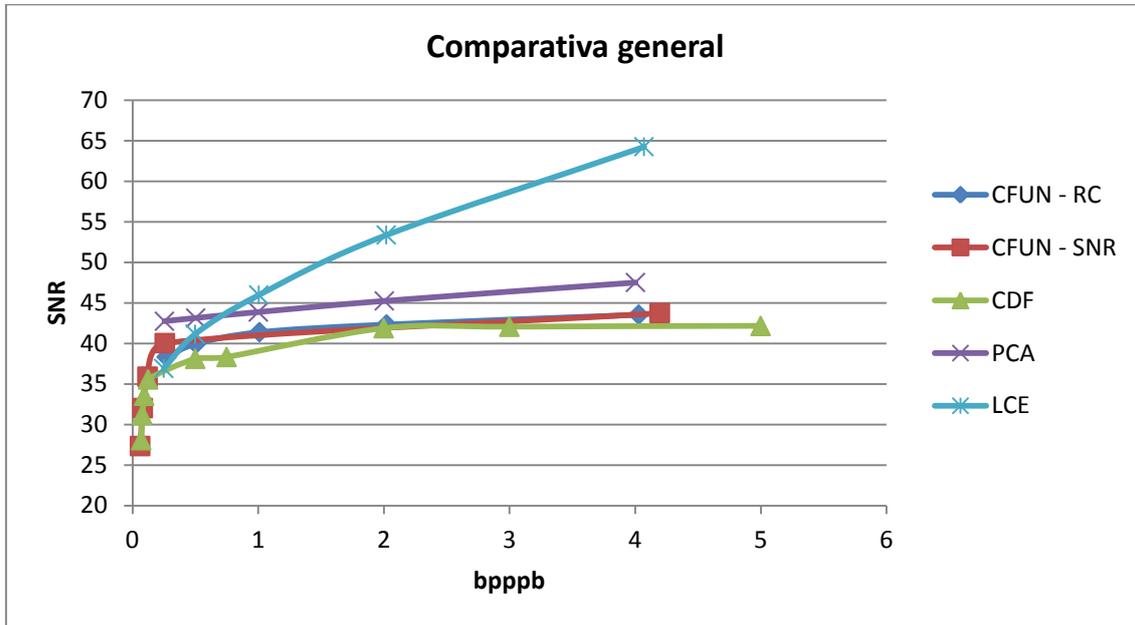
Gráfica 6 – Comparativa general de los SAM de las simulaciones usando la imagen del sensor CRISM

4.1.4.- Resultados obtenidos con la imagen del sensor IASI

Las conclusiones que se sacan tras analizar los resultados conseguidos de las simulaciones realizadas haciendo uso de la imagen del sensor IASI, son equivalentes a las vistas para la imagen del sensor AIRS. Es importante destacar que como se vio anteriormente, ambos sensores presentan características similares en lo que respecta al rango del espectro en el que trabajan, o el elevado número de bandas espectrales que son capaces de captar, y como consecuencia, parece lógico que se consigan resultados análogos.

De esta manera, se puede ver en la Gráfica 7 que el algoritmo que proporciona mejores resultados tras descomprimir la imagen es el algoritmo LCE para ratios de compresión peores de 0,75 bpppb. En ese rango de compresión, éste es el que claramente obtiene un SNR más elevado, seguido del PCA. Por otro lado, de las metodologías propuestas, es en esta imagen en donde la calidad de la compresión más se asemeja entre las conseguidas por el algoritmo FUN modificado y el compresor propuesto, si bien siguen siendo los resultados obtenidos mediante la descompresión de la imagen con Data Fusion los que peor resultados generan. Es importante mencionar que para elevados ratios de compresión (menos de 0,5 bpppb), los resultados obtenidos con el algoritmo de unmixing superan a los proporcionados

por el algoritmo LCE, si bien sigue siendo el algoritmo PCA el que proporciona los mejores resultados.



Gráfica 7 – Comparativa general de los SNR de las simulaciones usando la imagen del sensor IASI

4.2.- Resumen de resultados

Como se ha podido ver en todas las gráficas analizadas anteriormente, las simulaciones hechas haciendo uso del compresor CFUN generan prácticamente el mismo resultado independientemente de cuál sea el criterio de parada seleccionado, si fijando el SNR o el ratio de compresión objetivo. Esto es lógico teniendo en cuenta que con el criterio de parada solamente se actúa sobre la mayor o menor compresión que alcanza la imagen, pero al usar ambos el mismo algoritmo los resultados son equivalentes.

Un apunte importante a tener en consideración con respecto a los resultados obtenidos a través del compresor propuesto es que, como ya se ha dicho con anterioridad, estos van en función del algoritmo de Data Fusion que se use para descomprimir la imagen. Esto se puede apreciar en las diferentes gráficas recogidas entre los anexos I y IV, de donde se extrae que si bien actualmente el que mejor resultado proporciona es el algoritmo CoEF-MHI en términos generales, en un futuro, con nuevos algoritmos de Data Fusion que mejoren los resultados logrados por él en esta aplicación, se logrará de forma pasiva una mejora de los resultados presentados

en este TFM. Existen además en la actualidad otros algoritmos de Data Fusion que proporcionan resultados muy competentes, como son los algoritmos: Coupled Nonnegative Matrix Factorization (CNMF), Hyperspectral Image Superresolution (HySURE), Bayesian Sparse y Bayesian Naive, que no se han probado en este TFM y que tal vez proporcionen mejores resultados.

Por otra parte hay que tener presente que, si bien en la mayor parte de los casos estudiados, tanto el algoritmo LCE como el algoritmo PCA proporcionan los mejores resultados, el objetivo de este TFM es el de presentar nuevas metodologías de compresión de imágenes que diesen también buenos resultados y que a la vez tuvieran grandes ventajas en comparación a los primeros, como por ejemplo su elevada sencillez de ejecución, su alta capacidad de paralelización, su gran facilidad de uso (algoritmo FUN), la posibilidad de ser integrado en hardware, entre otras.

4.3.- Compresión adicional mediante algoritmos CCSDS

Para completar el presente apartado, tal y como se expresó en la introducción del mismo, se han tratado de mejorar aún más los ratios de compresión logrados haciendo pasar una vez que se han comprimido esas imágenes, los productos de dicha compresión por dos diferentes standards de compresión: el CCSDS-122 y el CCSDS-123.

4.3.1.- CCSDS-122

El CCSDS-122 se ha utilizado con los resultados proporcionados por el compresor CFUN. Éste es un algoritmo que solamente comprime imágenes bidimensionales, por lo que no se le puede entregar una imagen hiperespectral o ultraspectral para que la comprima. Como se explicó en el capítulo 2 en donde se propuso el compresor CFUN, al trabajar con él, de la imagen hiperespectral se obtienen dos matrices: la U de vectores de proyección y la M_{red} que presenta la información de la imagen reducida, que resulta de premultiplicar a la hiperespectral original por la matriz traspuesta de U , expresión 2.4. Las bandas de la imagen M_{red} son ortogonales entre sí, es decir, sin redundancias espectrales, y cabe destacar que esta imagen no ha sido comprimida espacialmente. Por este motivo, se pretende comprimir espacialmente, de forma independiente, cada una de las bandas de la imagen M_{red} usando para ello el algoritmo compresor CCSDS-122

El ratio de compresión total que se consigue tras utilizar el compresor CCSDS – 122 sigue la siguiente expresión:

$$r_{CCSDS-122} = \frac{N_U + r_{C_{M_{red}}} \cdot N_{M_{red}}}{N_U + N_{M_{red}}} \quad (4.1)$$

Siendo:

N_U : Tamaño en MBs de la matriz U antes de la compresión por el CCSDS – 122

$r_{C_{M_{red}}}$: Ratio de compresión alcanzado por la imagen reducida con el CCSDS - 122

$N_{M_{red}}$: Tamaño en MBs de la imagen reducida previo a pasar por el CCSDS – 122

El ratio de compresión total alcanzado tras comprimir la imagen hiperespectral con el compresor CFUN más el CCSDS – 122 se calcula multiplicando el ratio conseguido en ambas etapas. Así:

$$r_{C_{Total}} = r_{C_{FUN}} \cdot r_{C_{CCSDS-122}} \quad (4.2)$$

Siendo:

$r_{C_{FUN}}$: el ratio de compresión que se alcanza con el compresor CFUN

Los resultados alcanzados se exponen en las gráficas incluidas dentro del anexo V. En ellas se puede comprobar que en general, para todas las pruebas realizadas, el $r_{C_{Total}}$ alcanzado aumenta tras pasar la imagen comprimida por el CCSDS-122 salvo en dos casos particulares: al tratar de comprimir la M_{red} lograda por el algoritmo FUN adaptado para las imágenes del AIRS y del IASI fijando el ratio de compresión como criterio de parada. De forma que en la primera de ellas, el $r_{C_{Total}}$ solamente empeora cuando los $r_{C_{FUN}}$ son los más bajos de los probados, es decir, para valores de $r_{C_{FUN}}$ de 4, 8 y 16, mientras que en el caso de la imagen del IASI, el $r_{C_{Total}}$ empeora de manera notable para cualquier $r_{C_{FUN}}$ fijado. Teniendo en cuenta que el compresor CCSDS – 122 necesita de redundancia espacial para comprimir, parece ser que la posible redundancia que presenta esta imagen se elimina en parte al comprimirla con el compresor CFUN, de forma que cuando se hacen pasar las imágenes por el CCSDS – 122, el ratio de compresión disminuye. Si bien, el motivo principal para ello puede ser el hecho de que se esté utilizando el CCSDS-122 con su configuración lossless, lo cual puede llevar en ciertos casos a

ratios de compresión inferiores a la unidad, de forma que en lugar de comprimir, en estos casos aumenta la información a almacenar.

Cabe destacar que si bien analizando los datos no parece lógico que para las mismas imágenes y el mismo compresor (CFUN) se consigan resultados diferentes según sea el criterio de parada establecido (como se ha concluido anteriormente), el motivo de ello se centra en que al establecer un SNR objetivo, el algoritmo se mueve en unos ratios de compresión diferentes a los fijados como criterio de parada (4, 8, 16, 32 y 64), eso hace que este caso se pueda dar.

No obstante, viendo que las compresiones medias obtenidas con el algoritmo CCSDS – 122 no proporcionan una mejora significativa (tabla 1) pero sí complica el proceso global de compresión habría que desechar la idea de introducir este paso final, al menos en su configuración lossless. Sería interesante como línea futura de investigación probar ésta metodología con la configuración lossy del compresor CCSDS-122, así como con el compresor JPEG.

	$r_{CCSDS-122(lossless)}$ (AIRS)	$r_{CCSDS-122(lossless)}$ (AVIRIS)	$r_{CCSDS-122(lossless)}$ (CRISM)	$r_{CCSDS-122(lossless)}$ (IASI)
Algoritmo FUN (rc)	0.964	1.17	1.31	0.87
Algoritmo FUN (SNR)	1.07	1.18	1.25	1.44

Tabla 1. Ratios de compresión medios obtenidos por el CCSDS – 122

4.3.2.- CCSDS-123

Si bien el algoritmo CCSDS – 122 solamente es aplicable a imágenes bidimensionales, el algoritmo CCSDS – 123 está diseñado para imágenes multiespectrales e hiperspectrales, por lo que se adapta perfectamente a ser utilizado con la salida del compresor CDF, debido a que como ya se ha indicado, entra a él una imagen hiperspectral para tener como resultado una hiperspectral comprimida espacialmente y una imagen multiespectral comprimida espectralmente. De esta forma, el algoritmo CCSDS – 123 se puede aplicar a las dos imágenes de salida del compresor, logrando de esta manera un mayor ratio de compresión que el que se tenía originalmente.

Si bien el algoritmo CCSDS – 123 es un algoritmo lossless, al contrario que el algoritmo CCSDS – 122, aprovecha la redundancia espectral y es por eso que sí proporciona mejoras

importantes en lo que se refiere a ampliar el ratio de compresión total de la imagen. El ratio de compresión total en este caso es:

$$r_{c_{Total}} = r_{c_{DF}} \cdot r_{c_{CCSDS-123}} \quad (4.3)$$

Siendo:

$r_{c_{DF}}$: El ratio de compresión conseguido con el compresor CDF

$r_{c_{CCSDS-123}}$: La compresión proporcionada por el algoritmo CCSDS – 123

El cálculo del $r_{c_{CCSDS-123}}$ depende del tamaño de ambas imágenes y del ratio de compresión conseguido al aplicar el algoritmo CCSDS-123 tanto a la imagen multiespectral como a la hiperespectral. De esta forma, se llega a la expresión 4.4:

$$r_{c_{CCSDS-123}} = \frac{r_{c_{Hy}} \cdot N_{Hy} + r_{c_M} \cdot N_m}{N_{Hy} + N_M} \quad (4.4)$$

Siendo:

$r_{c_{Hy}}$: Ratio de compresión conseguido para la imagen hiperespectral.

N_{Hy} : Tamaño en MBs de la imagen hiperespectral antes de la compresión por el algoritmo CCSDS – 123

r_{c_M} : Ratio de compresión alcanzado por la imagen multiespectral

N_M : Tamaño en MBs de la imagen multiespectral antes de ser comprimida por el algoritmo CCSDS – 123

Al contrario de lo que se pudo apreciar con el algoritmo CCSDS – 122, el ratio de compresión logrado con el algoritmo CCSDS – 123 ($r_{c_{CCSDS-123}}$) es considerablemente elevado, lo que se traduce en que aumenta en mucho el ratio total conseguido ($r_{c_{Total}}$). Los valores medios de $r_{c_{CCSDS-123}}$ alcanzados tras la aplicación del algoritmo se recogen en la tabla 2:

Compresor	$r_{c_{CCSDS-123(lossless)}}$ (AIRS)			$r_{c_{CCSDS-123(lossless)}}$ (AVIRIS)			$r_{c_{CCSDS-123(lossless)}}$ (CRISM)			$r_{c_{CCSDS-123(lossless)}}$ (IASI)		
	Mín.	Medio	Máx.	Mín.	Medio	Máx.	Mín.	Medio	Máx.	Mín.	Medio	Máx.
	2,78	3,87	4,95	2,53	3,42	4,31	2,77	2,97	3,18	2,84	3,43	4,09

Tabla 2. Ratios de compresión mínimos, medios y máximos obtenidos por el CCSDS – 123

En el Anexo VI de esta memoria se encuentran las gráficas que resumen los resultados alcanzados aplicando, tras el compresor propuesto, el algoritmo CCSDS – 123. Tal y como se aprecia en la tabla 2, no se obtienen en ningún momento ratios inferiores a la unidad, por lo que existe siempre compresión en las imágenes. De forma que se consiguen por lo tanto los mismos resultados en lo referente a calidad de la imagen descomprimida (SNR y SAM) al ser el algoritmo utilizado lossless, pero con una cantidad de bpppb muy inferior, hecho que se aprecia en las gráficas obtenidas para las simulaciones realizadas de todas las imágenes (anexo VI).

Observando dichas gráficas, se puede extraer como conclusión que en general se obtienen mejoras de ratios de compresión. Sin embargo, cuando estos ya son lo suficientemente elevados solamente con el compresor propuesto, no se consigue una ventaja tan importante al aplicar el algoritmo CCSDS – 123, tal y como se puede observar de una forma más acusada con las imágenes de los sensores AIRS e IASI. Esto se explica en parte debido a que esos elevados ratios de compresión previos a la aplicación del algoritmo CCSDS – 123 se consiguen con elevados ratios en bandas (P_b), del orden de 128, 256 y 512 para la imagen del sensor AIRS y de 512, 1024 y 2048 en el caso de la imagen del sensor IASI. Por otra parte, también se realiza compresión en píxeles (P_p), en donde se toman en estos casos imágenes relativamente pequeñas y se comprimen bastante espacialmente, con ratios que llegan hasta valores de 16. Consecuencia de lo cual, ni la redundancia espectral ni la espacial son excesivamente altas en las imágenes multiespectrales e hiperespectrales que se consiguen, por lo que el algoritmo CCSDS – 123 no puede comprimir las mucho más, haciendo que la mejora en el ratio de compresión de ellas se vea reducida. Algo equivalente sucede en las imágenes de los sensores AVIRIS y CRISM, ya que si bien los ratios en bandas que se utilizaron para realizar las simulaciones no son tan elevados (alcanzando como mucho valores de 64), dejan a la imagen multiespectral resultante con aproximadamente 4 y 8 bandas respectivamente en ambas imágenes. Esto, junto con el hecho de que también se reducen espacialmente con ratios que alcanzan valores de 32 (hay que tener en cuenta que estas imágenes son mayores), se ve reducido el efecto que produce el algoritmo CCSDS – 123 cuando el compresor propuesto comprime mucho tanto en bandas como en píxeles.

Resumiendo, si bien para las escenas captadas por los sensores AVIRIS y CRISM se generan mejoras considerables al comprimir las imágenes con el algoritmo CCSDS – 123, cuando los ratios de compresión obtenidos por el compresor CDF generan alrededor de 0,15

bpppb, no sería aconsejable tratar de aumentar la compresión obtenida haciendo uso de éste como paso adicional a la compresión inicial puesto que las mejoras conseguidas no compensan el aumentar la complejidad del proceso introduciendo el algoritmo CCSDS – 123.

4.4.- Versatilidad del compresor CFUN

4.4.1.-Posibilidad de realizar la compresión y descompresión de la imagen por macrobloques

El compresor CFUN trata de extraer de forma secuencial los vectores que mejor representen la información presente en la imagen, para luego proyectar esta imagen sobre dichos vectores. Estos vectores tienen ciertas características importantes. En primer lugar, son ortogonales entre sí, lo cual asegura que la información que es capaz de representar cada vector es información que no está representada por el resto de vectores extraídos, eliminando de esta manera la posibilidad de añadir información redundante, y por lo tanto innecesaria, en la imagen comprimida. En segundo lugar, el primer vector es directamente un píxel de la imagen, y el resto de vectores se corresponden con la parte de los píxeles de la imagen que no puede ser representada por los vectores previamente extraídos. Este hecho hace que el proceso de extracción de los vectores de proyección sea mucho más sencillo que si dichos vectores debieran ser inferidos de alguna otra manera, pudiéndose emplear en este caso el algoritmo modificado de Gram-Schmidt, siendo los productos escalares de vectores las operaciones más complejas requeridas. Por último, el compresor CFUN permite también que el proceso global de compresión sea dividido en procesos de compresión más pequeños ejecutados de forma independiente para diferentes fracciones de la imagen (macrobloques), con la ventaja de que, si el tamaño de los píxeles (vectores de proyección) es mucho menor que el número de píxeles en cada macrobloque, se obtendrán, para cada macrobloque, vectores más representativos que si se realizase el proceso para toda la imagen, obteniendo de esta manera menores pérdidas de información en el proceso de compresión para mismos ratios de compresión.

Además de ofrecer la posibilidad de obtener mejores resultados, el hecho de poder dividir el proceso de compresión en macrobloques independientes ofrece importantes ventajas:

- Tolerancia a fallos: si se produjese un error en un macrobloque, el resto de macrobloques no se verían afectados por este error.

- Paralelización del proceso: para aplicaciones que requieran su ejecución en tiempo real ésta es una característica muy importante.
- Reducción de la memoria requerida para almacenar las imágenes: no es necesario esperar a que la imagen se adquiriera completamente para ejecutar la compresión al poder empezarse a comprimir cada macrobloque desde que vayan siendo adquiridos.

Con el objetivo de verificar la posibilidad de ejecución del compresor CFUN empleando macrobloques, se han llevado a cabo simulaciones en las que las imágenes a comprimir se han dividido en macrobloques de 100x100 píxeles, de 50x50 píxeles, y de 25x25 píxeles, comparando en cada caso los resultados con los obtenidos en la ejecución del compresor CFUN empleando la imagen completa.

Las *Gráficas A.1.1.2.* y *A1.2.2.* del anexo I recogen los resultados de las simulaciones realizadas para la imagen obtenida por el sensor AIRS haciendo uso del compresor CFUN para las distintas pruebas planificadas: imagen completa y macrobloques correspondientes. Así, en el anexo II se presentan las *Gráficas A.2.1.2.* y *A.2.2.2.*, equivalentes a las anteriores pero que resultan de realizar las simulaciones a la imagen del sensor AVIRIS. Por su parte, las *Gráficas A.3.1.2.* y *A.3.2.2.* del anexo III se corresponderían con las que resultan de ejecutar el compresor CFUN para la imagen del sensor CRISM, y las *Gráficas A.4.1.2.* y *A.4.2.2.* recogen los resultados de las simulaciones realizadas con la imagen del sensor IASI.

Atendiendo a los resultados obtenidos se puede observar que el proceso de compresión empleando el compresor CFUN y dividiendo la imagen en macrobloques proporciona resultados similares a los obtenidos con la imagen completa, obteniéndose especial mejora en los resultados en aquellas imágenes en las que el número de bandas sea considerablemente menor que el número de píxeles por macrobloque.

Es importante mencionar llegados a este punto, que el compresor CFUN procede del algoritmo de unmixing FUN, el cual trata de seleccionar píxeles de la imagen como endmembers. Sin embargo, para el proceso de compresión, simplemente es necesario extraer ciertos vectores que permitan proyectar la información de la imagen de la forma más fiel posible. Por este motivo, el compresor CFUN puede ser ejecutado, idénticamente, por bandas en lugar de por píxeles, de forma equivalente a como se explica en el *capítulo 2* de esta memoria, obteniendo ventajas importantes a la hora de ser ejecutado por macrobloques, para aquellas imágenes en las que el número de bandas es muy elevado, como son el caso de las

imágenes obtenidas por los satélites IASI y AIRS. En estos casos, la condición para que la división en macrobloques sea más beneficiosa, sería deseable, de forma análoga a lo expuesto anteriormente, que el número de píxeles sea mucho menor que el número de bandas por macrobloques.

Se concluye pues que, el compresor CFUN puede ser ejecutado por macrobloques obteniendo mejores resultados que si se ejecutase para la imagen completa, y aprovechando para ello las ventajas derivadas de su paralelización. Para que este proceso sea óptimo, la imagen puede dividirse por píxeles o por bandas, siendo el segundo caso el recomendado para imágenes con un número muy elevado de bandas.

4.4.2.-Posibilidad de realizar la compresión y descompresión de la imagen tanto por píxeles como por bandas

El compresor CFUN procede del algoritmo de unmixing FUN, el cual trata de seleccionar píxeles de la imagen como endmembers y por lo tanto se ejecuta por píxeles. Sin embargo, para el proceso de compresión, simplemente es necesario extraer ciertos vectores que permitan proyectar la información de la imagen de la forma más fiel posible. Por este motivo, el compresor CFUN puede ser ejecutado, idénticamente, por bandas en lugar de por píxeles, o lo que es lo mismo, los vectores de proyección extraídos pueden ser píxeles o bandas. El proceso llevado a cabo por el compresor CFUN en ambos casos es exactamente el mismo, modificándose únicamente la orientación con la que se introduce la imagen a comprimir, es decir, el orden en que se introducen los datos en el compresor.

Cuando el compresor CFUN es ejecutado por píxeles, éste descomprime cada píxel, de forma independiente, como una combinación lineal de los vectores extraídos. De forma análoga, cuando se ejecuta por bandas, el algoritmo CFUN descomprime cada banda, de forma independiente, como una combinación lineal de los vectores extraídos.

Con el objetivo de verificar esta propiedad del compresor CFUN se han realizado simulaciones empleando una imagen obtenida por el sensor AVIRIS sobre la región de Washington DC. Los resultados obtenidos por estas simulaciones se encuentran en la tabla 3. En ella, las columnas correspondientes al campo CFUN_p corresponden con los valores de las métricas obtenidos ejecutando el algoritmo por píxeles, mientras que los que están debajo de CFUN_b, son los valores de las métricas utilizadas para el compresor CFUN aplicado por bandas.

Washington DC				
CFUNp			CFUNb	
rc	SNR	SAM	SNR	SAM
4	40,622	0,631	44.744	0.324
8	35,407	1,200	40.653	0.515
16	30,686	2,088	37.236	0.737
32	22,622	5,186	33.769	1.066
64	17,918	8,337	23.227	3.899

Tabla 3. Resultados de las simulaciones sobre Washington DC del compresor CFUN por píxeles y por bandas

A la vista de los resultados se concluye que el proceso de compresión que lleva a cabo el algoritmo CFUN puede ser llevado a cabo con éxito tanto por píxeles como por bandas. De esta forma se adquiere una especial ventaja en la ejecución por macrobloques ya que por un lado permite la configuración de los macrobloques con las dimensiones que proporcionen mejores resultados, y por otro lado, el compresor CFUN es capaz de adaptarse a la forma de adquisición de datos de los sensores, ya sean Band-Sequential (BSQ) o Band-Interleaved-by-Pixel (BIP), ofreciendo la posibilidad de empezar el proceso de compresión en todos los casos antes de terminar el proceso de adquisición de datos.

4.4.3.- Filtro de ruido de la imagen

En el proceso de adquisición de datos llevado a cabo por los sensores hiperespectrales y ultraespectrales es inevitable la introducción de ruido en la imagen adquirida. La potencia del ruido introducido puede variar, atendiendo a las características del sensor. La relación entre potencia de la señal de la imagen y la potencia del ruido presente en la imagen se caracteriza con el parámetro Signal-to-Noise Ratio (SNR) tal y como se muestra en la expresión 3.1. Valores altos de este parámetro indican que la potencia de la señal de la imagen es mucho mayor que la potencia del ruido.

La existencia de ruido en las imágenes hiperespectrales y ultraespectrales afecta a los procesos de análisis que se realizan empleando dichas imágenes. Por éste motivo, ha surgido la necesidad de llevar a cabo un filtrado del ruido presente en la imagen como primer paso del análisis de la misma. Sin embargo, el proceso de filtro de ruido de la imagen es siempre llevado a cabo en tierra, cuando la imagen ya ha sido comprimida y descomprimida. Por lo tanto, con el objetivo de comprimir y descomprimir la imagen de la forma más fiel posible, realmente se está comprimiendo y descomprimiendo tanto la propia señal de la imagen, como la señal correspondiente al ruido. Este hecho introduce una limitación importante en los procesos de

compresión y descompresión, ya que estos se aprovechan de las redundancias, tanto espaciales como espectrales, presentes en la imagen, y el ruido, por lo general, tiene naturaleza aleatoria.

En relación a lo expuesto en el párrafo anterior, el compresor CFUN presenta una importante mejora respecto a los algoritmos de compresión que componen el estado del arte, ya que por la naturaleza del mismo, las pérdidas introducidas en el proceso de compresión permiten eliminar parte del ruido presente en la imagen. Esto se debe a que el compresor CFUN reconstruye la imagen a partir de la combinación lineal de vectores, técnica que es empleada también por otros algoritmos con el objetivo de filtrar el ruido de la imagen. Éste es el caso de algoritmo hyperspectral noise estimation (estNoise) [10], el cual asume que cada banda de la imagen hiperespectral puede ser representada como una combinación lineal de las bandas restantes, y emplea este criterio para eliminar el ruido presente en la imagen.

Con el objetivo de verificar este hecho, se han llevado a cabo simulaciones empleando para ello la imagen obtenida por el sensor AVIRIS sobre la región de Washington DC. A partir de esta imagen se han generado tres imágenes adicionales, a las que se le han añadido ruido de forma artificial, atendiendo a distintos valores de SNR. El ruido añadido ha sido ruido gaussiano de media cero. Las imágenes con ruido artificial añadido se han comprimido y descomprimido, usando para ello el compresor CFUN (por píxeles y por bandas) y el compresor LCE, y se ha comparado en cada caso la imagen descomprimida con la imagen de entrada del compresor (imagen con ruido artificial), y con la imagen original de Washington DC (imagen sin ruido añadido).

Los resultados de las simulaciones realizadas se encuentran en las tablas 4 y 5, en donde se presentan los resultados obtenidos de las pruebas hechas al comparar las imágenes de Washington DC comprimidas respecto a la imagen Washington DC original (sin ruido) y respecto a la que se le añadió ruido gaussiano. En las columnas de la tabla cuyo nombre es CFUNp, se exponen los resultados de las simulaciones haciendo uso del compresor CFUN por píxeles de la imagen, en aquellas denominadas CFUNb se recogen los resultados obtenidos al aplicar el compresor CFUN por bandas de la imagen, y en las llamadas LCE, se agrupan los resultados de las simulaciones realizadas con el algoritmo LCE. En éstas últimas columnas se pueden apreciar registros de las tablas 4 y 5 sin rellenar, esto se debe a la imposibilidad del algoritmo LCE de descomprimir las imágenes que previamente había comprimido.

Ruido añadido		Washington DC + Ruido gaussiano					
		CFUNp		CFUNb		LCE	
SNR	rc	SNR	SAM	SNR	SAM	SNR	SAM
20	4	22,864	4,948	22,983	4,847	-15,219	88,959
	8	21,924	5,509	22,045	5,360		
	16	20,845	6,243	21,404	5,702		
	32	18,807	8,056	20,798	6,032		
	64	16,373	10,371	18,900	7,241		
40	4	38,370	0,825	40,297	0,627	-14,986	89,168
	8	34,551	1,324	38,068	0,783		
	16	30,482	2,125	35,595	0,975		
	32	22,537	5,263	32,968	1,252		
	64	18,012	8,256	23,152	3,954		
60	4	41,175	0,587	44,666	0,329	17,633	3,011
	8	35,465	1,192	40,616	0,519	17,327	4,530
	16	30,681	2,089	37,217	0,740	15,477	7,892
	32	22,623	5,185	33,761	1,068	12,645	10,880
	64	17,918	8,337	23,226	3,900	10,480	12,534

Tabla 4. Simulaciones sobre Washington DC del compresor CFUN por píxeles y por bandas, y del LCE comparando con la imagen original con ruido añadido

Ruido añadido		Respecto a imagen Washington DC sin ruido gaussiano					
		CFUNp		CFUNb		LCE	
SNR	rc	SNR	SAM	SNR	SAM	SNR	SAM
20	4	26,877	3,088	27,183	2,917	-15,246	88,954
	8	27,663	2,800	28,130	2,467		
	16	25,628	3,547	27,601	2,426		
	32	21,369	6,038	26,147	2,749		
	64	17,681	8,767	21,661	4,654		
40	4	39,429	0,727	42,084	0,487	-14,987	89,168
	8	35,183	1,232	39,646	0,611		
	16	30,760	2,058	36,570	0,814		
	32	22,583	5,233	33,511	1,115		
	64	18,028	8,235	23,207	3,909		
60	4	41,193	0,586	44,706	0,326	17,633	3,013
	8	35,472	1,191	40,639	0,517	17,327	4,530
	16	30,684	2,088	37,230	0,738	15,477	7,892
	32	22,623	5,185	33,767	1,066	12,645	10,880
	64	17,918	8,337	23,227	3,899	10,480	12,534

Tabla 5. Simulaciones sobre Washington DC del compresor CFUN, por píxeles y por bandas, y del LCE comparando con la imagen original sin ruido gaussiano añadido

A la vista de los resultados pueden extraerse varias conclusiones. En primer lugar, se verifica que la imagen reconstruida tras comprimir y descomprimir las imágenes con ruido empleando el algoritmo CFUN en sus dos versiones son más semejantes a la imagen original, sin ruido, que a las imágenes de entrada del compresor con ruido gaussiano añadido, tal y como muestran los valores de SNR y SAM obtenidos en ambos casos. Esto verifica que el

proceso de compresión y descompresión llevado a cabo por el algoritmo CFUN elimina parte del ruido presente en la imagen, siendo la imagen resultante más similar a la imagen ideal, sin ruido. En segundo lugar, atendiendo a los resultados obtenidos empleando el compresor LCE, se verifica que la presencia de ruido en la escena fotografiada afecta notablemente al proceso de compresión llevado a cabo por los algoritmos tradicionales de compresión, afectando de forma importante a los resultados obtenidos.

Capítulo 5. Conclusiones

En este capítulo se exponen las conclusiones a las que se ha llegado tras completar los objetivos que se marcaron al inicio del TFM, teniendo en cuenta los resultados de las pruebas realizadas para todos los casos que se han expuesto a lo largo de esta memoria.

5.1.- Conclusiones relacionadas con las propuestas de compresión de imágenes presentadas en este TFM

En la actualidad existen satélites que cuentan con sensores capaces de captar imágenes hiperespectrales y ultraspectrales de la superficie terrestre. Estas imágenes tienen numerosas aplicaciones debido a la gran cantidad de información que aportan. Sin embargo, la capacidad de adquisición de datos de este tipo de sensores es mayor, y ha aumentado más rápidamente en los últimos años, que la capacidad de los satélites en los que se encuentran dichos sensores, para transmitir estas imágenes a la tierra, donde son procesadas. Debido a este hecho la compresión de imágenes hiperespectrales y ultraspectrales ha adquirido gran importancia, si bien dicho proceso ofrece una problemática que no se encuentra en absoluto solventada.

A la necesidad de comprimir las imágenes de forma que se alcancen ratios de compresión suficientemente elevados que permitan enviar las imágenes adquiridas a tierra, con la menor pérdida de información, se suman las restricciones añadidas por el hecho de que la compresión debe realizarse a bordo de los satélites en los que se encuentran los sensores que adquieren dichas imágenes. Por este motivo, los algoritmos empleados para realizar la compresión de imágenes hiperespectrales y ultraspectrales idealmente deben ser “hardware friendly”, es decir, ser lo más sencillo posible, no contener operaciones matriciales complejas y ser fácilmente paralelizables e implementables, ya que para la compresión de este tipo de imágenes a bordo de los satélites, las restricciones de área, potencia y tiempo están muy presentes. Además es deseable que los algoritmos empleados para la compresión sean tolerantes a fallos.

En este trabajo fin de máster se han desarrollado dos metodologías alternativas, específicas para la compresión de imágenes hiperespectrales y ultraspectrales, que cumplan con las necesidades previamente explicadas. Para estudiar la bondad de las metodologías propuestas, se han llevado a cabo numerosas simulaciones, empleando imágenes hiperespectrales y ultraspectrales reales captadas por distintos sensores, y comparando los resultados obtenidos con los resultados conseguidos través de los algoritmos que componen el estado del arte para las mismas imágenes.

La primera metodología propuesta consiste en dividir, de forma sumamente simple, la imagen hiperespectral o ultraspectral a comprimir, obteniendo dos imágenes que en conjunto contienen mucha menos información que la imagen inicial, de forma que son estas

imágenes las enviadas a tierra. Una vez en tierra, para recuperar la imagen hiperespectral o ultraspectral original, las imágenes recibidas son fusionadas empleando un algoritmo de fusión de imágenes hiperespectrales y ultraspectrales, como pueden ser los algoritmos GGSA, GGLMSE, CoEf-HMI, CNMF, Bayesian Sparse o Bayesian Naive [11], Hyperspectral image Superresolution (HySURE) [12]. La principal ventaja que ofrece esta metodología es que, a diferencia de las metodologías tradicionales, el proceso de compresión es sumamente simple, siendo el proceso de descompresión, llevado a cabo por el algoritmo de fusión de datos seleccionado, la parte más compleja del proceso. Debido a que la descompresión se realiza en tierra, no existen tantas limitaciones de área, potencia y tiempo, con lo que la complejidad del proceso de fusión de datos empleado no afecta la bondad del proceso. Además, esta metodología permite seleccionar a priori el ratio de compresión que se desea obtener.

La bondad de los resultados obtenidos empleando esta metodología para el proceso de compresión y descompresión de imágenes hiperespectrales y ultraspectrales dependerá principalmente del algoritmo de fusión de datos empleado en la descompresión. En este trabajo se han empleado para ello únicamente los algoritmos de fusión GGSA, GGLMSE y CoEf-HMI, siendo el algoritmo CoEf-HMI el que ha proporcionado los mejores resultados en la gran mayoría de los casos. Sin embargo, existe la posibilidad de que el empleo de otro algoritmo de fusión en la descompresión, o incluso el desarrollo de nuevas técnicas de fusión de datos, mejores de forma pasiva los resultados obtenidos empleando esta metodología para el proceso de compresión y descompresión de imágenes hiperespectrales y ultraspectrales, lo cual supone una importante ventaja. En vista de los resultados obtenidos, se concluye que el empleo de la metodología basada en fusión de datos para la compresión de imágenes hiperespectrales y ultraspectrales permite alcanzar ratios de compresión elevados con una calidad muy aceptable en la imagen descomprimida.

La segunda metodología propuesta para el proceso de compresión y descompresión de imágenes hiperespectrales y ultraspectrales está basada en los conceptos de desmezclado espectral, y en particular en el algoritmo de desmezclado espectral Fast algorithm for linearly UNmixing hyperspectral images (FUN). Tras realizar pequeñas modificaciones al algoritmo FUN se ha desarrollado el compresor CFUN, el cual ha demostrado tener numerosas ventajas:

- Permite introducir el SNR y el ratio de compresión que se desean obtener en el proceso de compresión y descompresión de la imagen hiperespectral o ultraspectral.

- Es hardware-friendly. No realiza operaciones matriciales complejas y es altamente paralelizable.
- Admite la división de la imagen en macrobloques, ejecutando cada uno de ellos de forma totalmente independiente, sin reducción de la calidad de los resultados de compresión obtenidos.
- Tolerancia a fallos. La división de la imagen en macrobloques independientes ofrece cierta robustez a fallos, ya que la existencia de un fallo en un macrobloque no afectaría al resto de macrobloques de la imagen.
- Admite la ejecución del algoritmo tanto por píxeles como por bandas, sin realizar ninguna modificación del mismo, siendo esto posible tanto si se trabaja con la imagen completa como si se trabaja con macrobloques. Esto facilita la adaptación del compresor a todo tipo de sensores, independientemente del proceso de adquisición de datos del mismo (BSQ o BIP).
- En el proceso de compresión y descompresión empleando el compresor CFUN se elimina parte del ruido presente en la imagen a comprimir, sin necesidad de hacer ningún cálculo adicional.

En vista de los resultados obtenidos en las simulaciones realizadas, se concluye que el compresor CFUN proporciona resultados altamente competentes en relación con los algoritmos que componen el estado del arte, especialmente para ratios de compresión elevado. Este algoritmo proporcionando ventajas adicionales derivadas de su versatilidad y sencillez de ejecución, de su carácter sencillo y paralelo que facilitan su implementación en hardware, y de su fácil adaptabilidad a los distintos procesos de adquisición de datos (BIP y BSQ). Además, este método de compresión se diferencia del resto de métodos de compresión por el hecho de ser capaz de eliminar parte del ruido presente en la imagen.

Referencias.

- [1] Enrique de la Fuente Milanés (2012) “Compresión de imágenes hiperespectrales en tarjetas gráficas programables”. Proyecto Fin de Carrera, Escuela Politécnica Ingeniería informática, Universidad de Cáceres.
- [2] Naoto Yokoya, Student Member, IEEE, Takehisa Yairi, and Akira Iwasaki (2012) “Coupled Nonnegative Matrix Factorization Unmixing for Hyperspectral and Multispectral Data Fusion”. *IEEE Transactions On Geoscience and Remote Sensing*, Vol. 50, No. 2, February 2012.
- [3] Zhao Chen, Hanye Pu, Bin Wang, Senior Member, IEEE, and Geng-Ming Jiang, Member, IEEE (2014) “Fusion on Hyperspectral and Multispectral Images: A Novel Framework Based on Generalization of Pan-Sharpener Methods”. *IEEE Geoscience and Remote Sensing Letters*, Vol.11, No. 8, August 2014.
- [4] Mohamed Amine Bendoumi, Mingyi He, Member, IEEE, and Shaohui Mei, Member, IEEE. “Hyperspectral Image Resolution Enhancement Using High-Resolution Multispectral Image Based on Spectral Unmixing”. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 52, No. 10, October 2014
- [5]* Guerra, R., Lopez, S., & Sarmiento, R. (2015, “A computationally efficient algorithm for fusing multispectral and hyperspectral images,” *Geoscience and Remote Sensing, IEEE Transactions*.
- [6] Guerra, R., Santos, L., Lopez, S., & Sarmiento, R. (2015). A New Fast Algorithm for Linearly Unmixing Hyperspectral Images. *Geoscience and Remote Sensing, IEEE Transactions on*, 53(12), 6752-6765.
- [7] Du, Q., & Fowler, J. E. (2007). Hyperspectral image compression using JPEG2000 and principal component analysis. *Geoscience and Remote Sensing Letters, IEEE*, 4(2), 201-205.
- [8] Abrardo, A., Barni, M., & Magli, E. (2011, May). Low-complexity predictive lossy compression of hyperspectral and ultraspectral images. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on* (pp. 797-800). IEEE.
- [9] Jose Enrique Sánchez, Estanislau Augé, Josep Santaló, Ian Blanes, Joan Serra-Sagristà, Aaron Kiely, “Review and implementation of the emerging ccsds recommended standard for

multispectral and hyperspectral lossless image coding,” CCP 2011, Proceedings, vol.1, june 2011

[10] J. M. Bioucas-Dias and J. M. Nascimento, “Hyperspectral subspace identification,” *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 46, no. 8, pp. 2435–2445, 2008.

[11] Q. Wei, J. Bioucas-Dias, N. Dobigeon, and J.-Y. Tourneret, “Hyperspectral and multispectral image fusion based on a sparse representation,” *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 53, no. 7, pp. 3658–3668, 2015.

[12] M. Simões, J. Bioucas-Dias, L. B. Almeida, and J. Chanussot, “A convex formulation for hyperspectral image superresolution via subspacebased regularization,” *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 53, no. 6, pp. 3373–3388, 2015.

* Referencia a la espera de ser publicada

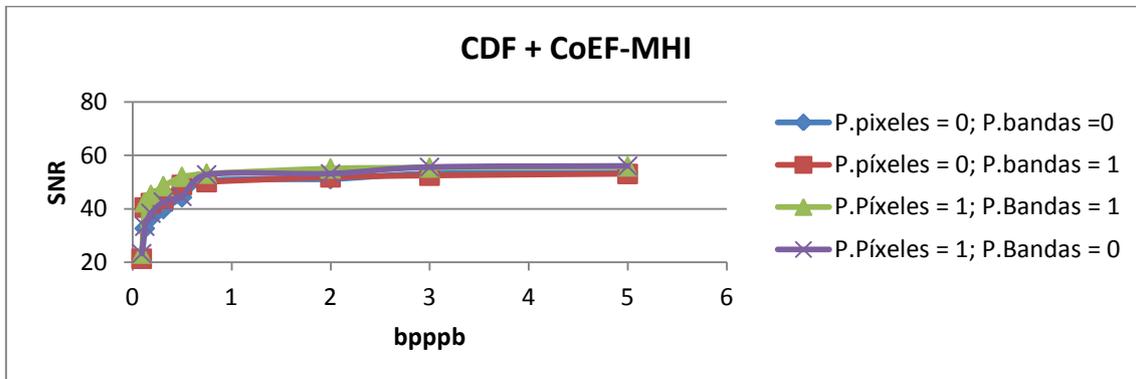
Anexos

En este capítulo se van a recoger todas las gráficas que han resultado de cada una de las simulaciones realizadas a lo largo de este TFM, y que apoyan todas las conclusiones a las que se ha llegado hasta el momento durante el desarrollo de la memoria.

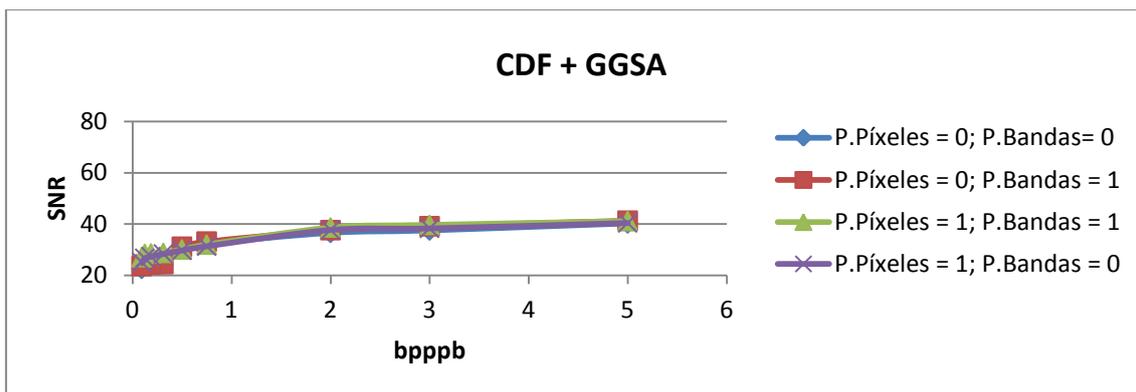
A.1.- Anexo I

A.1.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS. Métrica: SNR

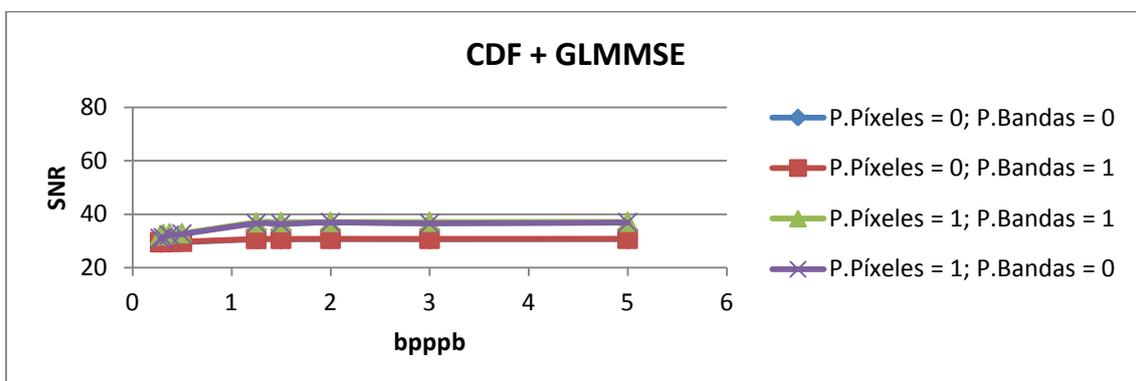
A.1.1.1.- Compresor CDF + Data Fusion



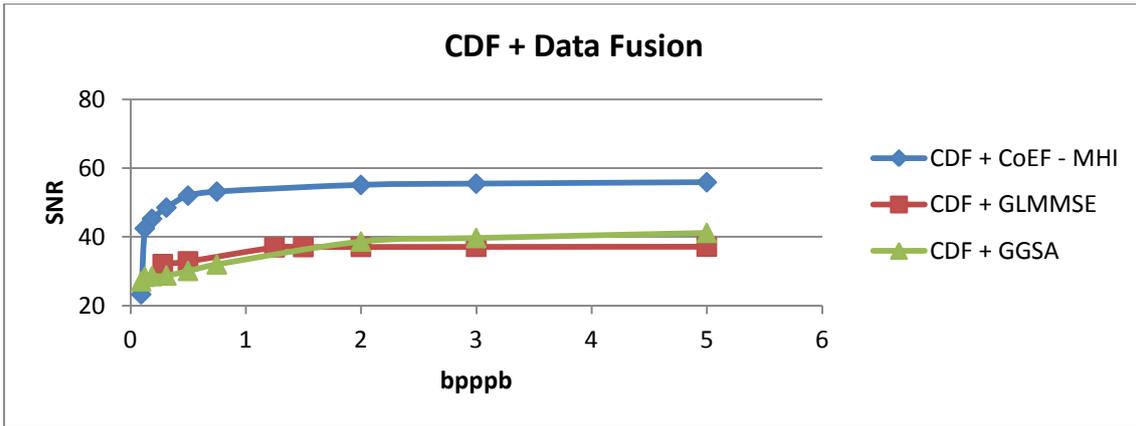
Gráfica A.1.1.1.1: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo CoEF - MHI



Gráfica A.1.1.1.2: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GGSA

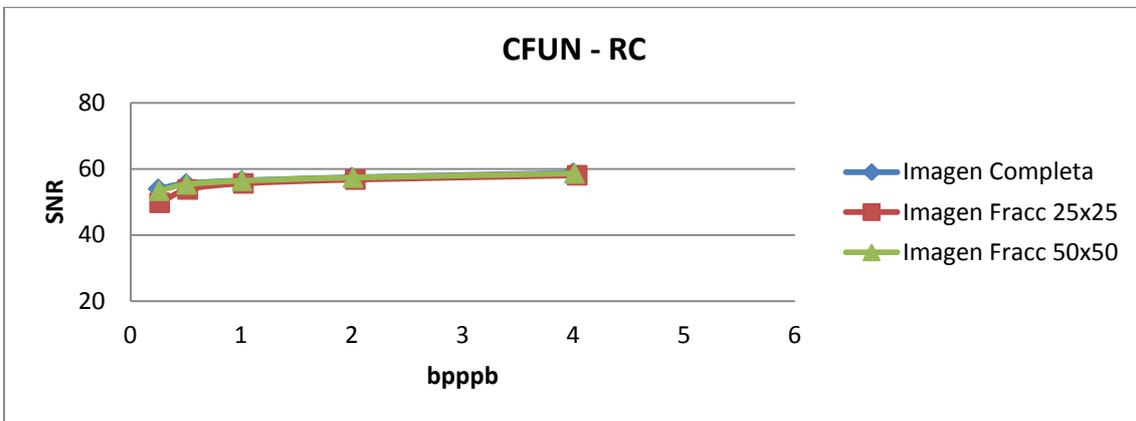


Gráfica A.1.1.1.3: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GLMMSE

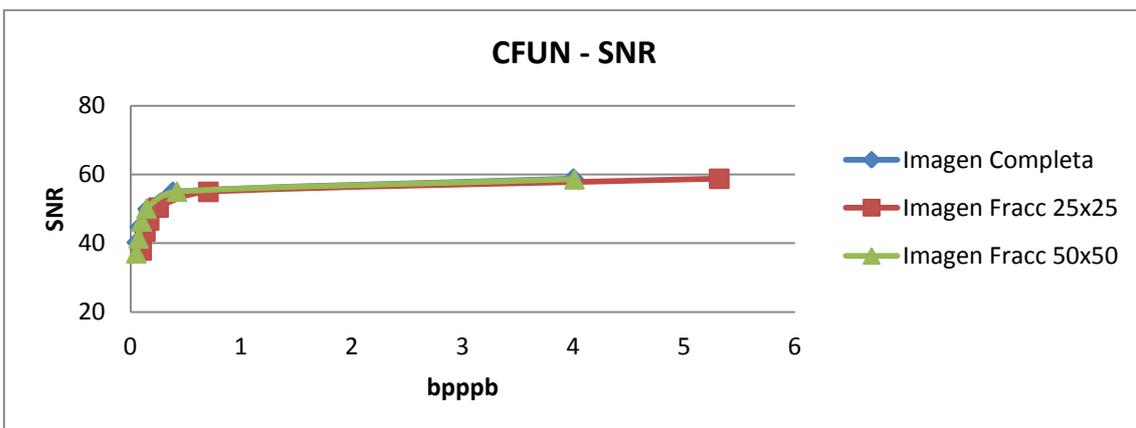


Gráfica A.1.1.1.4: Comparación de los mejores resultados obtenidos para cada uno de los algoritmos de Data Fusion, tras ser comprimida la imagen con el compresor CDF

A.1.1.2.- Compresor CFUN

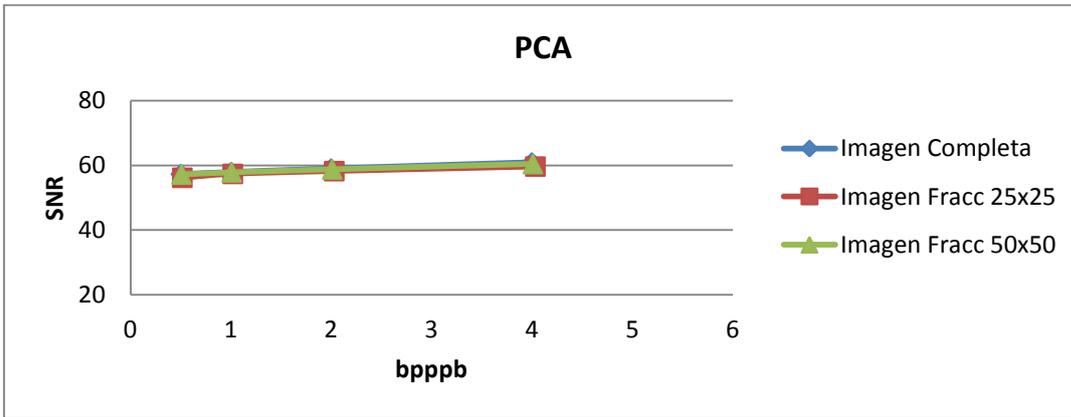


Gráfica A.1.1.2.1: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

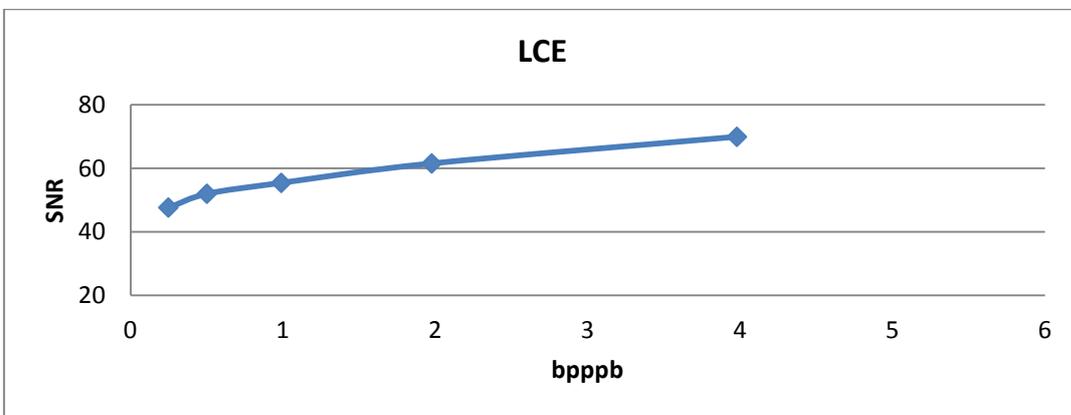


Gráfica A.1.1.2.2: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

A.1.1.3.- Algoritmos de comparación

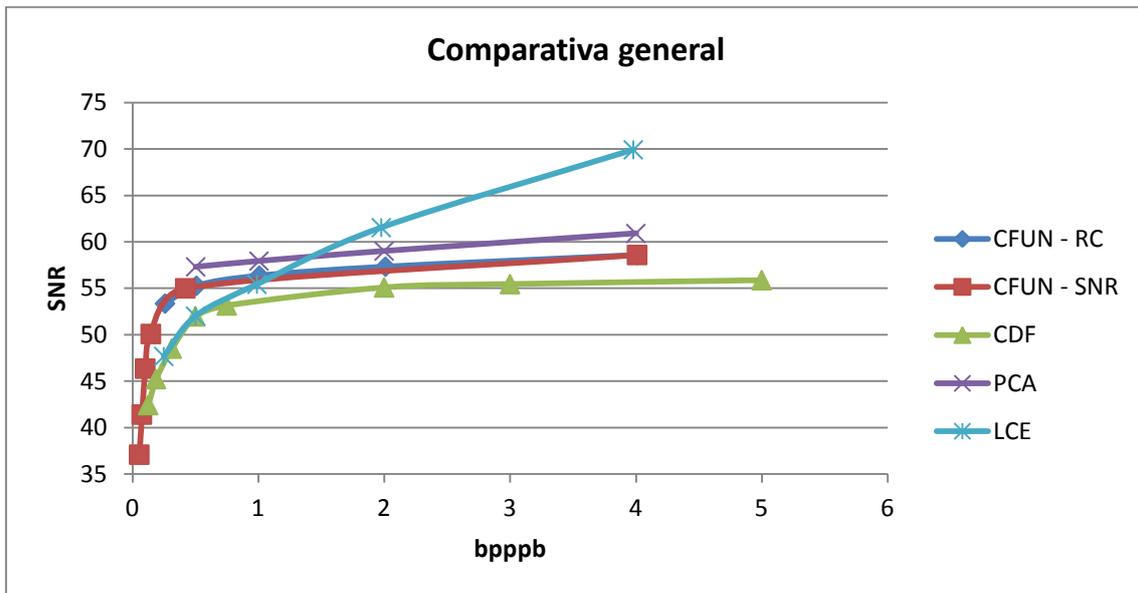


Gráfica A.1.1.3.1: Resultados de la compresión usando el PCA con diferentes configuraciones de la imagen (entera o macrobloques)



Gráfica A.1.1.3.2: Resultados de la compresión usando el LCE

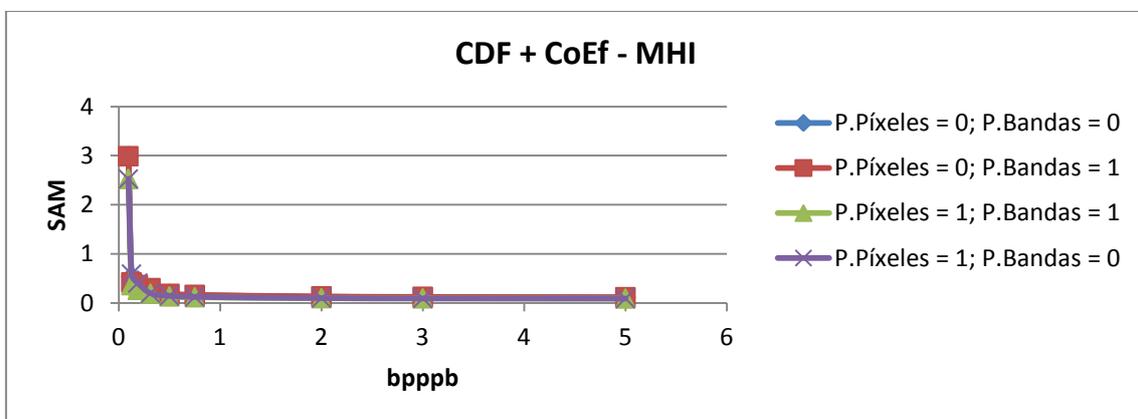
A.1.1.4.- Comparativa general



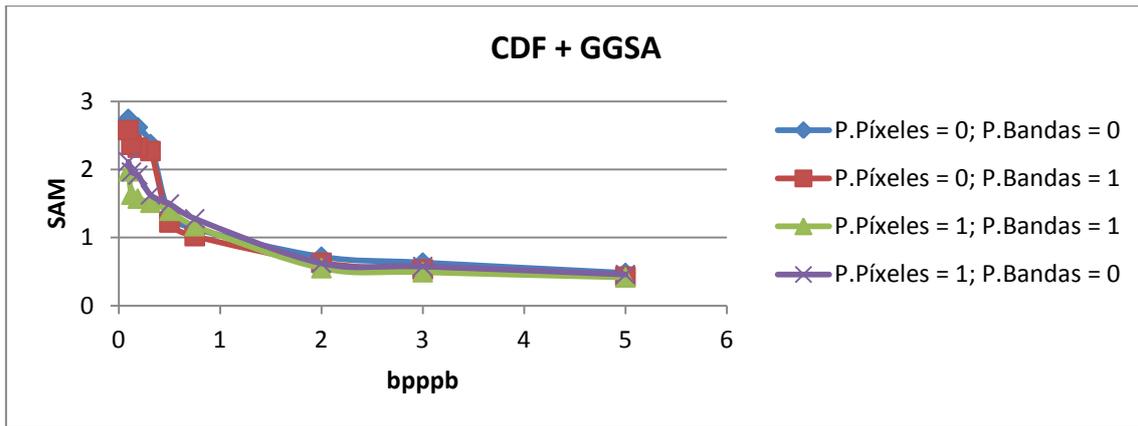
Gráfica A.1.1.4.1: Comparativa general entre los resultados de los algoritmos del estado del arte de compresión y los mejores resultados conseguidos con las metodologías propuestas

A.1.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS. Métrica: SAM

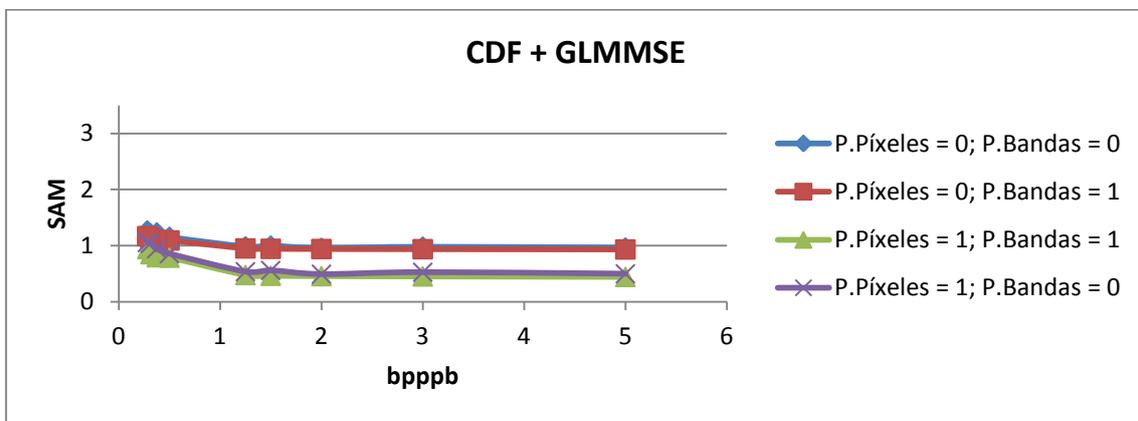
A.1.2.1.- Compresor CDF + Data Fusion



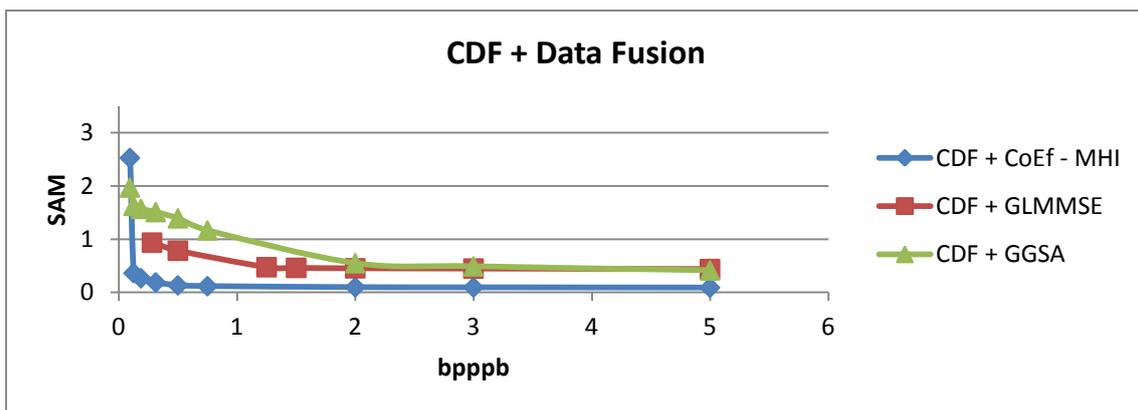
Gráfica A.1.2.1.1: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo CoEf - MHI



Gráfica A.1.2.1.2: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GGSA

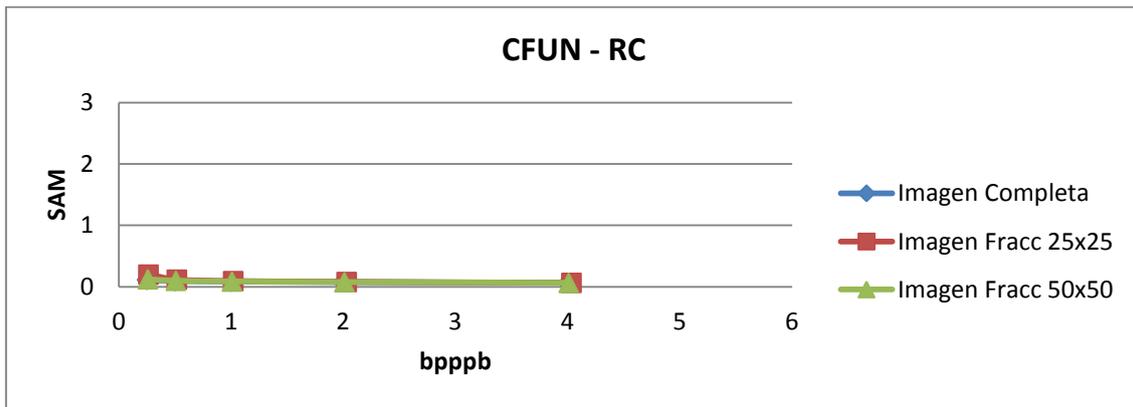


Gráfica A.1.2.1.3: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GLMMSE

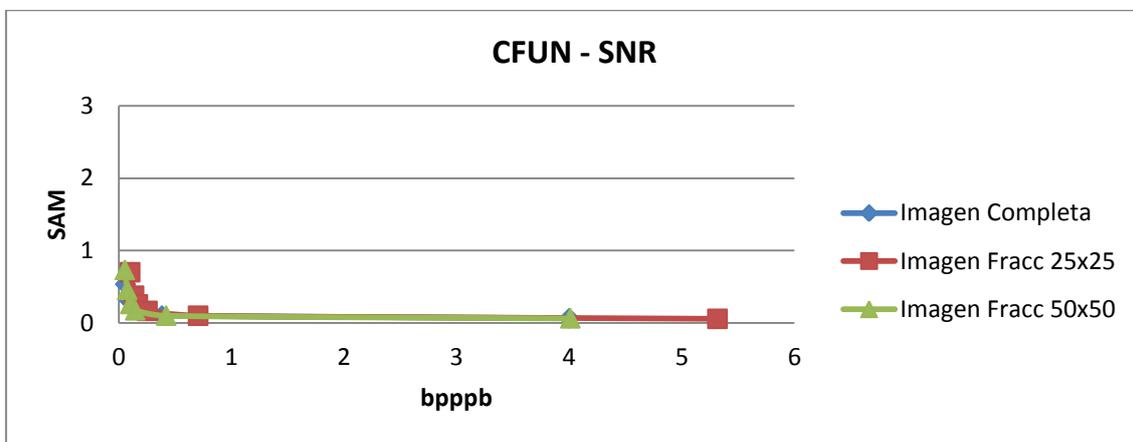


Gráfica A.1.2.1.4: Comparación de los mejores resultados obtenidos para cada uno de los algoritmos de Data Fusion, tras ser comprimida la imagen con el compresor CDF

A.1.2.2.- Compresor CFUN

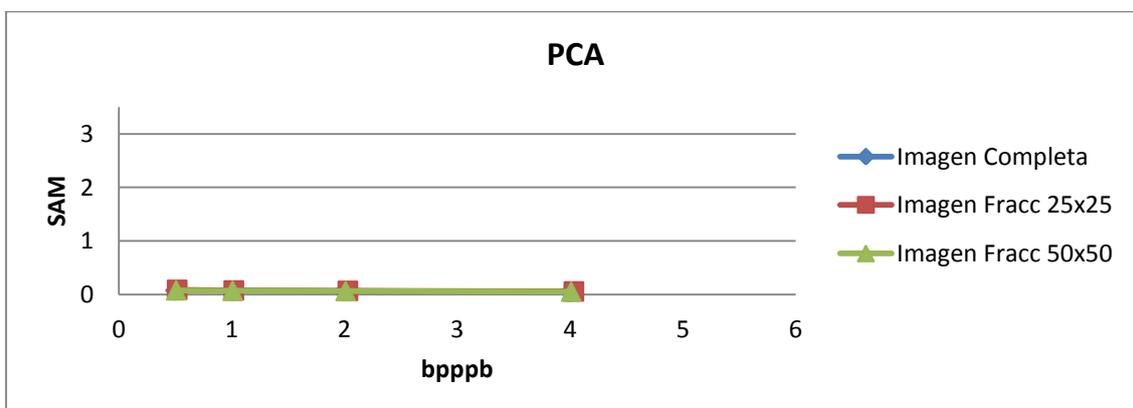


Gráfica A.1.2.2.1: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

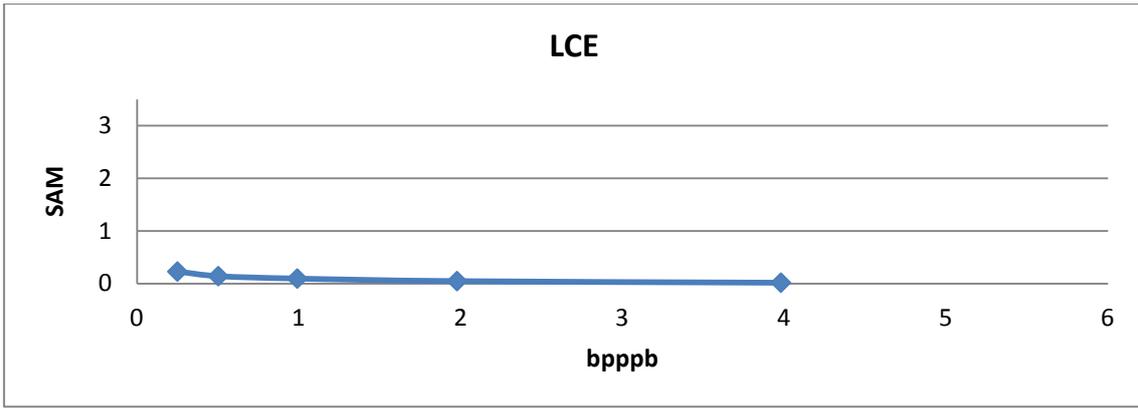


Gráfica A.1.2.2.2: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

A.1.2.3.- Algoritmos de comparación

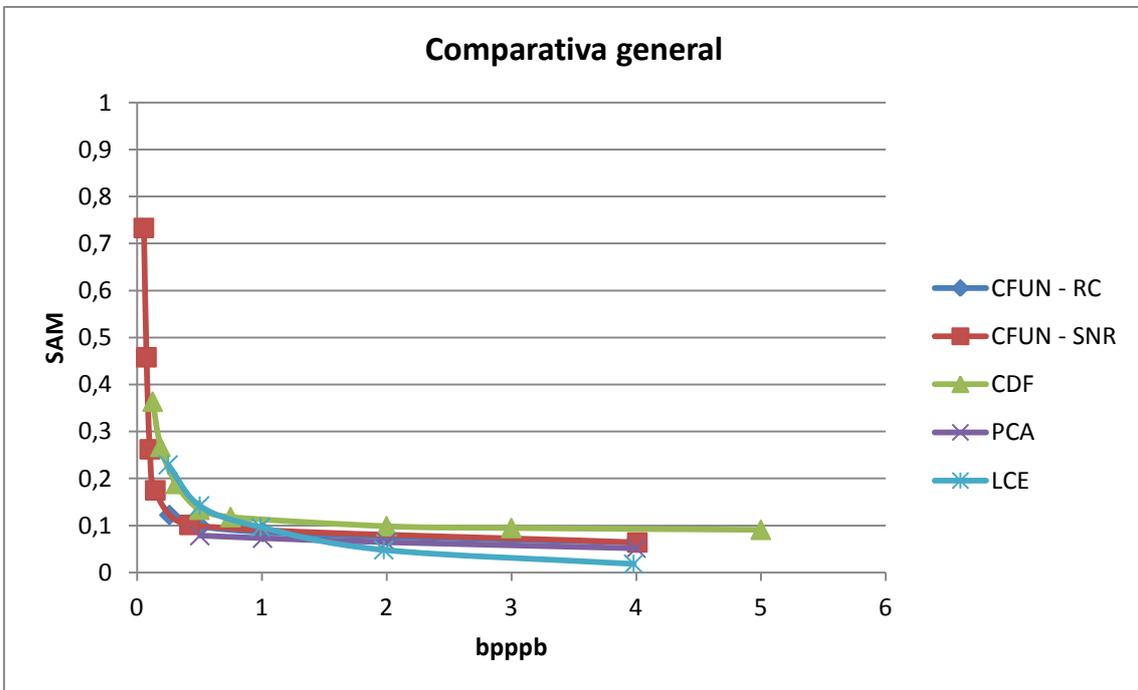


Gráfica A.1.2.3.1: Resultados de la compresión usando el PCA con diferentes configuraciones de la imagen (entera o macrobloques)



Gráfica A.1.2.3.2: Resultados de la compresión usando el LCE

A.1.2.4.- Comparativa general

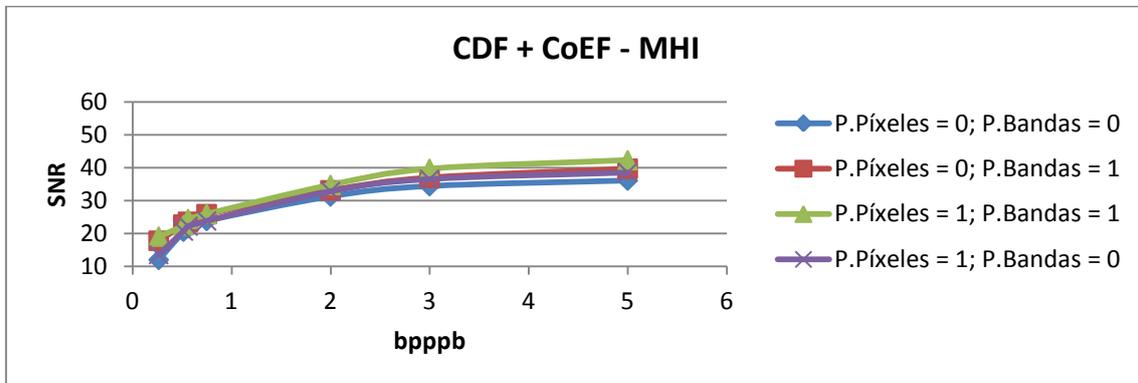


Gráfica A.1.2.4.1: Comparativa general entre los resultados de los algoritmos del estado del arte de compresión y los mejores resultados conseguidos con las metodologías propuestas

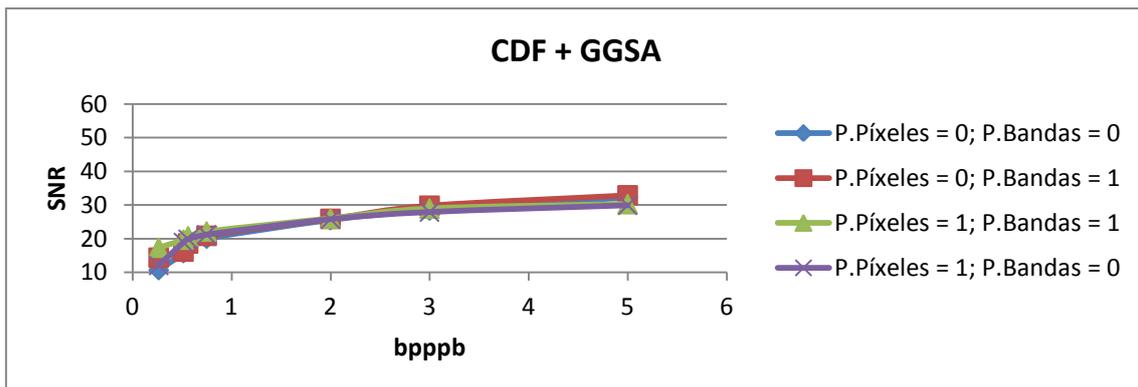
A.2.- Anexo II

A.2.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS. Métrica: SNR

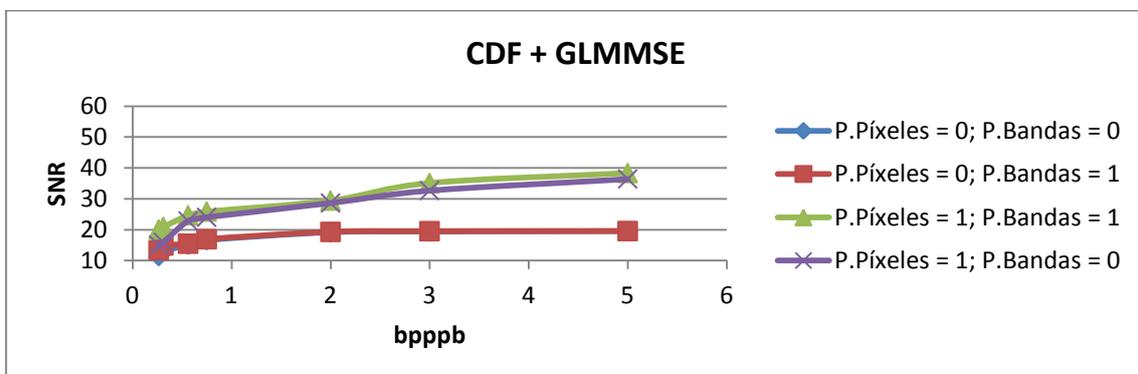
A.2.1.1.- Compresor CDF + Data Fusion



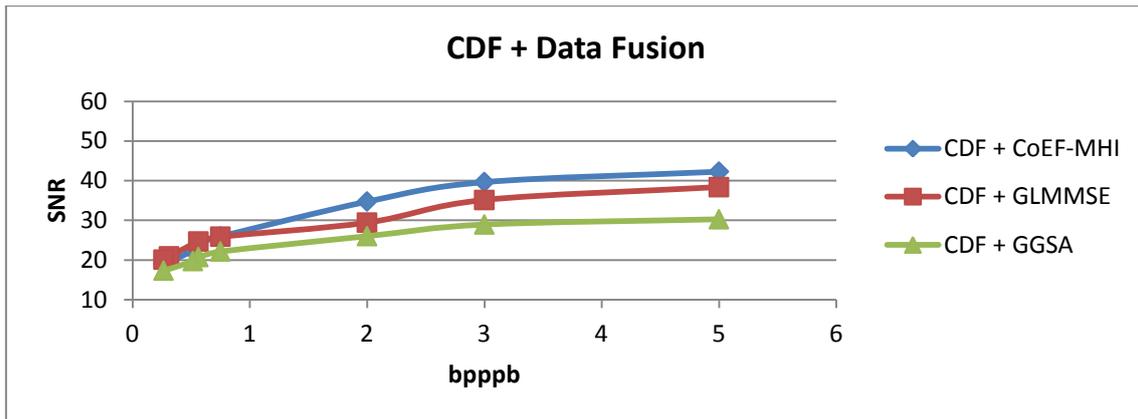
Gráfica A.2.1.1.1: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo CoEF - MHI



Gráfica A.2.1.1.2: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GGSA

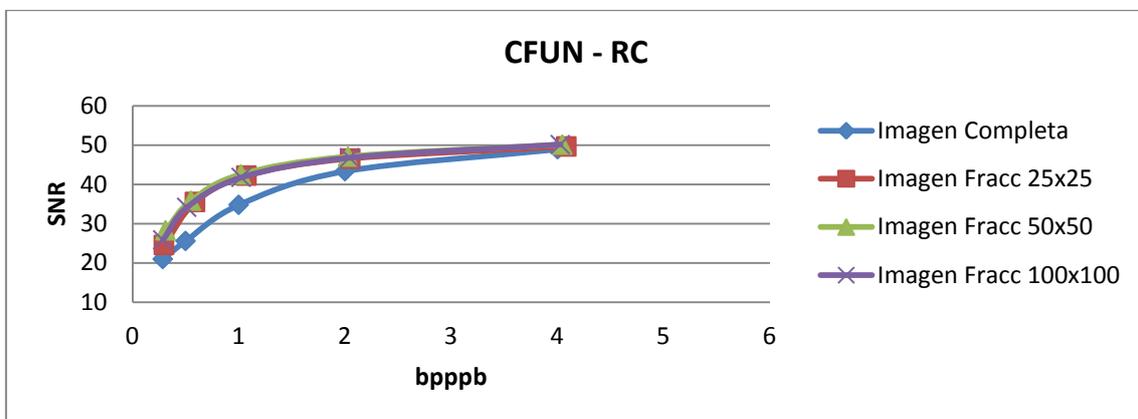


Gráfica A.2.1.1.3: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GLMMSE

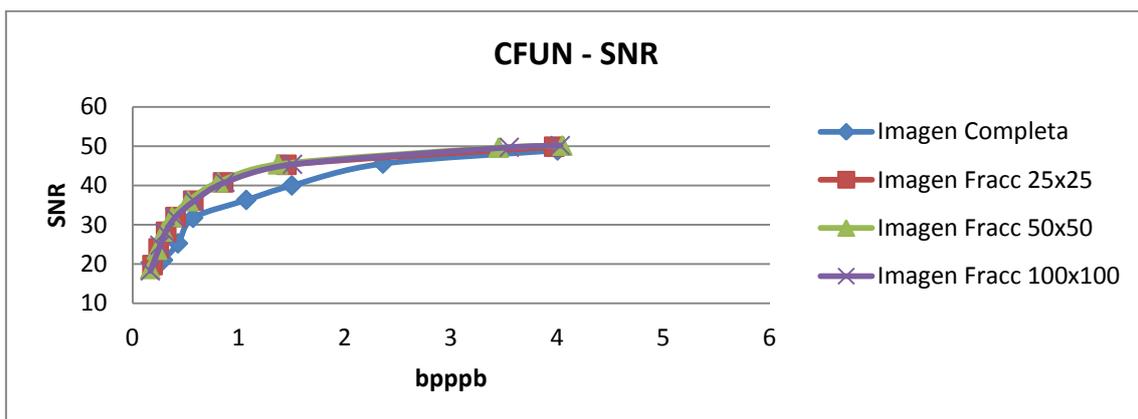


Gráfica A.2.1.1.4: Comparación de los mejores resultados obtenidos para cada uno de los algoritmos de Data Fusion, tras ser comprimida la imagen con el compresor CDF

A.2.1.2.- Compresor CFUN

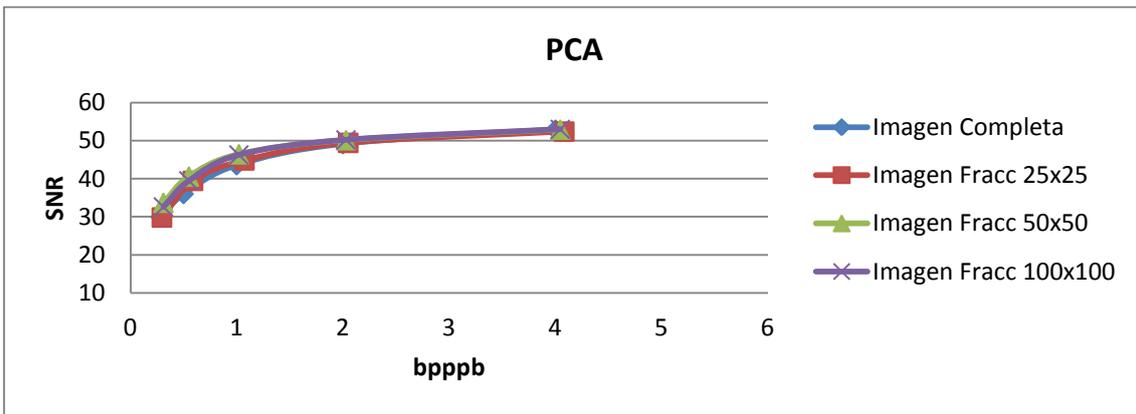


Gráfica A.2.1.2.1: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

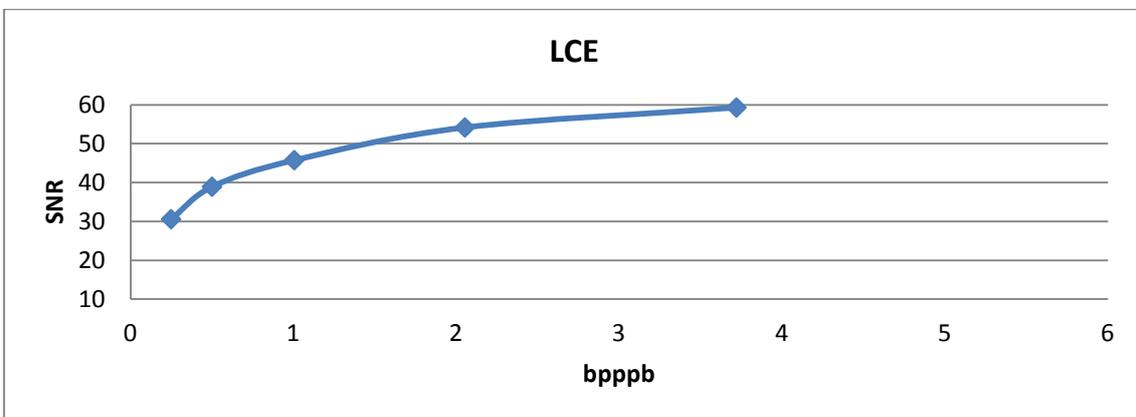


Gráfica A.2.1.2.2: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

A.2.1.3.- Algoritmos de comparación

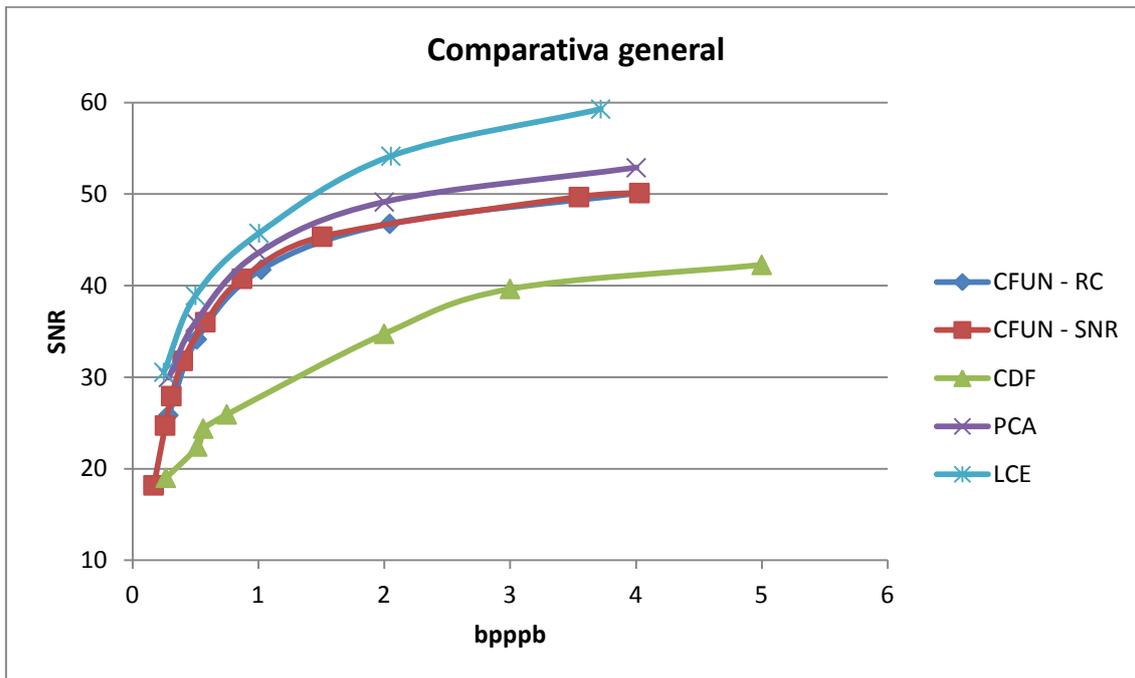


Gráfica A.2.1.3.1: Resultados de la compresión usando el PCA con diferentes configuraciones de la imagen (entera o macrobloques)



Gráfica A.2.1.3.2: Resultados de la compresión usando el LCE

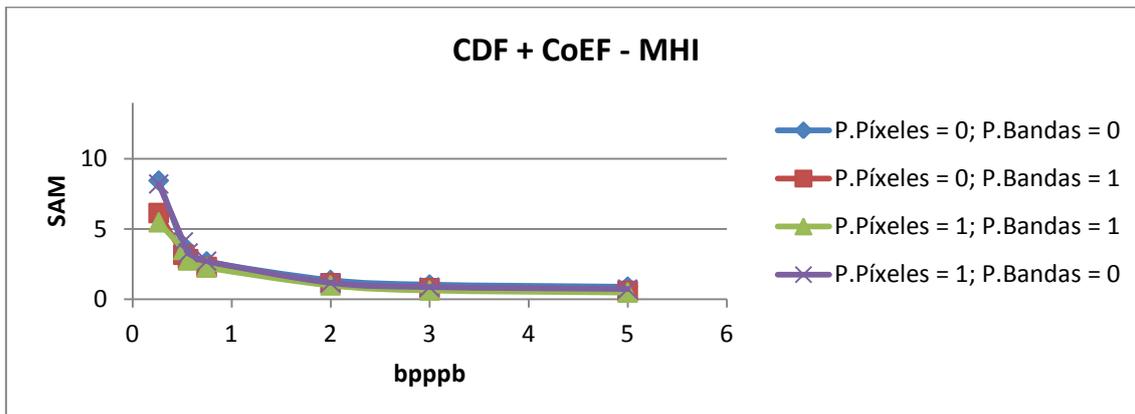
A.2.1.4.- Comparativa general



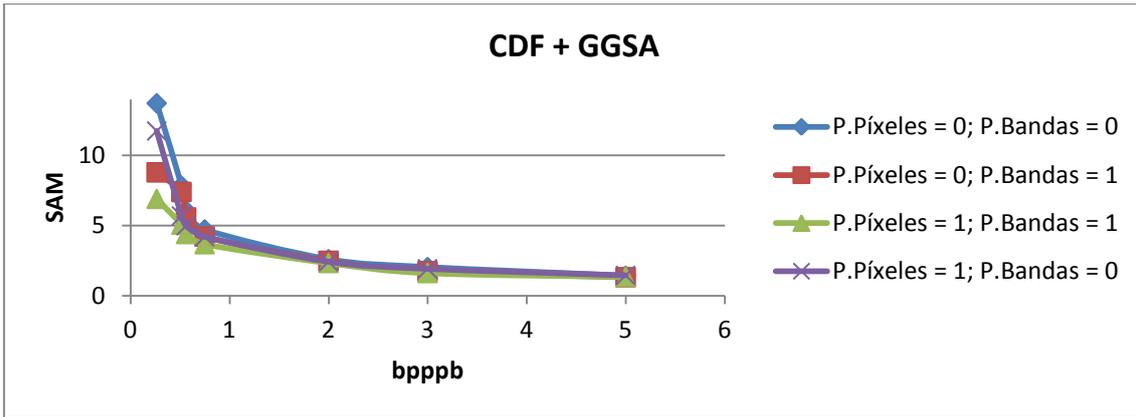
Gráfica A.2.1.4.1: Comparativa general entre los resultados de los algoritmos del estado del arte de compresión y los mejores resultados conseguidos con las metodologías propuestas

A.2.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS. Métrica: SAM

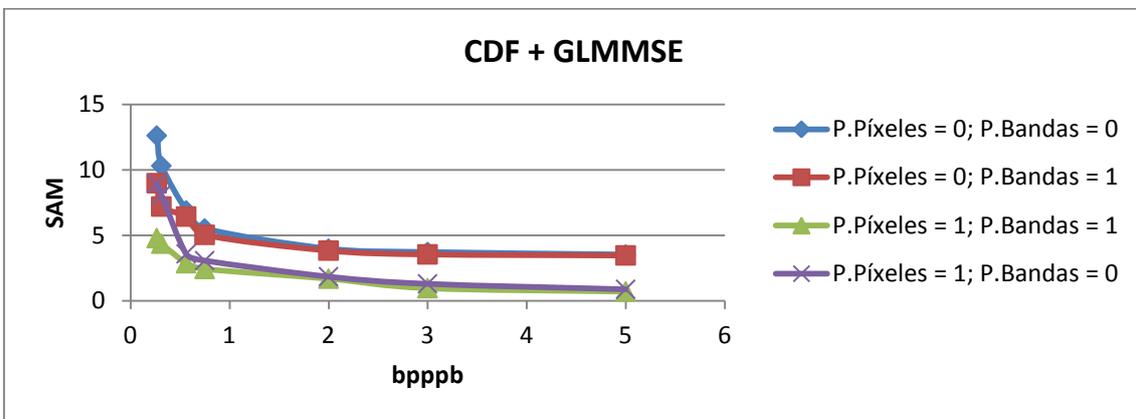
A.2.2.1.- Compresor CDF + Data Fusion



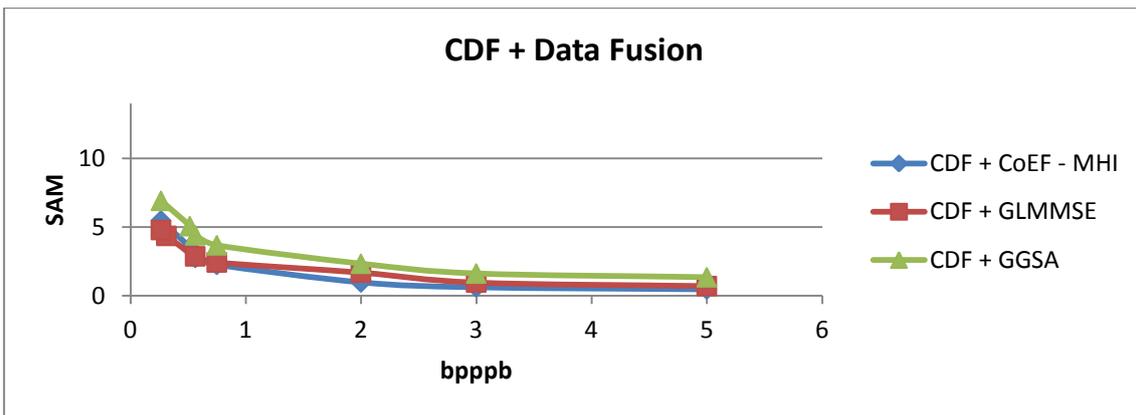
Gráfica A.2.2.1.1: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo CoEF - MHI



Gráfica A.2.2.1.2: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GGSA

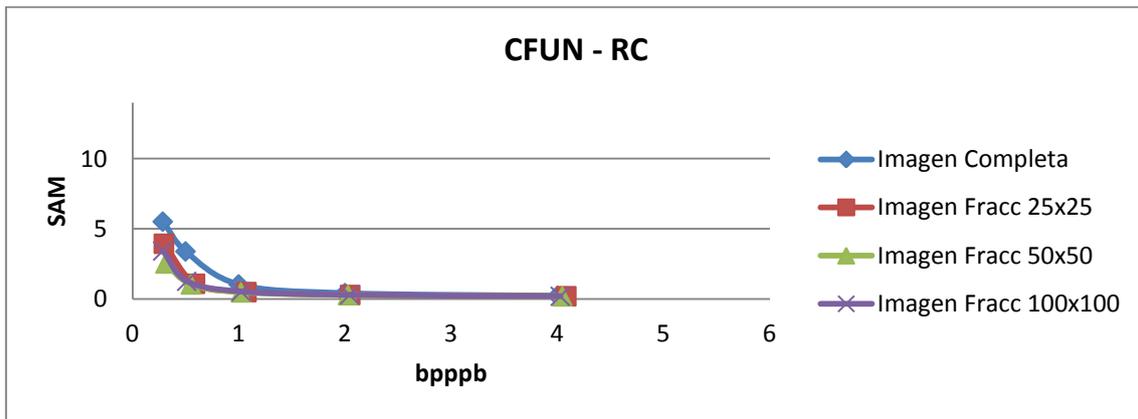


Gráfica A.2.2.1.3: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GLMMSE

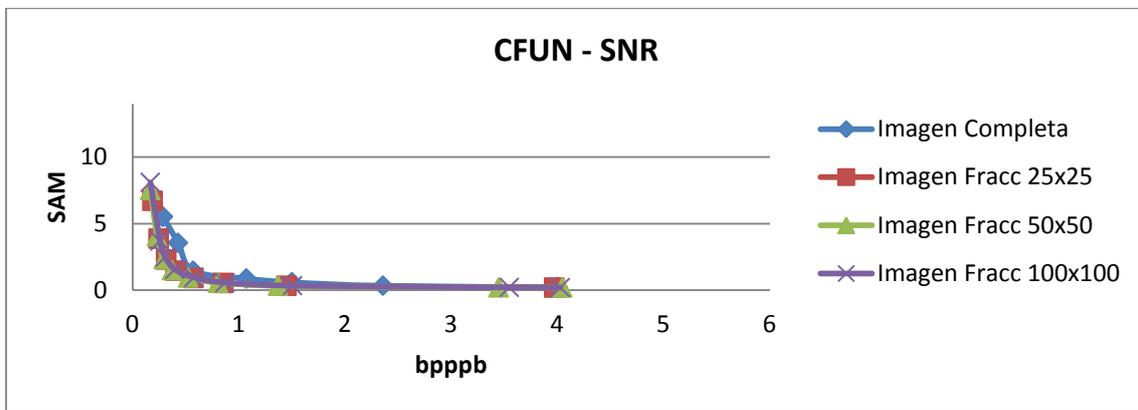


Gráfica A.2.2.1.4: Comparación de los mejores resultados obtenidos para cada uno de los algoritmos de Data Fusion, tras ser comprimida la imagen con el compresor CDF

A.2.2.2.- Compresor CFUN

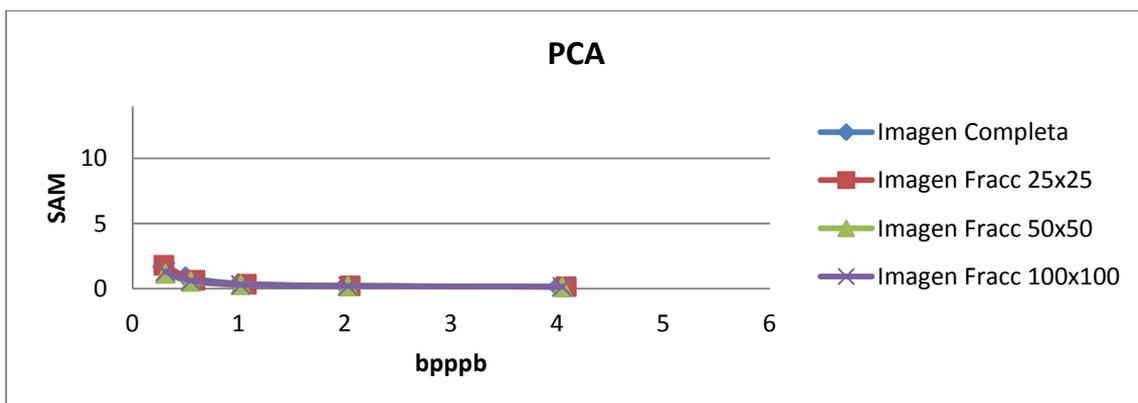


Gráfica A.2.2.2.1: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

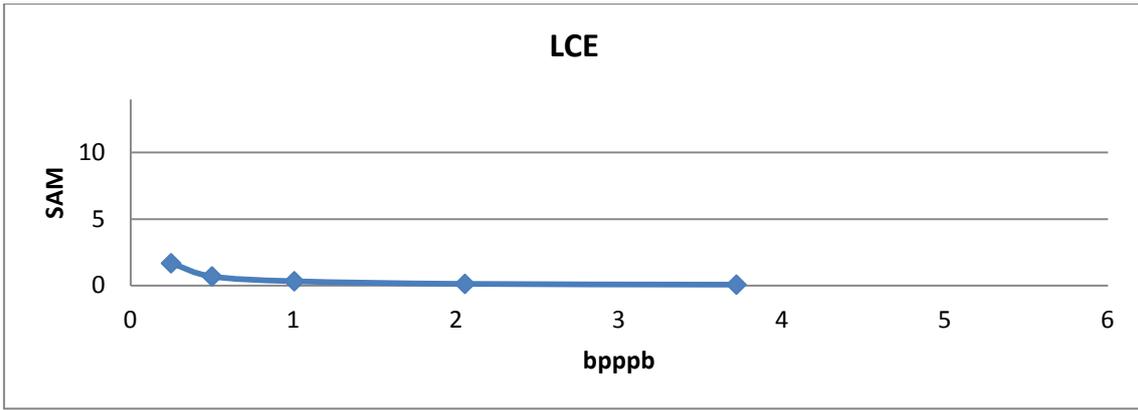


Gráfica A.2.2.2.2: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

A.2.2.3.- Algoritmos de comparación

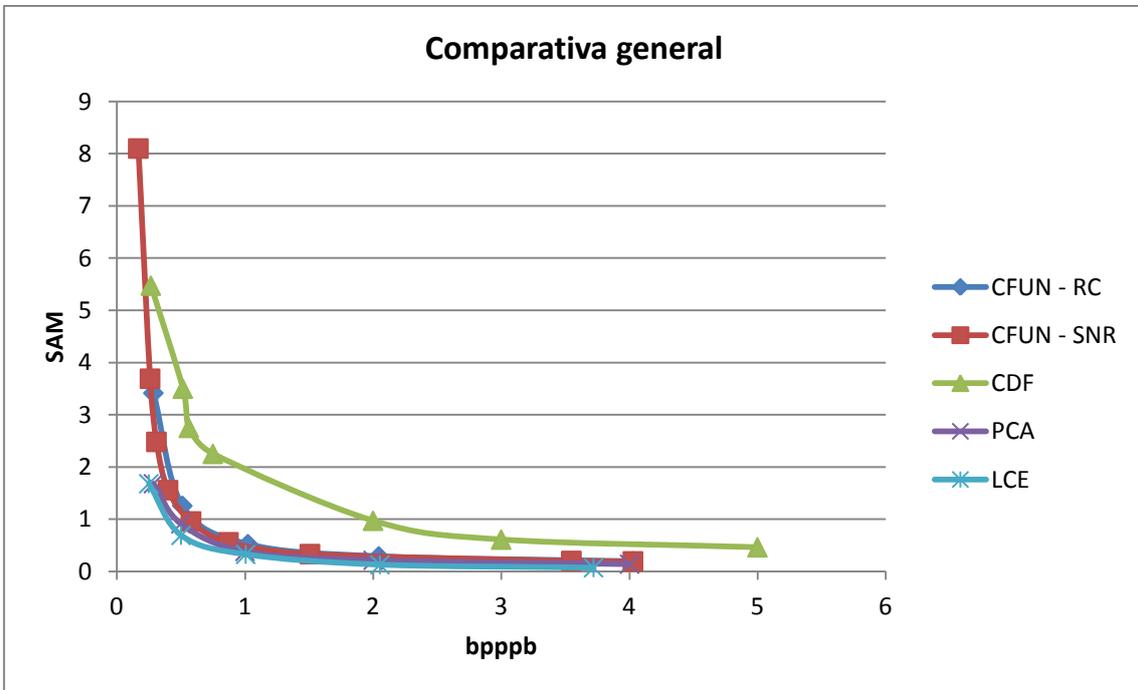


Gráfica A.2.2.3.1: Resultados de la compresión usando el PCA con diferentes configuraciones de la imagen (entera o macrobloques)



Gráfica A.2.2.3.2: Resultados de la compresión usando el LCE

A.2.2.4.- Comparativa general

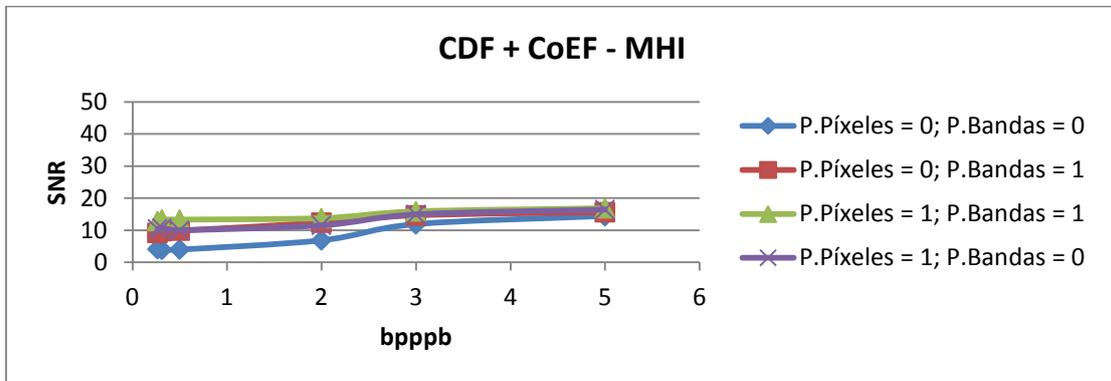


Gráfica A.2.2.4.1: Comparativa general entre los resultados de los algoritmos del estado del arte de compresión y los mejores resultados conseguidos con las metodologías propuestas

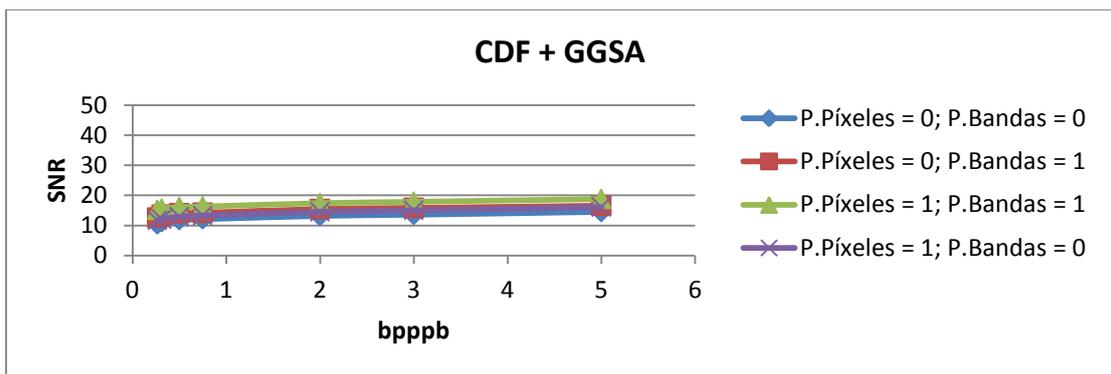
A.3.- Anexo III

A.3.1.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM. Métrica: SNR

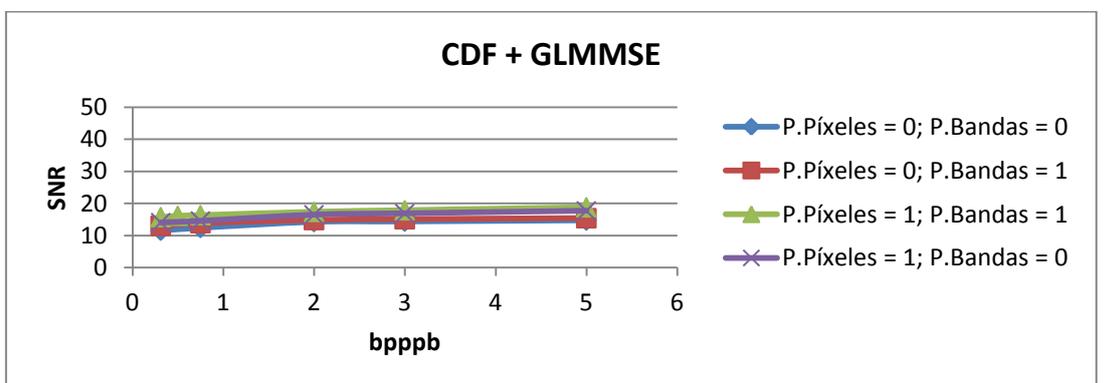
A.3.1.1.- Compresor CDF + Data Fusion



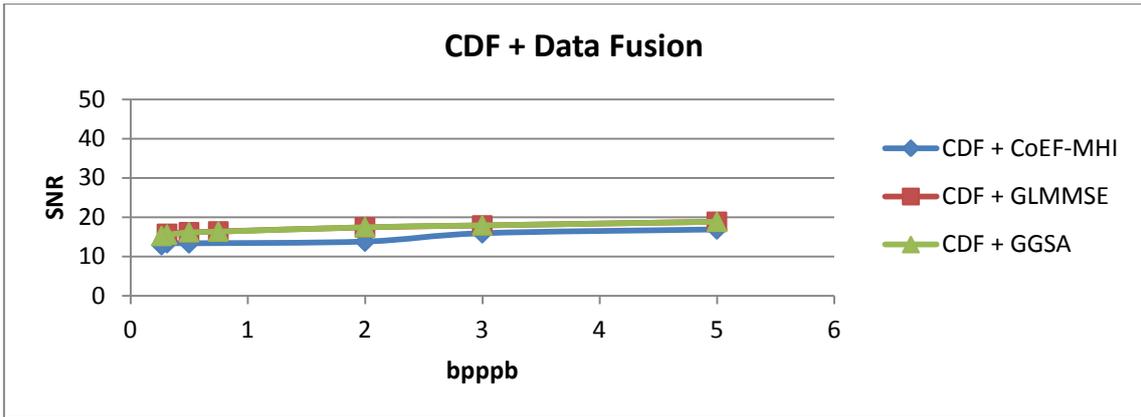
Gráfica A.3.1.1.1: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo CoEF - MHI



Gráfica A.3.1.1.2: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GGSA

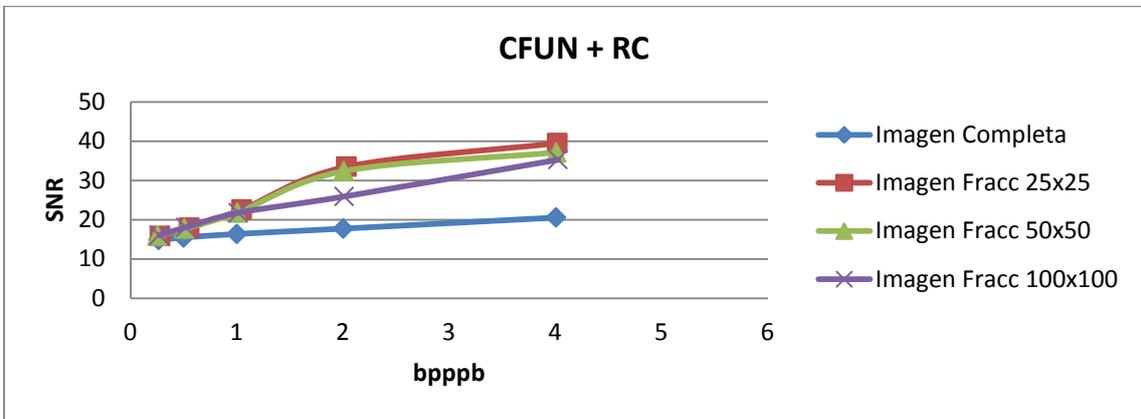


Gráfica A.3.1.1.3: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GLMMSE

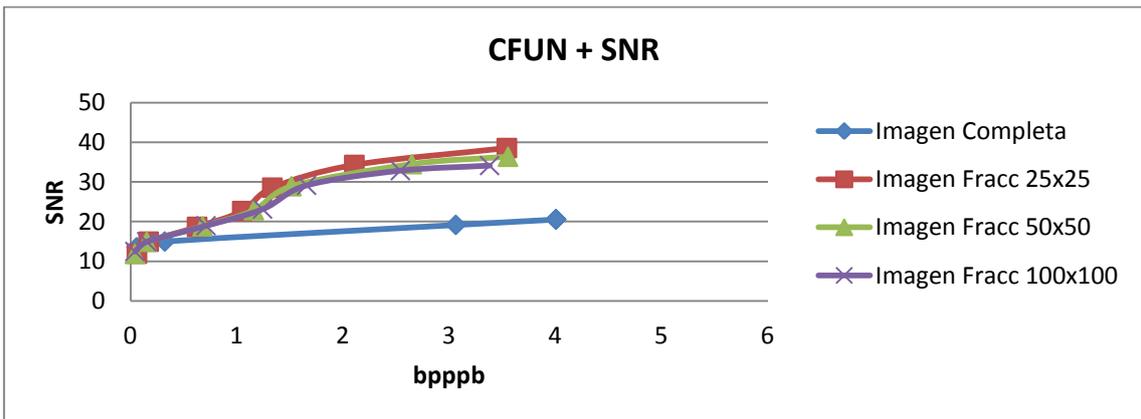


Gráfica A.3.1.1.4: Comparación de los mejores resultados obtenidos para cada uno de los algoritmos de Data Fusion, tras ser comprimida la imagen con el compresor CDF

A.3.1.2.- Compresor CFUN

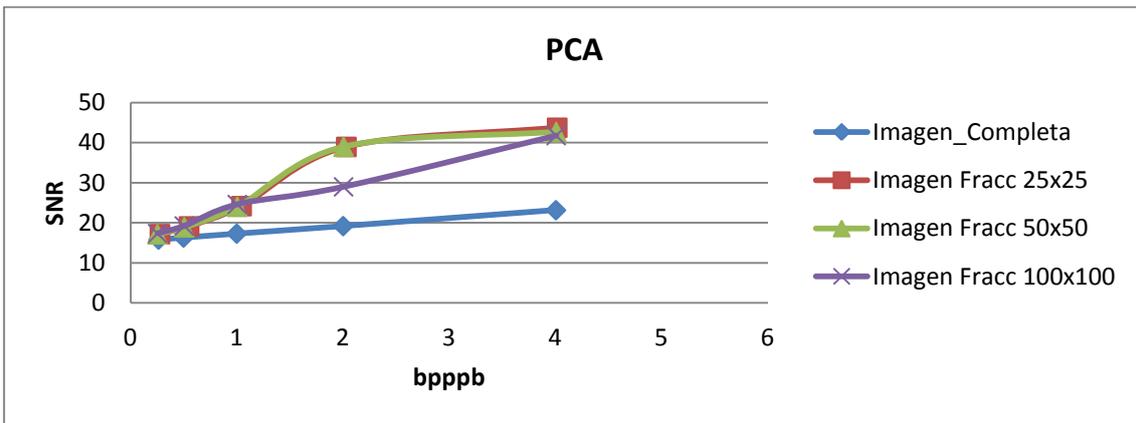


Gráfica A.3.1.2.1: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

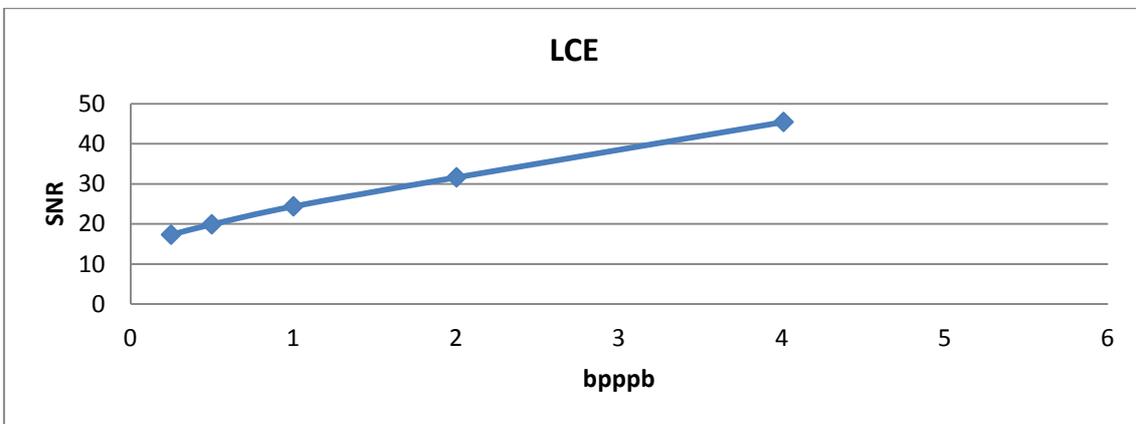


Gráfica A.3.1.2.2: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

A.3.1.3.- Algoritmos de comparación

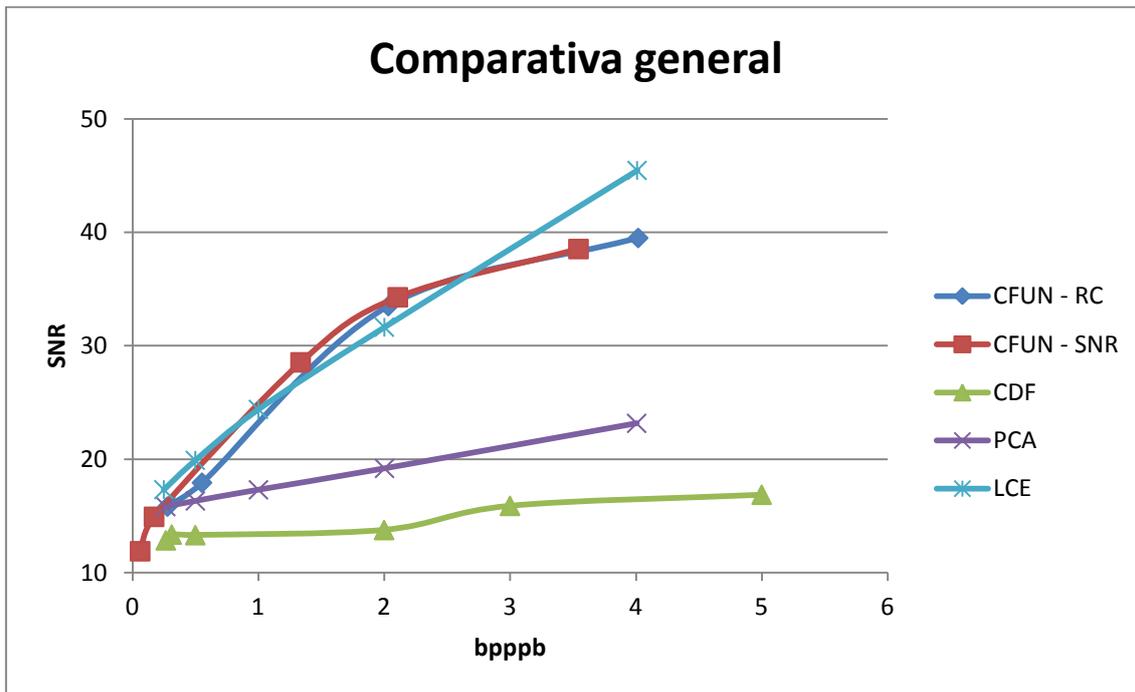


Gráfica A.3.1.3.1: Resultados de la compresión usando el PCA con diferentes configuraciones de la imagen (entera o macrobloques)



Gráfica A.3.1.3.2: Resultados de la compresión usando el LCE

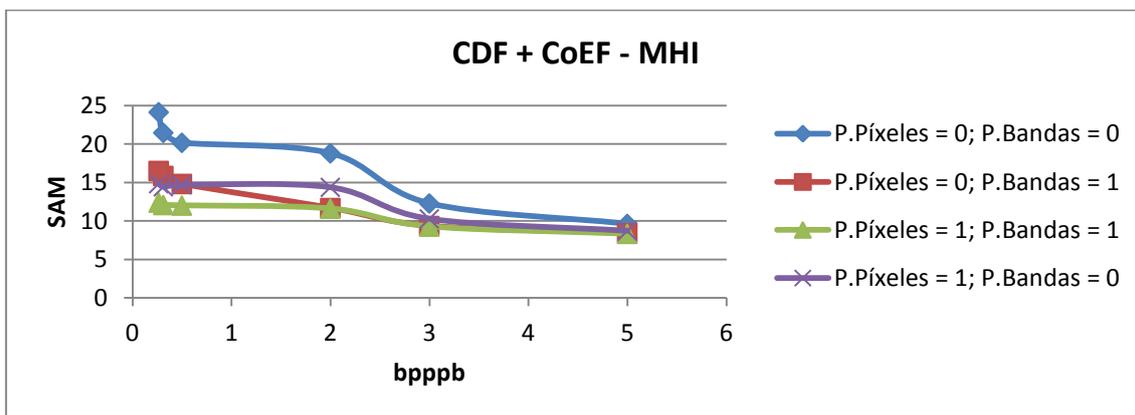
A.3.1.4.- Comparativa general



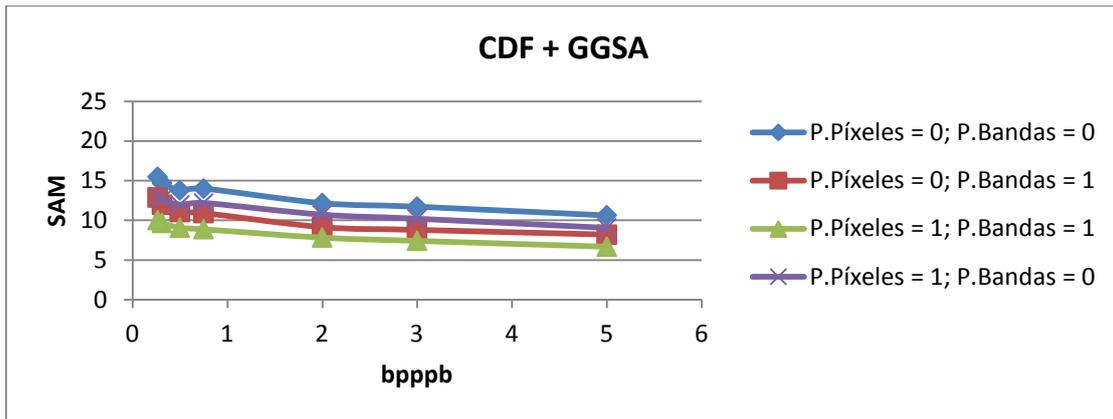
Gráfica A.3.1.4.1: Comparativa general entre los resultados de los algoritmos del estado del arte de compresión y los mejores resultados conseguidos con las metodologías propuestas

A.3.2.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM. Métrica: SAM

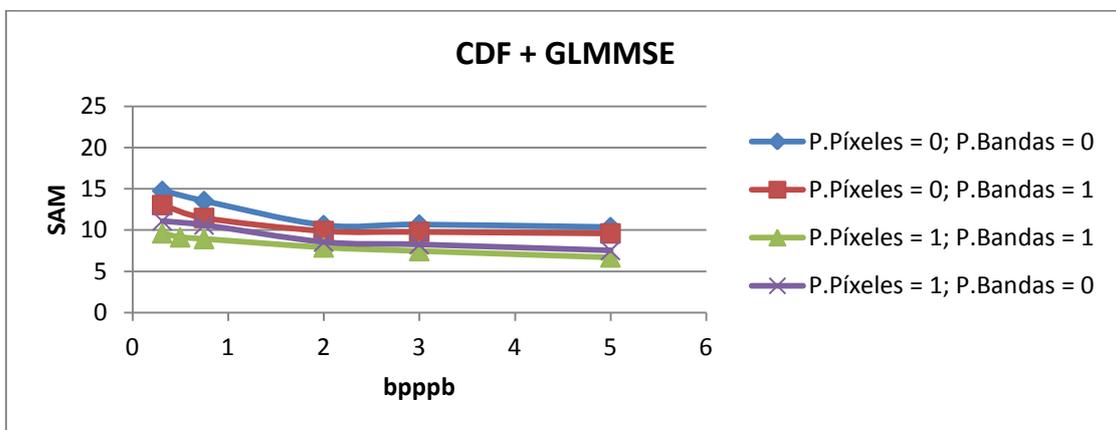
A.3.2.1.- Compresor CDF + Data Fusion



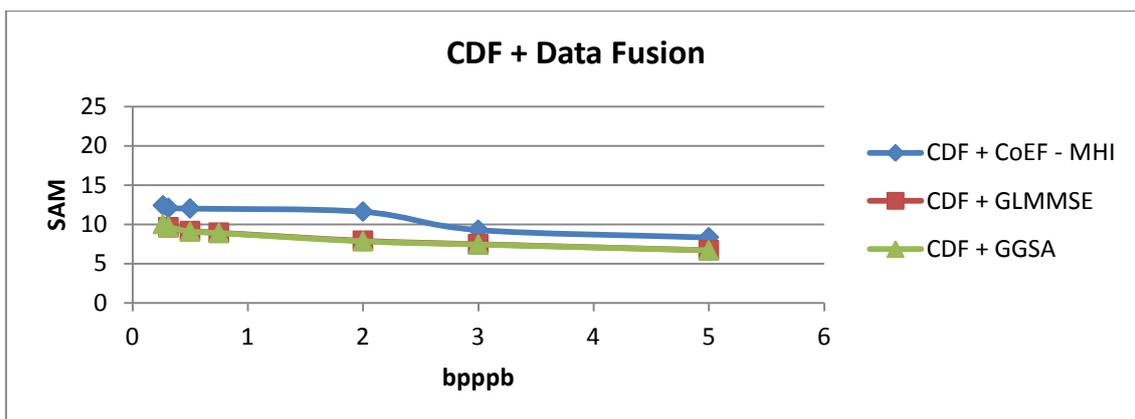
Gráfica A.3.2.1.1: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo CoEF - MHI



Gráfica A.3.2.1.2: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GGSA

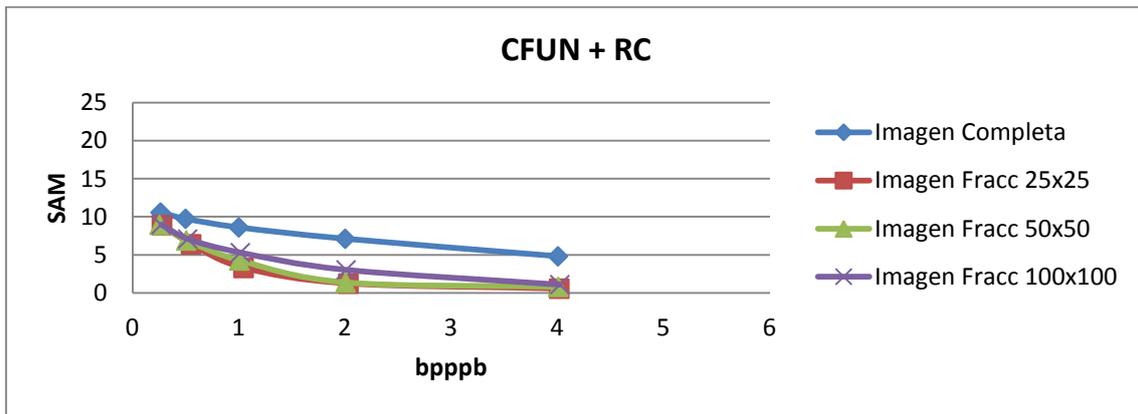


Gráfica A.3.2.1.3: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GLMMSE

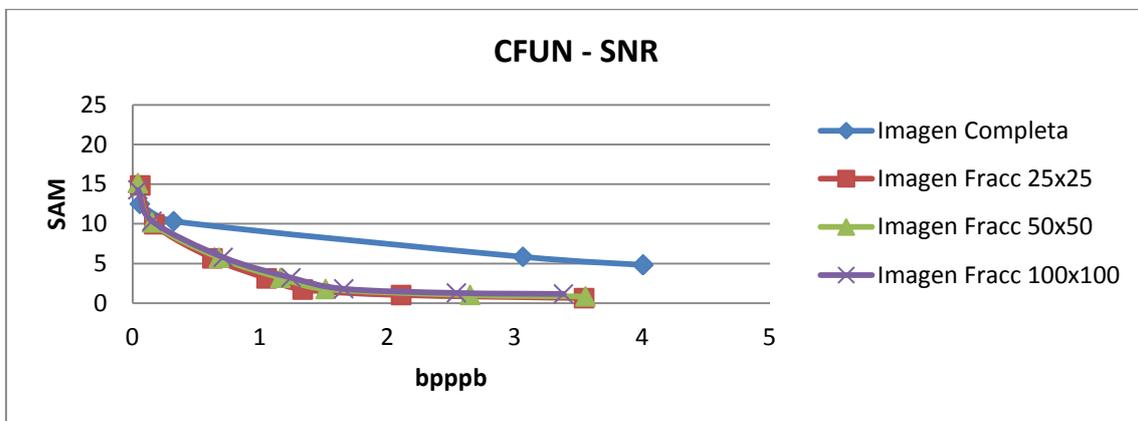


Gráfica A.3.2.1.4: Comparación de los mejores resultados obtenidos para cada uno de los algoritmos de Data Fusion, tras ser comprimida la imagen con el compresor CDF

A.3.2.2.- Compresor CFUN

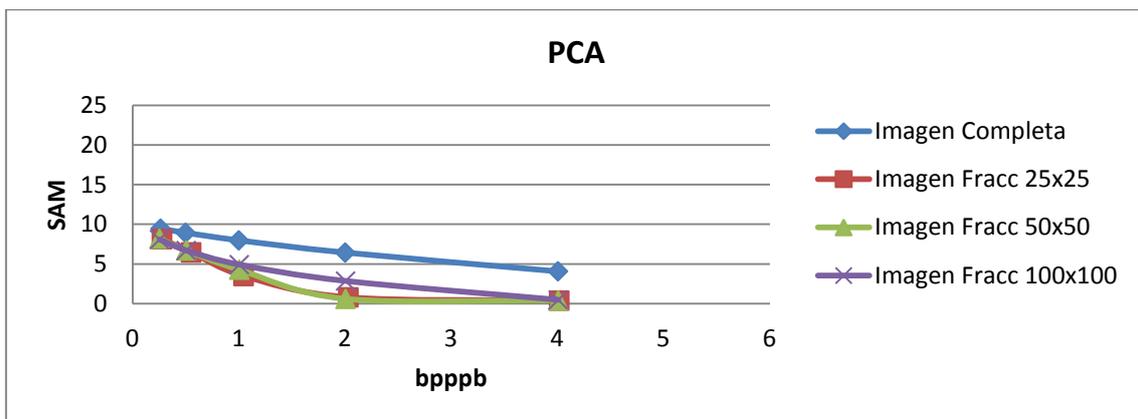


Gráfica A.3.2.2.1: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

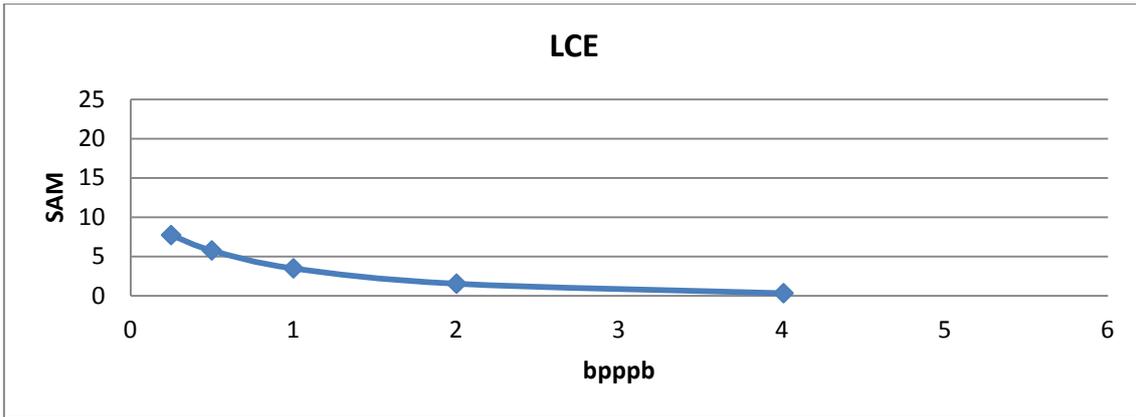


Gráfica A.3.2.2.2: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

A.3.2.3.- Algoritmos de comparación

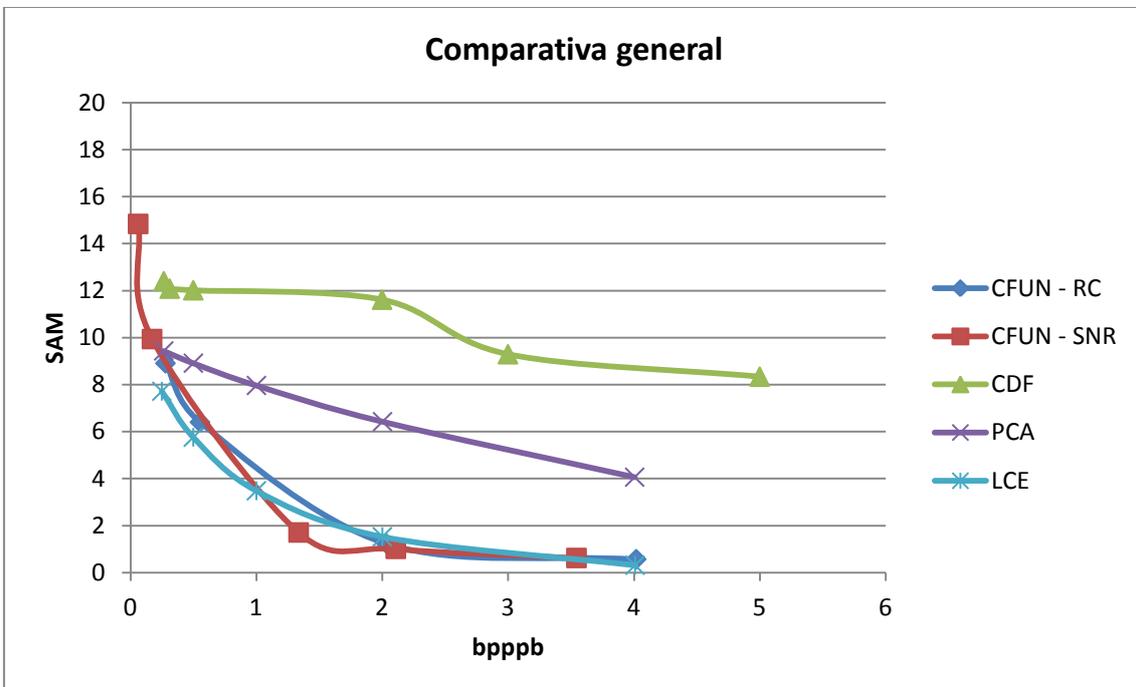


Gráfica A.3.2.3.1: Resultados de la compresión usando el PCA con diferentes configuraciones de la imagen (entera o macrobloques)



Gráfica A.3.2.3.2: Resultados de la compresión usando el LCE

A.3.2.4.- Comparativa general

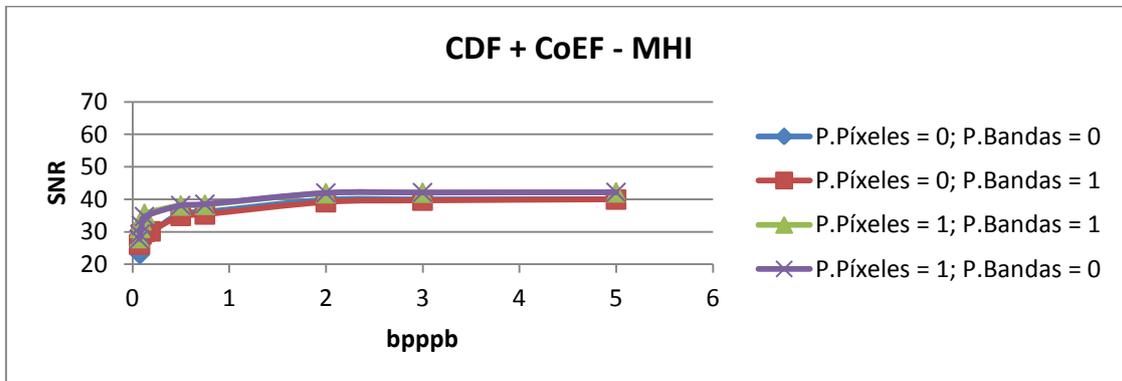


Gráfica A.3.2.4.1: Comparativa general entre los resultados de los algoritmos del estado del arte de compresión y los mejores resultados conseguidos con las metodologías propuestas

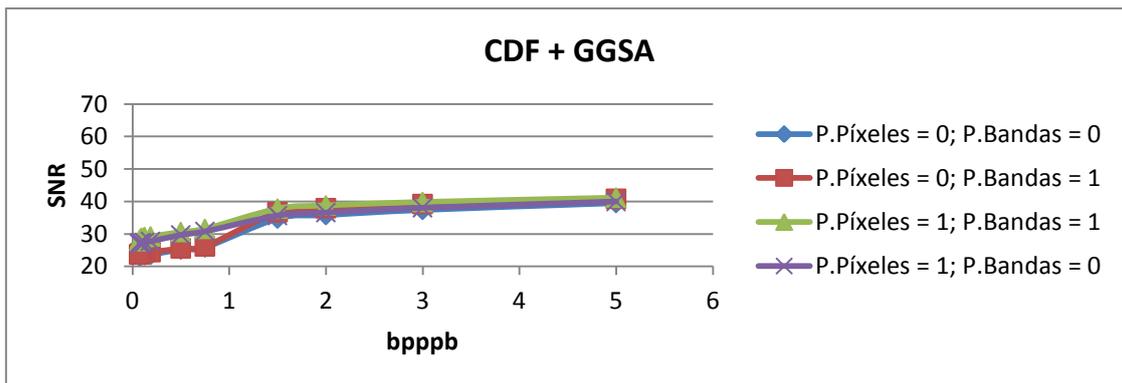
A.4.- Anexo IV

A.4.1.- Gráficas de resultados de las simulaciones para la imagen del sensor IASI. Métrica: SNR

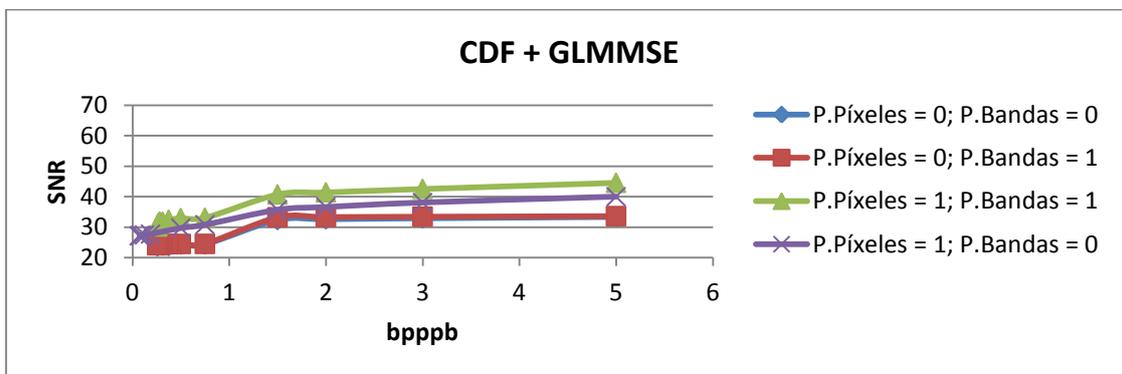
A.4.1.1.- Compresor CDF + Data Fusion



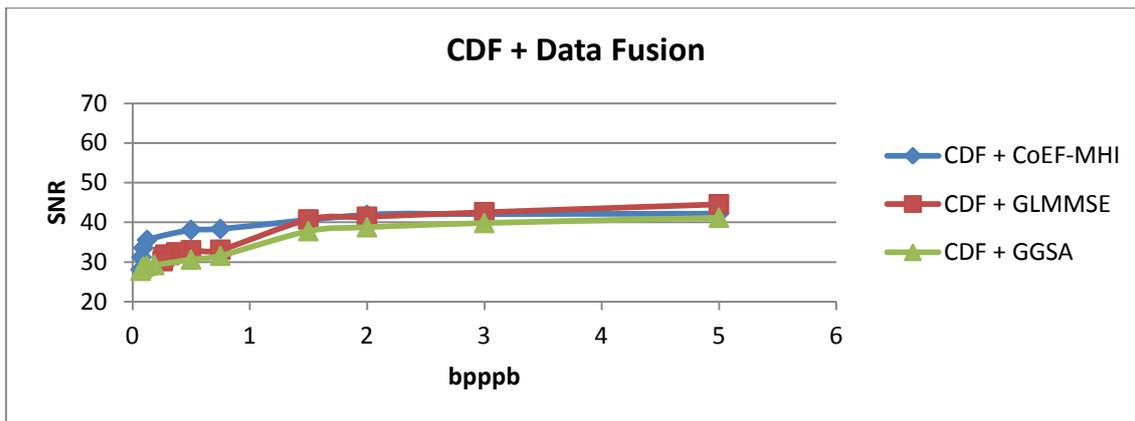
Gráfica A.4.1.1.1: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo CoEF - MHI



Gráfica A.4.1.1.2: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GGSA

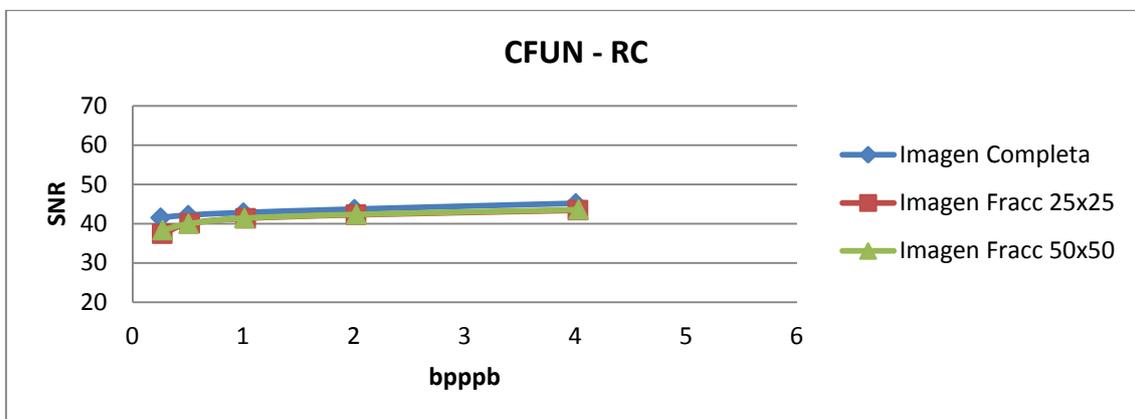


Gráfica A.4.1.1.3: Combinaciones de las posibles compresiones espectral y espacial del compresor CDF. Descompresión usando algoritmo GLMMSE

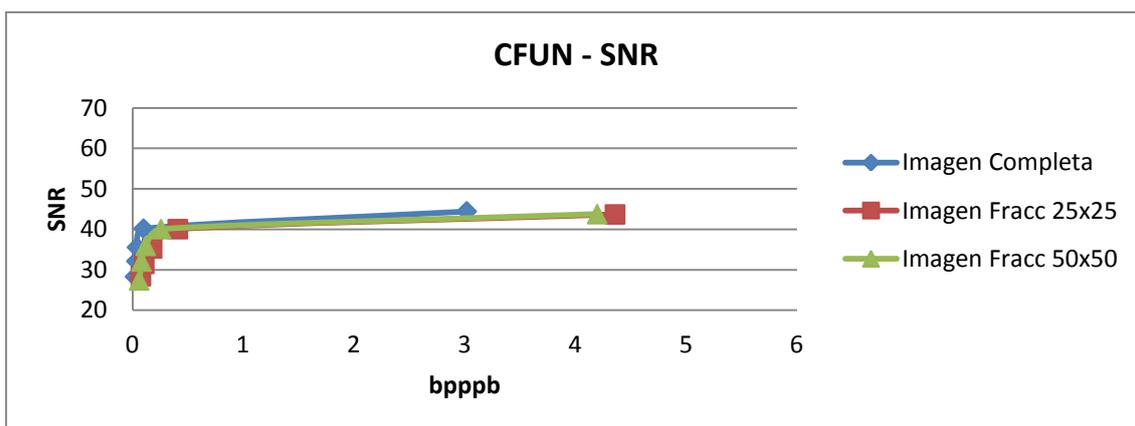


Gráfica A.4.1.1.4: Comparación de los mejores resultados obtenidos para cada uno de los algoritmos de Data Fusion, tras ser comprimida la imagen con el compresor CDF

A.4.1.2.- Compresor CFUN

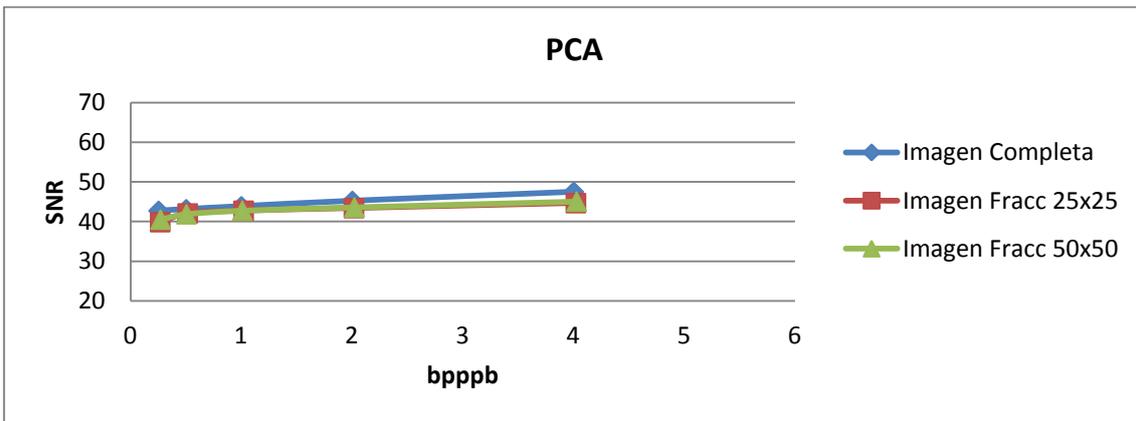


Gráfica A.4.1.2.1: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

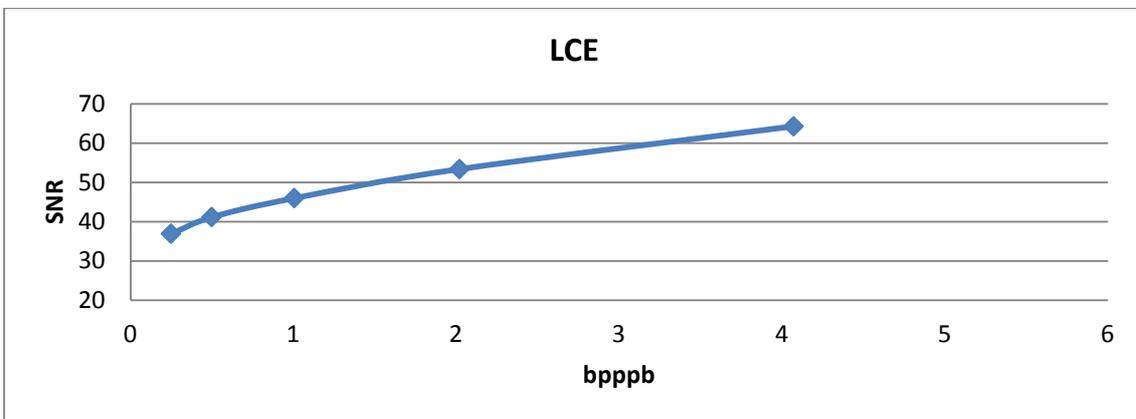


Gráfica A.4.1.2.2: Resultados del compresor CFUN con diferentes configuraciones de la imagen (entera o macrobloques)

A.4.1.3.- Algoritmos de comparación

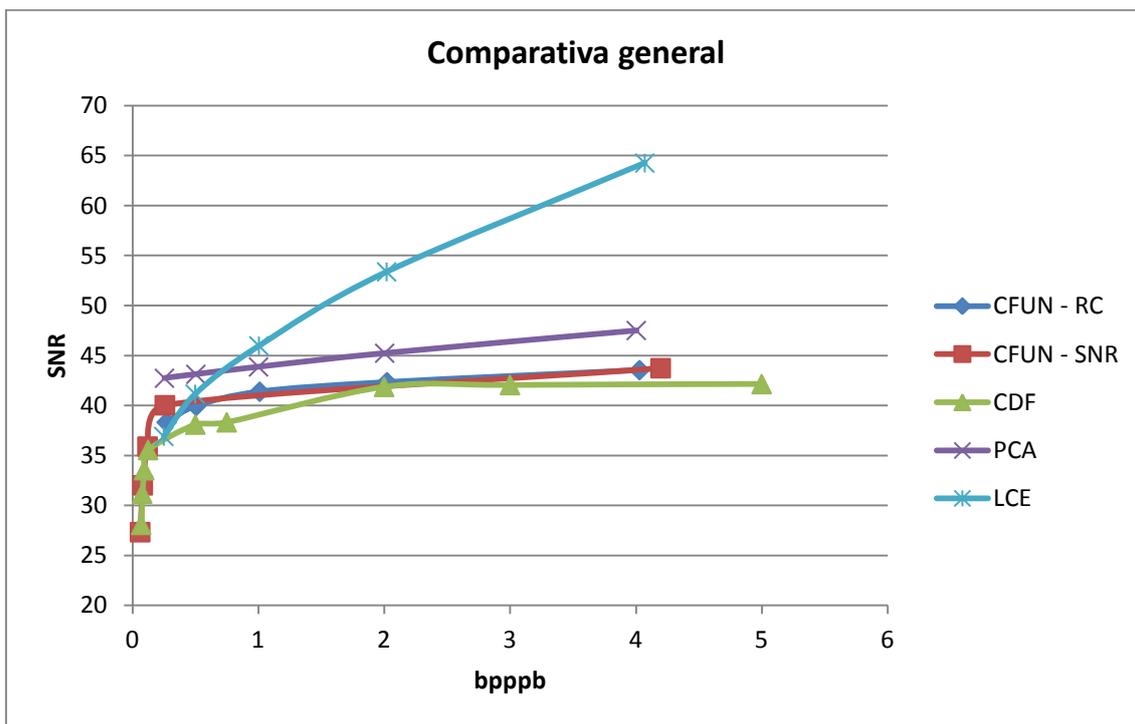


Gráfica A.4.1.3.1: Resultados de la compresión usando el PCA con diferentes configuraciones de la imagen (entera o macrobloques)



Gráfica A.4.1.3.2: Resultados de la compresión usando el LCE

A.4.1.4.- Comparativa general

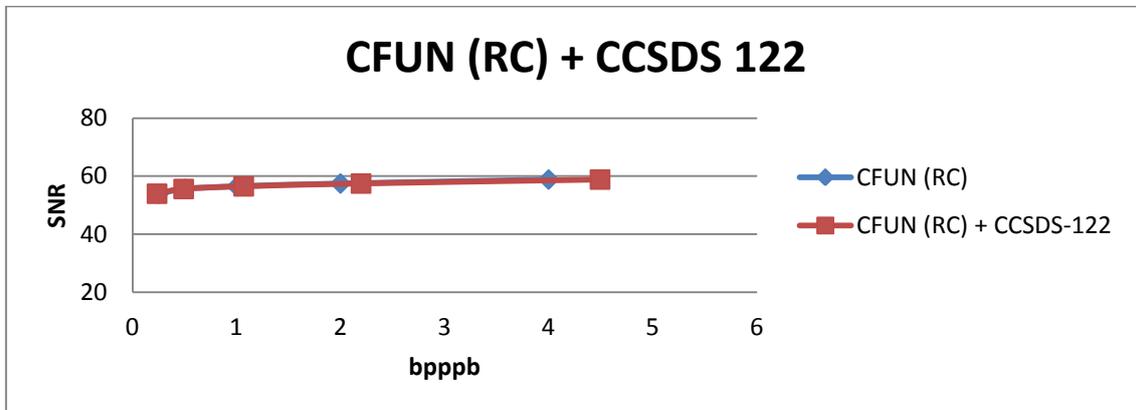


Gráfica A.4.1.4.1: Comparativa general entre los resultados de los algoritmos del estado del arte de compresión y los mejores resultados conseguidos con las metodologías propuestas

A.5.- Anexo V

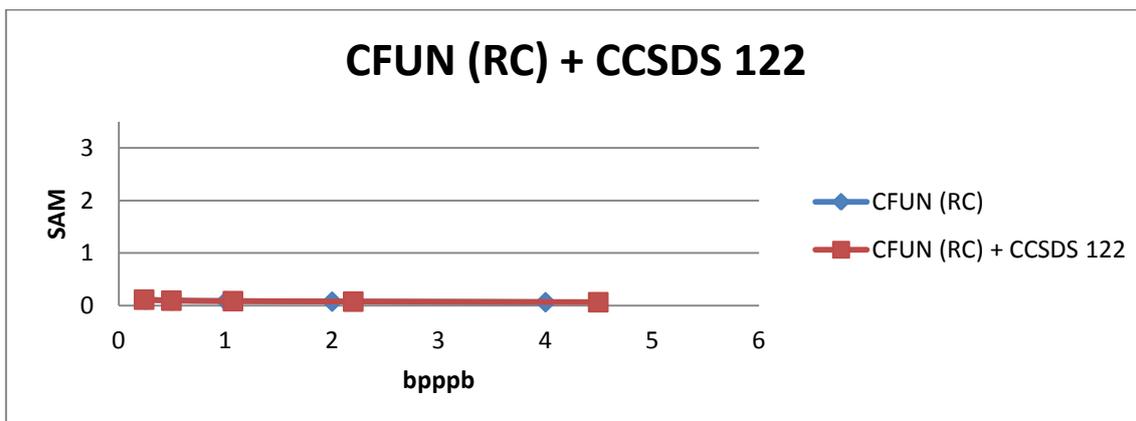
A.5.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS CDF+CCSDS-122

A.5.1.1.- Métrica SNR – CFUN (RC)



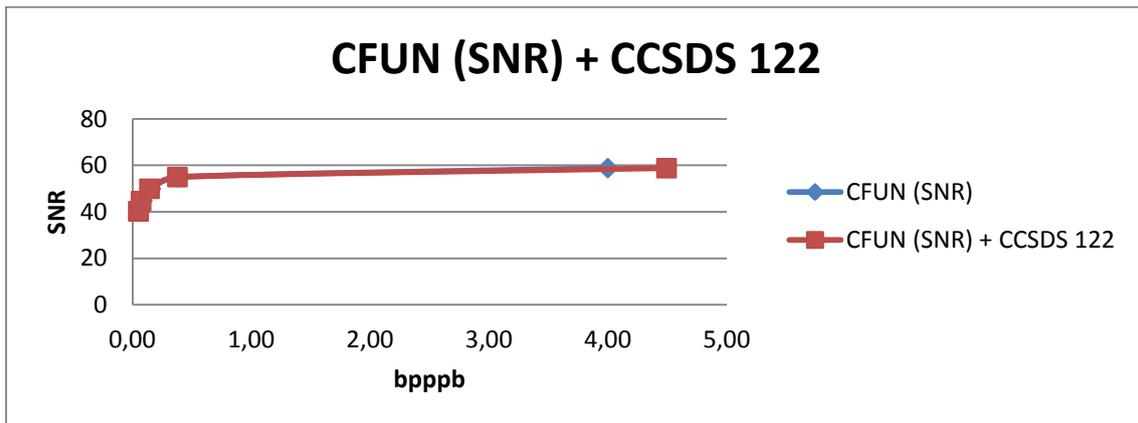
Gráfica A.5.1.1.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.5.1.2.- Métrica SAM – CFUN (RC)



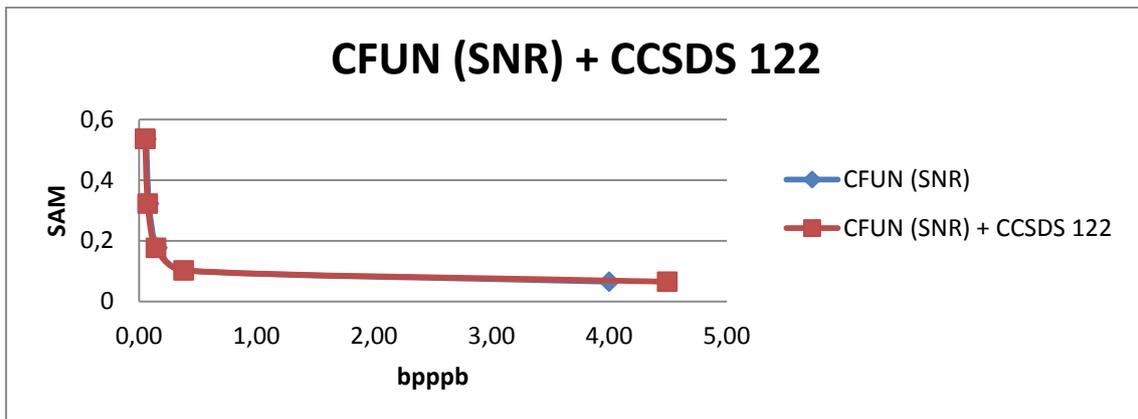
Gráfica A.5.1.2.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.5.1.3.- Métrica SNR – CFUN (SNR)



Gráfica A.5.1.3.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

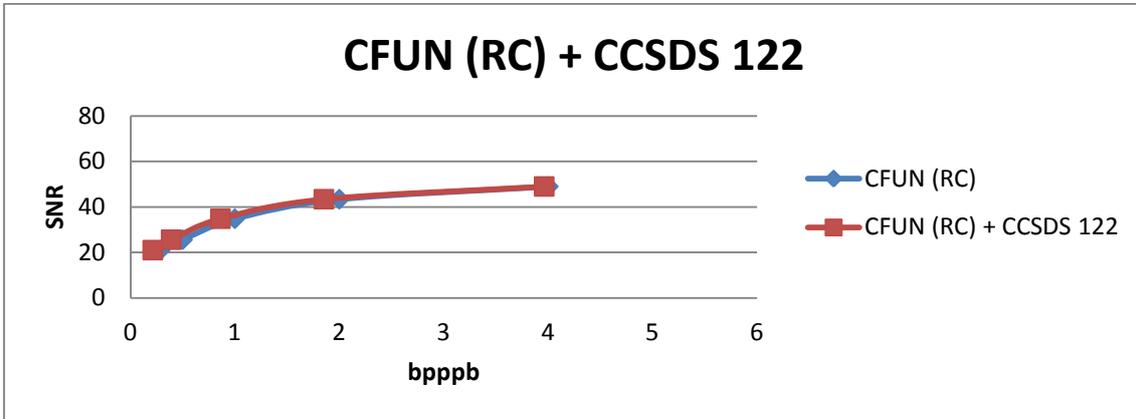
A.5.1.4.- Métrica SAM – CFUN (SNR)



Gráfica A.5.1.4.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

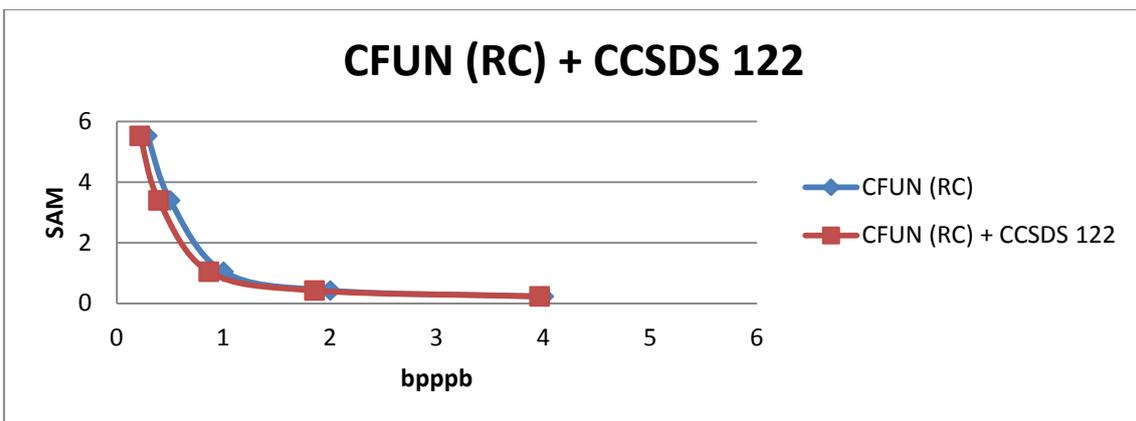
A.5.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS CDF+CCSDS-122

A.5.2.1.- Métrica SNR – CFUN (RC)



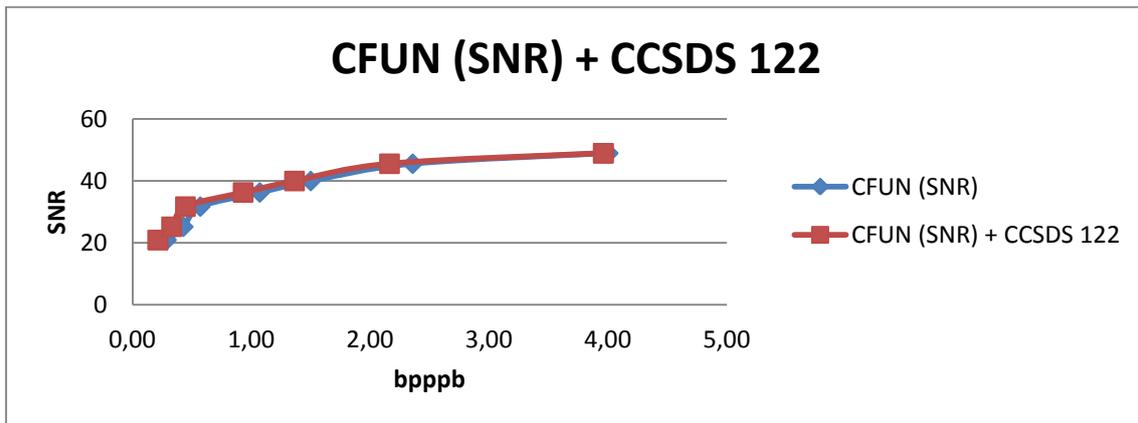
Gráfica A.5.2.1.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.5.2.2.- Métrica SAM – CFUN (RC)



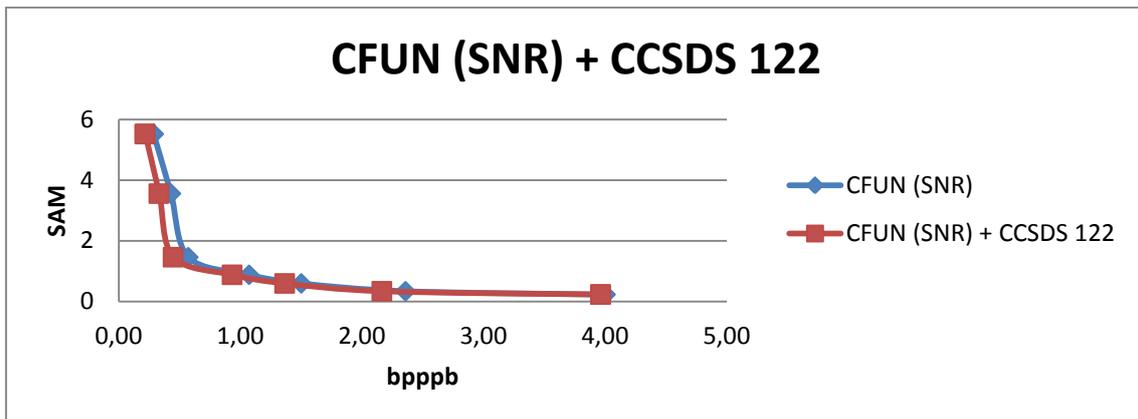
Gráfica A.5.2.2.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.5.2.3.- Métrica SNR – CFUN (SNR)



Gráfica A.5.2.3.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

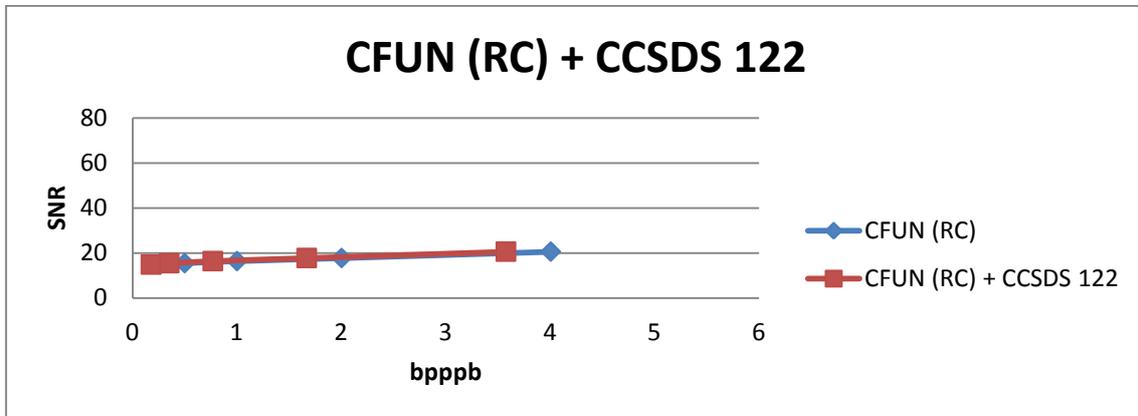
A.5.2.4.- Métrica SAM – CFUN (SNR)



Gráfica A.5.2.4.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

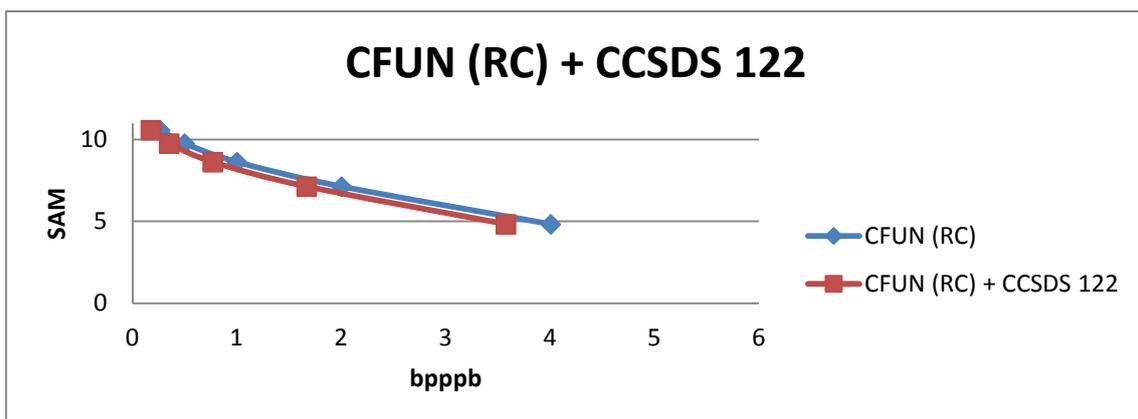
A.5.3.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM CDF+CCSDS-122

A.5.3.1.- Métrica SNR – CFUN (RC)



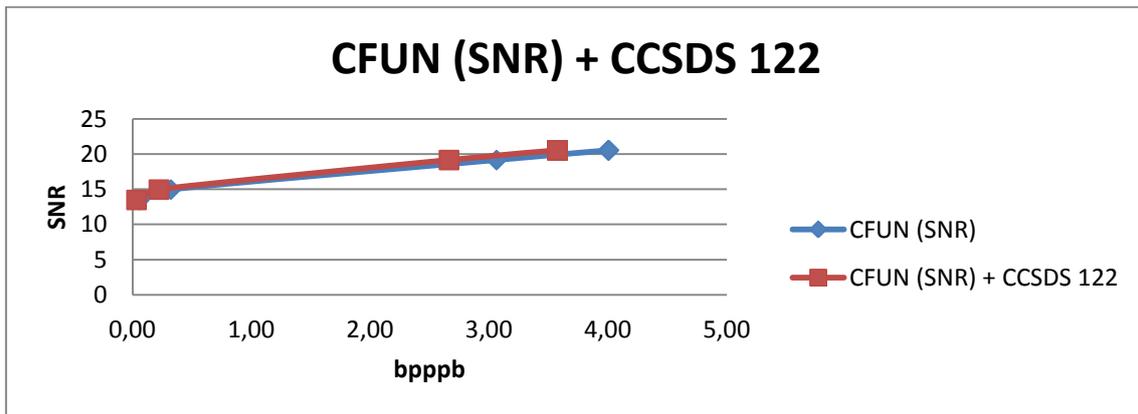
Gráfica A.5.3.1.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.5.3.2.- Métrica SAM – CFUN (RC)



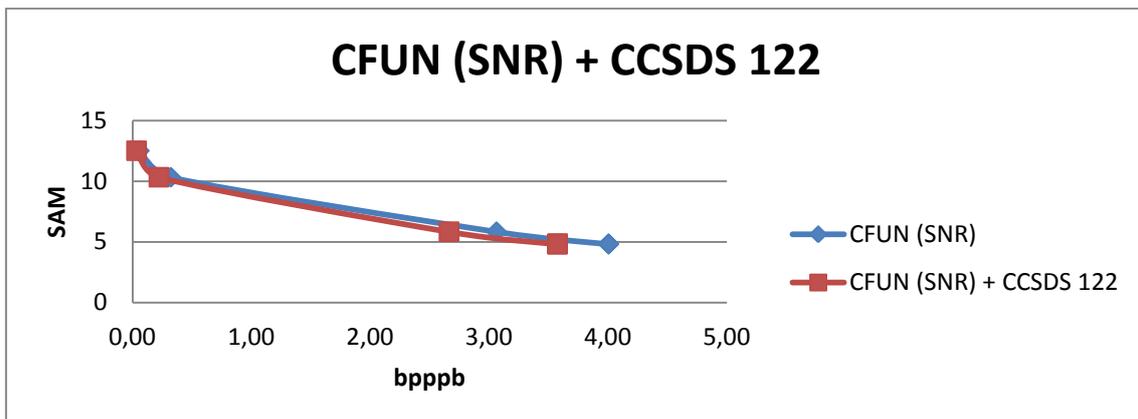
Gráfica A.5.3.2.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.5.2.3.- Métrica SNR – CFUN (SNR)



Gráfica A.5.2.3.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

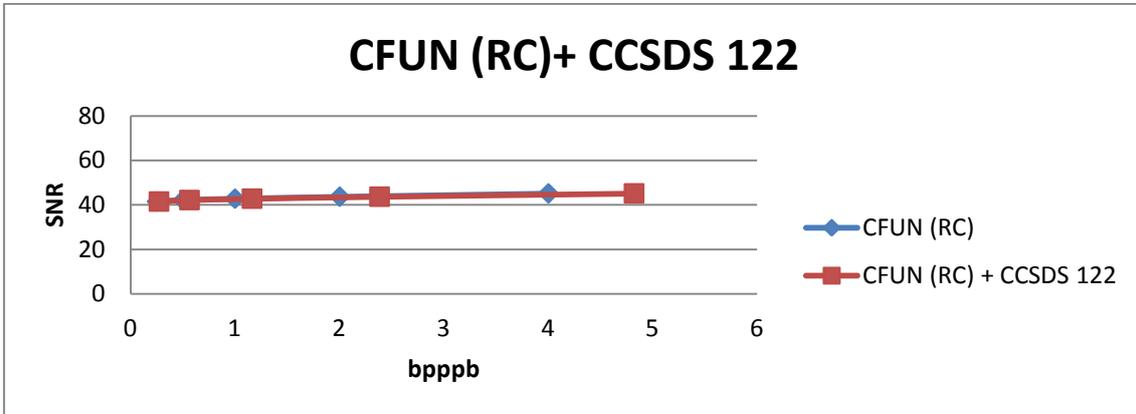
A.5.2.4.- Métrica SAM – CFUN (SNR)



Gráfica A.5.2.4.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

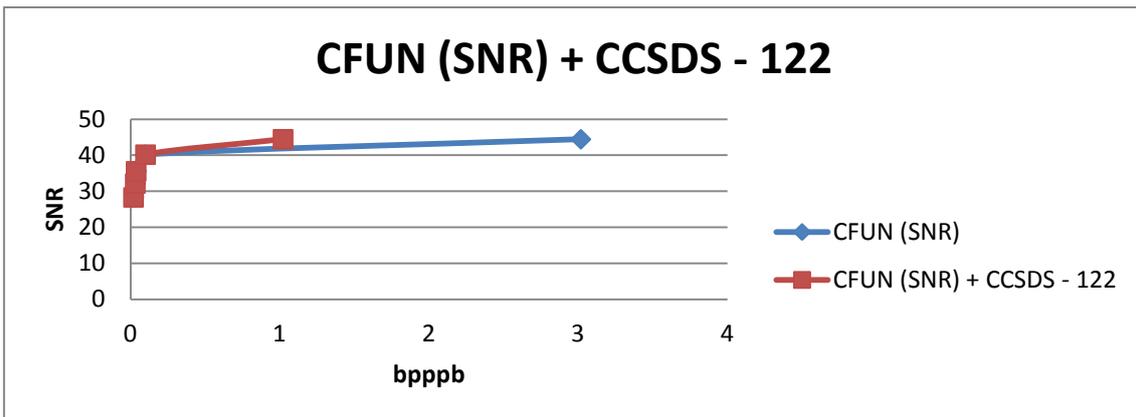
A.5.4.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM CDF+CCSDS-122

A.5.4.1.- Métrica SNR – CFUN (RC)



Gráfica A.5.4.1.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.5.4.2.- Métrica SNR – CFUN (SNR)

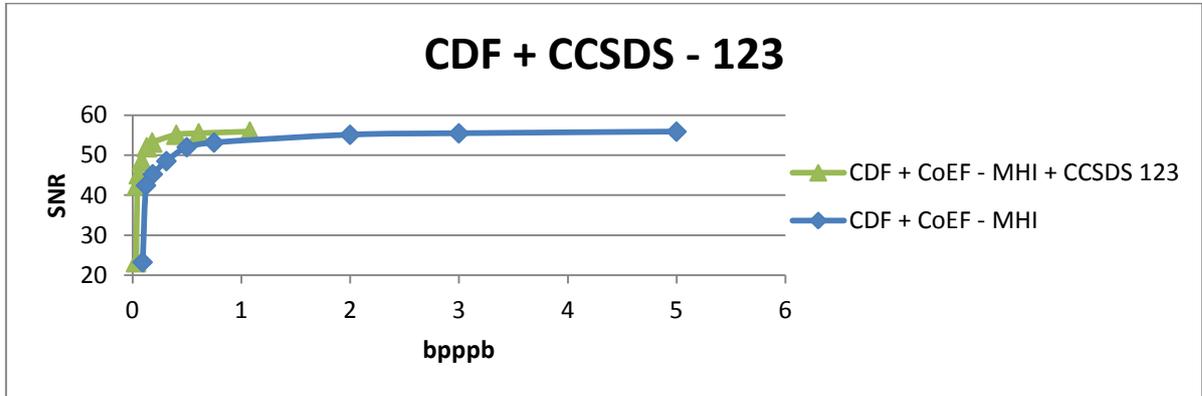


Gráfica A.5.4.2.1: Comparativa entre los resultados del compresor CFUN y el del mismo compresor tras aplicar a la imagen el CCSDS-122

A.6.- Anexo VI

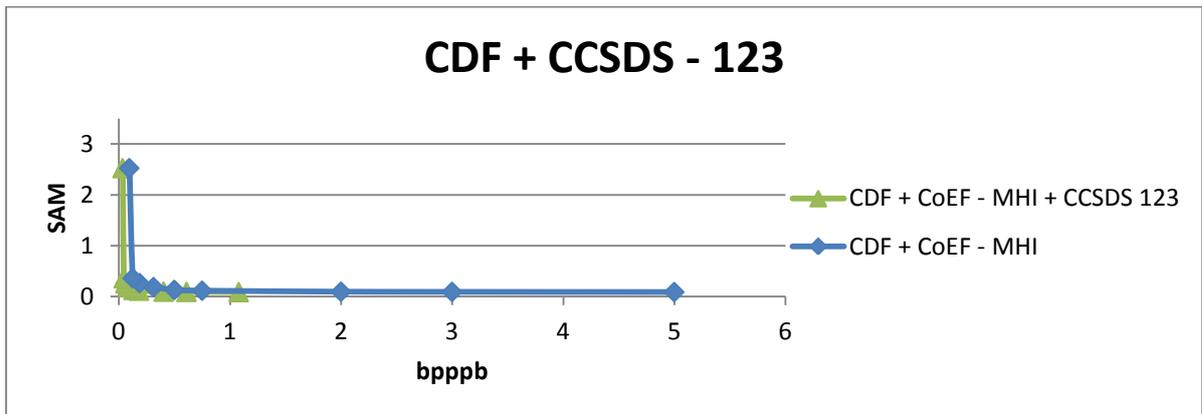
A.6.1.- Gráficas de resultados de las simulaciones para la imagen del sensor AIRS CDF+CCSDS-123

A.6.1.1.- Métrica SNR



Gráfica A.6.1.1.1: Comparativa entre los resultados del compresor CDF y el del mismo compresor tras aplicar a la imagen el CCSDS-123

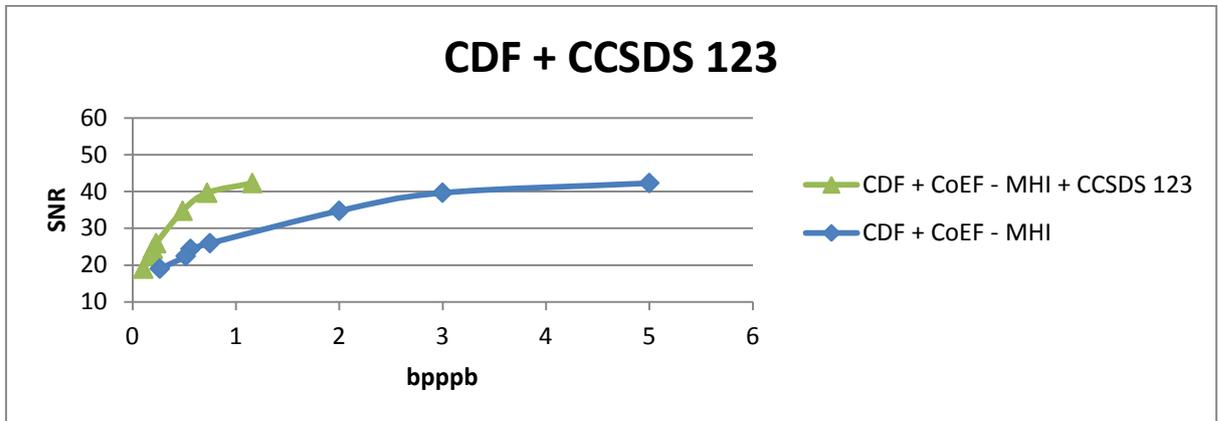
A.6.1.2.- Métrica SAM



Gráfica A.6.1.2.1: Comparativa entre los resultados del compresor CDF y el del mismo compresor tras aplicar a la imagen el CCSDS-123

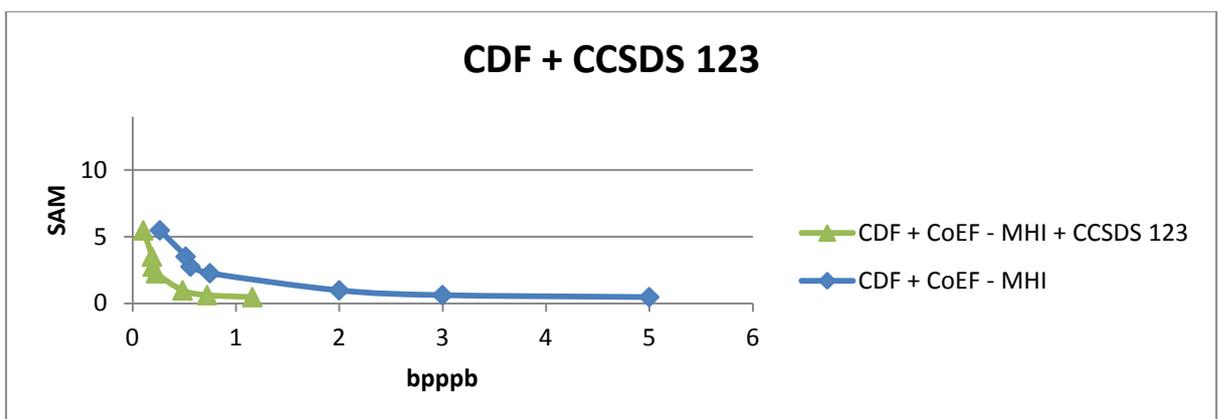
**A.6.2.- Gráficas de resultados de las simulaciones para la imagen del sensor AVIRIS.
CDF+CCSDS-123**

A.6.2.1.- Métrica SNR



Gráfica A.6.2.1.1: Comparativa entre los resultados del compresor CDF y el del mismo compresor tras aplicar a la imagen el CCSDS-123

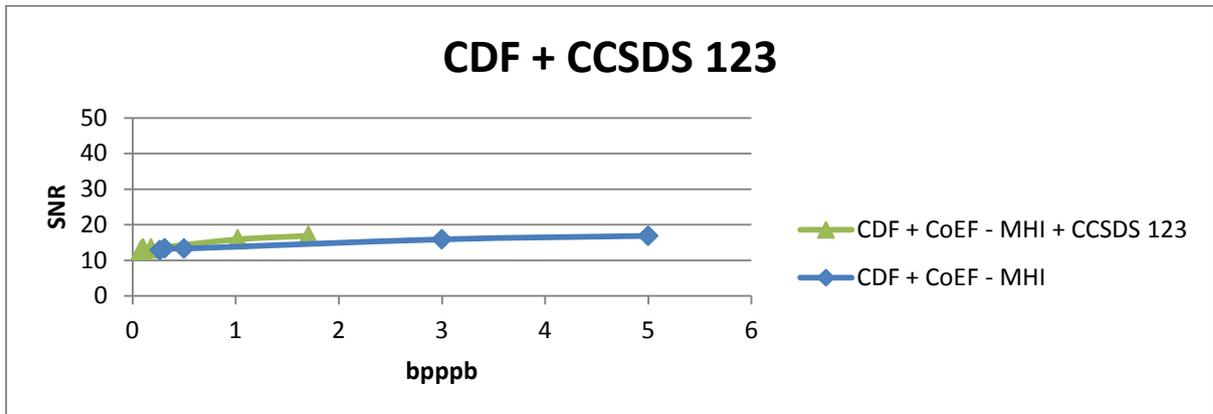
A.6.2.2.- Métrica SAM



Gráfica A.6.2.2.1: Comparativa entre los resultados del compresor CDF y el del mismo compresor tras aplicar a la imagen el CCSDS-123

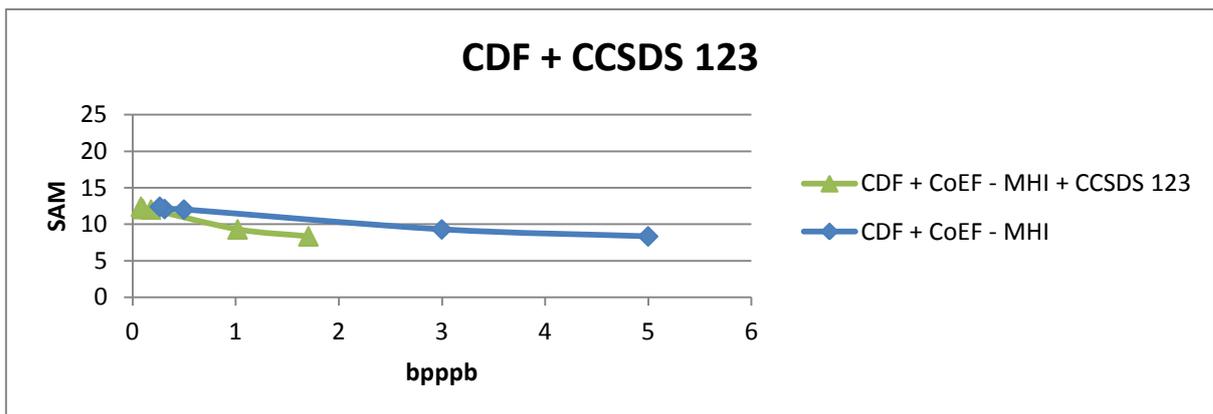
**A.6.3.- Gráficas de resultados de las simulaciones para la imagen del sensor CRISM.
CDF+CCSDS-123**

A.6.3.1.- Métrica SNR



Gráfica A.6.3.1.1: Comparativa entre los resultados del compresor CDF y el del mismo compresor tras aplicar a la imagen el CCSDS-123

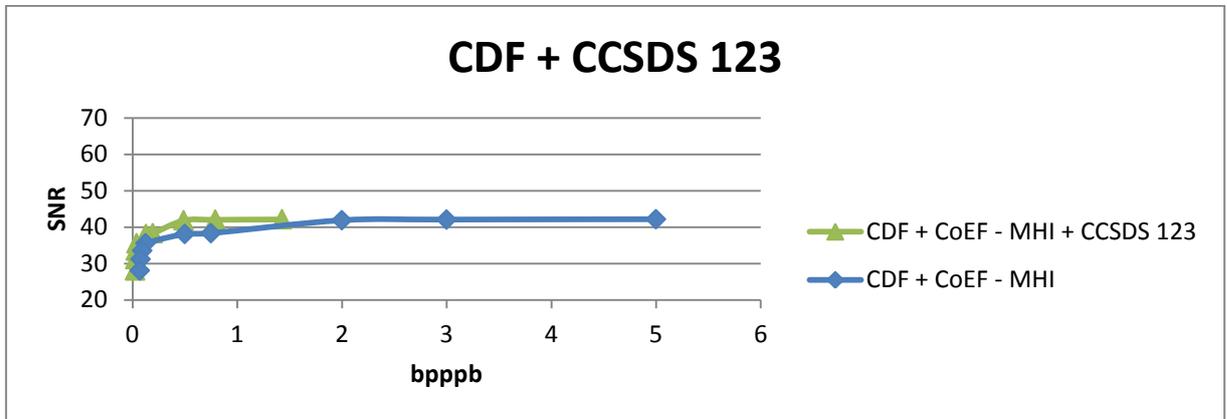
A.6.3.2.- Métrica SAM



Gráfica A.6.3.2.1: Comparativa entre los resultados del compresor CDF y el del mismo compresor tras aplicar a la imagen el CCSDS-123

*A.6.4.- Gráficas de resultados de las simulaciones para la imagen del sensor IASI.
CDF+CCSDS-123*

A.6.4.1.- Métrica SNR



Gráfica A.6.4.1.1: Comparativa entre los resultados del compresor CDF y el del mismo compresor tras aplicar a la imagen el CCSDS-123

