UNIVERSIDAD DE GRANADA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
INFORMÁTICA Y DE TELECOMUNICACIÓN

Aproximación rala de imágenes naturales basada en optimización no convexa y aplicaciones a restauración

Tesis Doctoral

Luis Mancera Pascual

Ingeniero Superior en Informática

2008

Editor: Editorial de la Universidad de Granada
Autor: Luis Mancera Pascual
D.L.: Gr. 2952 - 2008
ISBN: 978-84-338-4717-1

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN E INTELIGENCIA ARTIFICIAL

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA Y DE TELECOMUNICACIÓN

Aproximación rala de imágenes naturales basada en optimización no convexa y aplicaciones a restauración

Tesis Doctoral

Autor: Luis Mancera Pascual
Ingeniero Superior en Informática

Director: Francisco Javier de la Portilla Muelas
Doctor Ingeniero de Telecomunicación

2008

Título:
Aproximación rala de imágenes naturales basada en optimización no convexa y aplicaciones a restauración

Autor:
Luis Mancera Pascual

Tribunal:

Presidente	:	Rafael Molina Soriano
Vocales	:	Nick G. Kingsbury Carlos García Puntonet Jesús Malo López
Secretario	:	Jesús Chamorro Martínez
Suplentes	:	Luis Salgado Álvarez de Sotomayor Gabriel Cristóbal Pérez

Acuerdan otorgar la calificación de:

Granada, 11 de Febrero de 2008

A mi padre

At page 59, vol. I, we find this sentence – "He was advancing by the only road that was ever traveled by the stranger as he approached the Hut; or, he came up the valley." This is merely a vagueness of speech. [···] The whole would be clearer thus – "He was advancing by the valley – the only road traveled by a stranger approaching the Hut." We have here sixteen words, instead of Mr. Cooper's twenty-five.

E.A. Poe – Comments on F. Cooper's "Wyandotte".

... and this round gold is but the image of the rounder globe, which, like a magician's glass, to each and every man in turn but mirrors back his own mysterious self. Great pains, small gains for those who ask the world to solve them; it cannot solve itself.

H. Melville – Moby Dick or the whale, ch. 99, "The Doubloon"

Agradecimientos

En primer lugar, quiero agradecer efusivamente a Javier Portilla su gran compromiso con esta Tesis, y el enorme esfuerzo e interés que ha puesto en convertirme en un científico. Nadie podría haber hecho mejor esta tarea. También quiero expresar mi profundo agradecimiento a Rafael Molina, a quien admiro como profesor, como investigador y como persona, y de quien valoro su cercanía y sus oportunos consejos.

Jose A. Guerrero-Colón ha sido mi compañero de fatigas y apoyo incondicional durante estos años. Sin Jesús Chamorro esta Tesis no existiría. Le debo la pasión por el procesamiento de imágenes y, junto a Javier, la oportunidad de entrar en la investigación. Además, Joaquín Valdivia y Juan Luis Castro siempre han sido un ejemplo para mí, y sus enseñanzas y consejos los guardo como oro en paño. Agradezco al grupo de Procesamiento de la Información Visual por acogerme y a todo el Departamento de Ciencias de la Computación e Inteligencia Artificial por los años imborrables que he pasado en él.

Agradezco también a Nick Kingsbury por recibirme en la Universidad de Cambridge con los brazos abiertos, y al Grupo de Procesamiento de Señal del Departamento de Ingeniería de dicha Universidad, donde he pasado tan buenos ratos. También agradezco a Mario Figueiredo, Nick Kingsbury y Michael Elad sus comentarios (y sus críticas) sobre nuestro trabajo.

Gracias a Ignacio Requena y a Andrés Cano por su ayuda para superar la burocracia que lleva escribir y depositar una Tesis.

Gracias sobre todo a Sonia y a Luis, que dan sentido a todo lo que hago, y a toda mi familia, tanto directa como política.

Este trabajo ha sido financiado por los proyectos TIC-2003-01504 y TEC2006/13845/TCM del Ministerio de Educación y Ciencia.

Resumen

Las necesidades actuales han potenciado el rápido crecimiento de las aplicaciones del procesamiento de imágenes. Al principio, estos problemas se aproximaban de una forma totalmente heurística. Actualmente, cada vez se da más importancia a desarrollar buenos modelos de las imágenes que permitan una aplicación genérica a gran variedad de tareas. Nuestro cerebro puede discriminar la información relevante en una imagen distorsionada, ya que los objetos del mundo que nos rodean tienen una determinada estructura típica que se refleja en las imágenes naturales (aquellas que representan el mundo que nos rodea), mientras que las imágenes aleatorias no tienen, en general, ninguna estructura. Para realizar de manera automática este tipo de procesamiento, es de vital importancia contar con un buen conocimiento *a priori* sobre la estructura típica de las imágenes naturales.

Inspirándonos en lo que sabemos del procesamiento de los estímulos visuales en el cerebro, queremos representar las imágenes con el menor número posible de muestras, para facilitar su descripción estadística, captación, procesamiento y/o almacenamiento. La capacidad de expresar información con pocos elementos se puede ver considerablemente aumentada si transformamos los píxeles a un nuevo dominio redundante (que tiene más coeficientes que píxeles hay en la imagen). Dado un vector y un conjunto de vectores que definen un dominio redundante, el problema de aproximación rala consiste en minimizar una cierta medida del error cometido al expresar el vector dado combinando linealmente un número dado de vectores (desconocidos) del conjunto. Debido a la complejidad de este problema, las aproximaciones tradicionales han consistido en heurísticos voraces o en relajar la función de coste asociada para que sea convexa. Además, se ha puesto mucho esfuerzo en encontrar las condiciones teóricas bajo las cuales estas técnicas resuelven óptimamente el problema. Existe una tercera clase de métodos basados en umbralización iterativa. Han demostrado ser más eficientes y dar mejores resultados, tanto en su capacidad de compactación de la energía como en su aplicación a problemas de restauración, que los heurísticos voraces o las estrategias clásicas de optimización. Algunas de sus variantes más exitosas todavía no están bien fundamentadas en la teoría, ni

su rendimiento bien estudiado en cuanto a la aplicación a restauración de imágenes.

En esta Tesis se estudian dos métodos de este último tipo, que nunca han sido derivados como solución a un problema de optimización. La derivación que proponemos demuestra que es relativamente sencillo aplicar técnicas conocidas de optimización para encontrar mínimos locales al problema de aproximación rala directamente. El primero de ellos lleva a cabo una minimización del error de aproximación, dado un valor de una determinada norma ℓ_p del vector que pondera la importancia de cada vector del conjunto escogido para formar la aproximación. Su convergencia se demuestra describiéndolo como un método basado en proyecciones ortogonales alternas entre dos conjuntos. Llamamos a este método ℓ_p-AP. Estudiamos los casos $p = 0$, para el que el método es sub-óptimo, y $p = 1$, para el que obtenemos el mínimo global. El énfasis de nuestros experimentos está en realizar pruebas exhaustivas para analizar su comportamiento en condiciones prácticas de procesamiento de imágenes. Veremos que el método ℓ_0-AP es claramente superior a ℓ_1-AP y a los métodos voraces en términos de compactación de energía. El segundo método se deriva en base al descenso en la dirección opuesta al gradiente en versiones cada vez menos suavizadas de una función continua y restringida equivalente a la función de coste (discontinua y sin restringir) del problema de aproximación rala. Llamamos ℓ_0-GM a este método. Veremos que los resultados de compactación de ℓ_0-GM mejoran a los de ℓ_0-AP, siendo comparable con el estado de la técnica en aproximación rala. También presentamos una versión convexa de este método, a la que llamamos ℓ_1-GM. Por último, hemos adaptado los métodos propuestos para resolver problemas de restauración. En concreto, proponemos el uso de ℓ_0-AP para eliminar artefactos de cuantificación espacial, y de ℓ_0-GM para la interpolación de píxeles o zonas perdidas de la imagen, para la interpolación de imágenes de color a partir de mosaicos y para el incremento de detalle o super-resolución. Veremos que obtenemos buenos resultados para estos problemas, siendo competitivos o superiores a otros métodos elegidos como buenos representantes del estado de la técnica actual.

Summary

In the modern world, current needs have promoted a fast increase in the interest in image processing applications. In the beginning, the applications were approximated as heuristics, using techniques particularly adapted to each case. Nowadays, it is increasingly important to develop good image models allowing a generic application to a wide variety of tasks. Our brain is able to discriminate the relevant information in a distorted image, because the objects of the world have a typical structure. This structure is reflected in natural images (those representing the real world we are living in), whilst random images do not have, in general, any structure at all. If we want to replicate this behaviour in a machine, it is very important to have a good *a priori* knowledge about the typical structure of natural images.

Inspired on what we know about the processing of visual stimuli in our brain, we want to represent images with as few samples as possible, so making easier not only to describe them statistically, but also its capture, processing and storage. The ability to express images with few elements can be considerably increased if we transform the pixels from the image to a new redundant domain, where there are more coefficients than pixels in the image. Given a vector and a set of other vectors defining a redundant domain, the sparse approximation problem is formulated as minimizing a certain measurement of the error when expressing the vector as a linear combination of a given number of (unknown) vectors from the given set. Because of the inherent complexity of this problem, most approximations have been traditionally based on greedy heuristics or convex relaxation of the cost function. In addition, much effort has been made to find the theoretical conditions under which these two types of approximation find the global optimum for the sparse approximation problem. A third class of methods exists based on iterative thresholding. They have been shown to be more efficient and provide better results than greedy heuristics or classic optimization methods. However, some of their more successful variants are still not well grounded in theory, and their performance when applied to restoration has not been yet extensively studied. In this Thesis, two methods of this kind are examined. They had been already proposed as heuristics, but

never derived as solutions to an optimization problem. This Thesis shows that it is possible to apply classical optimization tools to obtain useful (though suboptimal) solutions to the sparse approximation problem.

The first proposed method minimizes the approximation error given a value of a determined ℓ_p-norm of the representation. Its convergence is proven by describing it as based on alternated orthogonal projections between two sets. We call this method ℓ_p-AP. We focus on the cases $p = 0$, where sub-optimal solutions are found, and $p = 1$, where the global optimum is achieved. The emphasis of our experiments is on analysing the behaviour of the methods in practical image processing conditions, by means of intensive experiments. We show that ℓ_0-AP is neatly superior to ℓ_1-AP and greedy methods. The second method is derived as a gradient descent in successively decreasingly smoothed versions of a continuous, constrained function which is equivalent to the (discontinuous, unconstrained) cost function of the sparse approximation problem. We call this method ℓ_0-GM. We show that ℓ_0-GM outperforms ℓ_0-AP, being its performance close to the state-of-the-art in sparse approximation. We also show a convex version of this method, which we call ℓ_1-GM.

Last but not least, we have adapted the proposed methods to be applied to several restoration problems. We propose to use ℓ_0-AP for removing spatial quantization artifacts and ℓ_0-GM for interpolation of lost pixels or missing regions of the image, for the interpolation of color images from mosaics and finally for increasing the detail level or super-resolve images. We show that the proposed methods provide good results to these problems, being superior or similar to other methods chosen as representatives from current state-of-the-art.

Índice general

. Índice de figuras **xxiii**

. Índice de tablas **xxxvii**

1. Introducción **1**
1.1. Introducción y objetivos 1
1.2. Contribución de esta Tesis 8

2. El problema de aproximación rala **11**
2.1. Raleza de análisis y raleza de síntesis 11
2.2. Formulación del problema de aproximación rala 15
2.3. El problema de aproximación rala en la literatura 16
2.3.1. Heurísticos voraces 17
2.3.2. El problema de relajación convexa y *Basis Pursuit* . . 19
2.3.3. Umbralización iterativa 21
2.4. Condiciones de equivalencia al minimizar las normas ℓ_1 y ℓ_0 25

3. Aproximación rala usando proyecciones alternas **29**
3.1. El método ℓ_p-AP 30
3.1.1. ℓ_0-AP 32
3.1.2. ℓ_1-AP 35
3.2. Minimización del error cuadrático para una selección de coeficientes dada 39
3.3. Implementación 41
3.3.1. Representaciones 41
3.3.2. Convergencia y criterio de parada 41
3.4. Resultados y discusión 42
3.4.1. Comparación de algunos métodos previos 43
3.4.2. Comparación de ℓ_0-AP con ℓ_1-AP y métodos previos 44
3.4.3. Tiempo de computación 49
3.5. Conclusiones 50

4. Aproximación rala aplicando descenso de gradiente 53
4.1. Formulación continua de la función de coste 54
4.2. Minimización local con norma ℓ_0: IHT 56
4.3. Minimización global con norma ℓ_0: ℓ_0-GM 58
4.3.1. Usando una sóla solución para cualquier nivel de raleza 62
4.4. Implementación 64
4.5. Resultados y discusión del método ℓ_0-GM 64
4.6. Descenso de gradiente para minimizar la norma ℓ_1: IST & ℓ_1-GM 65
4.6.1. Formulación alternativa de la función de coste de la relajación convexa 65
4.6.2. Minimización de la función de coste con umbral fijo: IST 69
4.6.3. Una minimización convexa más eficiente: ℓ_1-GM . . . 69
4.6.4. Ventajas prácticas de ℓ_1-GM 70
4.7. Conclusiones 71

5. Aplicación a restauración de imágenes 81
5.1. Consistencia con la observación 83
5.2. Formulación usando raleza en sentido de síntesis 84
5.3. Estimación usando ℓ_p-AP y raleza en sentido de síntesis . . . 85
5.4. Formulación usando raleza en sentido de análisis 87
5.5. Estimación usando ℓ_p-GM y raleza en sentido de análisis . . 88

6. Varias aplicaciones 91
6.1. Eliminación de artefactos de cuantificación 91
6.1.1. Introducción 91
6.1.2. Conjunto de consistencia 92
6.1.3. Implementación 93
6.1.4. Resultados y discusión 95
6.1.5. Conclusiones 101
6.2. Interpolación de regiones perdidas 101
6.2.1. Introducción 101
6.2.2. Conjunto de consistencia 102
6.2.3. ℓ_0-AP: nueva estrategia de búsqueda del radio 103
6.2.4. Implementación 104
6.2.5. Resultados y discusión 105
6.2.6. Conclusiones 109
6.3. Interpolación espacial-cromática para mosaicos en cámaras digitales 111
6.3.1. Introducción 111
6.3.2. Conjunto de consistencia 112

6.3.3. Restricción añadida para favorecer la correlación cromática espacial . . . 113
6.3.4. Implementación . . . 114
6.3.5. Resultados y discusión . . . 114
6.3.6. Conclusiones . . . 116
6.4. Incremento de detalle . . . 117
6.4.1. Introducción . . . 117
6.4.2. Conjunto de consistencia . . . 121
6.4.3. Implementación . . . 122
6.4.4. Resultados y discusión . . . 122
6.4.5. Conclusiones . . . 123

7. Conclusiones y trabajo futuro 125
7.1. Conclusiones . . . 125
7.2. Trabajo futuro . . . 128

. English translation 129

8. Introduction 131
8.1. Introduction and objectives . . . 131
8.2. Contribution of this Thesis . . . 137

9. The Sparse Approximation Problem 141
9.1. Analysis-based sparseness vs. Synthesis-based sparseness . . . 141
9.2. Formulation of the sparse approximation problem . . . 144
9.3. The sparse approximation problem in the literature . . . 146
9.3.1. Greedy heuristics . . . 147
9.3.2. Convex relaxation problem and *Basis Pursuit* . . . 148
9.3.3. Iterative shrinkage . . . 150
9.4. Equivalence conditions when minimising ℓ_1 and ℓ_0-norms . . . 153

10. Sparse approximation using alternating projections 157
10.1. ℓ_p-AP method . . . 158
10.1.1. ℓ_0-AP . . . 159
10.1.2. ℓ_1-AP . . . 162
10.2. Mean square error minimisation for a given selection of coefficients . . . 167
10.3. Implementation . . . 168
10.3.1. Representations . . . 168
10.3.2. Convergence and stopping criterion . . . 169
10.4. Results and discussion . . . 170
10.4.1. Some previous methods . . . 170

10.4.2. Comparison of ℓ_0-AP, ℓ_1-AP and previous methods . 171
10.4.3. Computational load 176
10.5. Conclusions 177

11. Sparse approximation using gradient descent 181
11.1. An alternative formulation with a continuous cost function . 182
11.2. Local minimisation with ℓ_0-norm: IHT 184
11.3. Global minimisation with ℓ_0-norm: ℓ_0-GM 185
11.3.1. Using a single solution for all the sparseness levels . . 189
11.4. Implementation 191
11.5. Results and discussion for ℓ_0-GM 191
11.6. Gradient descent for minimisation of ℓ_1-norm: IST & ℓ_1-GM 192
11.6.1. Alternative formulation of the convex cost function . 192
11.6.2. Cost function minimisation with a fixed threshold: IST 196
11.6.3. A more efficient convex minimisation: ℓ_1-GM 196
11.6.4. Practical advantages of ℓ_1-GM 197
11.7. Conclusions 198

12. Application to image restoration 207
12.1. Consistency with an observation 209
12.2. Formulation using synthesis-sense sparseness 210
12.3. Estimation using ℓ_p-AP and synthesis-sense sparseness . . . 211
12.4. Formulation using analysis-sense sparseness 213
12.5. Estimation using ℓ_p-GM and analysis-sense sparseness 213

13. Some applications 215
13.1. Removing quantisation artifacts 215
13.1.1. Introduction 215
13.1.2. Consistency set 216
13.1.3. Implementation 217
13.1.4. Results and discussion 218
13.1.5. Conclusions 224
13.2. Interpolation of missing pixels 225
13.2.1. Introduction 225
13.2.2. Consistency set 225
13.2.3. ℓ_0-AP: new strategy for searching the radius 226
13.2.4. Implementation 227
13.2.5. Results and discussion 228
13.2.6. Conclusions 232
13.3. Spatial-chromatic interpolation in digital camera mosaics . . 234
13.3.1. Introduction 234
13.3.2. Consistency set 234

13.3.3. Additional constraint increasing the spatial-chromatic correlation . . . 235
13.3.4. Implementation . . . 236
13.3.5. Results and discussion . . . 236
13.3.6. Conclusions . . . 238
13.4. Detail increase . . . 241
13.4.1. Introduction . . . 241
13.4.2. Consistency set . . . 242
13.4.3. Implementation . . . 243
13.4.4. Results and discussion . . . 243
13.4.5. Conclusions . . . 244

14. Conclusions and future work **247**
14.1. Conclusions . . . 247
14.2. Future work . . . 249

. Apéndices / Appendices **251**

A. Conjunto de imágenes de prueba **253**

B. Test images set **255**

C. Minimización del error cuadrático medio de la reconstrucción dado un subconjunto de coeficientes activos. **257**
C.1. Primer caso: Subconjunto incompleto . . . 257
C.2. Segundo caso: Subconjunto completo . . . 259

D. Minimisation of the quadratic error of the reconstruction with a given support. **261**
D.1. First case: incomplete subset . . . 261
D.2. Second case: complete subset . . . 262

E. Fusión de dos marcos de Parseval en uno sólo **265**

F. A Parseval frame formed concatenating two Parseval frames **267**

G. Listado de publicaciones **269**

H. Publications **271**

. Bibliografía **273**

Índice de figuras

1.1. **Arriba - izquierda**, imagen estándar *Einstein*. **Arriba - derecha**, imagen aleatoria (ruido blanco gaussiano). **Abajo**, suma de las imágenes anteriores. 3
1.2. **Izquierda**, imagen obtenida usando el 10 % de los píxeles de mayor desviación respecto a la media en la imagen *Einstein*. **Derecha**, imagen construida utilizando el 10 % de los coeficientes de mayor amplitud en la representación de Fourier de la misma imagen. 4
1.3. **Izquierda**, sub-banda de la representación de la imagen *Peppers* bajo un banco de filtros de tipo ondícula (DT-CWT). Los píxeles oscuros representan coeficientes de alta amplitud y los claros aquellos con baja amplitud. **Derecha**, la misma sub-banda de una aproximación rala a la imagen. 5

2.1. Comparación de resultados de aproximación rala obtenidos usando los coeficientes de mayor amplitud de las respuestas lineales de diferentes representaciones y promediados en nuestro conjunto de prueba. El rendimiento se mide en función de la PSNR obtenida usando un cierto número de coeficientes, que aparece normalizado por el número total de píxeles en la imagen. 13

2.2. **Arriba-izquierda**, sub-banda de la escala de más alta frecuencia de la representación de análisis de *Peppers* usando DT-CWT con 8 escalas. Los puntos claros y oscuros corresponden respectivamente con amplitudes pequeñas y grandes de los coeficientes. El tamaño de la sub-banda ha sido duplicado en ambas direcciones mediante replicado de filas y columnas para coincidir con el tamaño de la imagen y luego se ha recortado a 64×64 para conseguir mejor visibilidad. **Arriba-centro**, la misma sub-banda, pero esta vez obtenida no linealmente con el método ℓ_0-AP propuesto en esta Tesis (ver Capítulo 3). **Arriba-derecha**, resultado de aplicar un umbral a las amplitudes del resultado del panel central (preservando 7 veces menos coeficientes que píxeles tiene la imagen). **Abajo-izquierda**, imagen original, que es reconstruida perfectamente por los coeficientes que corresponden con los paneles izquierdo y central de la fila superior. **Abajo-derecha**, aproximación obtenida con los coeficientes ralos del panel superior-derecho (35,67 dB). . . . 14
2.3. Forma de la p-ésima potencia de la norma ℓ_p en una dimensión para varios valores de p. 17

3.1. **Arriba**, explicación gráfica del método ℓ_0-AP. **Abajo**, lo mismo para ℓ_1-AP. Sólo se muestra una cara de la bola ℓ_1 para aumentar la claridad. 33
3.2. Gráfica en ejes logarítmicos de la calidad en la aproximación (PSNR, en dB) en función del número de iteraciones para ℓ_0-AP usando tres imágenes y dos niveles de raleza. La representación usada es DT-CWT. El número al final de las curvas es la PSNR en la convergencia. Los números que acompañan al punto de tangencia (indicado por las curvas punteadas) son la PSNR y el número de iteraciones al cumplir el criterio de parada. 34
3.3. Curvas de convergencia en escala semi-logarítmica para el método ℓ_1-AP usando tres imágenes y dos niveles de raleza. Los detalles son similares a los de la Figura 3.2. También está indicada la norma ℓ_0, normalizada por N, de la solución en la convergencia. 40
3.4. Resultados de compactación promediados en el conjunto de prueba para StOMP, DT+OP, IHT e IST. **Arriba**, usando DT-CWT con 8 escalas. **Abajo**, usando Curvelets con 6 escalas. 44

3.5. Resultados de compactación, promediados en el conjunto de prueba, de los métodos ℓ_0-AP, ℓ_1-AP, ℓ_1-AP+OP, IHT y DT+OP. **Arriba**, usando DT-CWT con 8 escalas. **Abajo**, usando Curvelets con 6 escalas. 46
3.6. Comparación visual de los métodos usando $0{,}0765 \cdot N$ coeficientes Curvelets en la imagen *Einstein*, donde N es el número de píxeles de la imagen. Los resultados están recortados a tamaño 128×128, comenzando en el píxel $(71, 41)$, para aumentar la visibilidad. **Columna izquierda**, desde arriba hacia abajo: imagen original, resultado de ℓ_1-AP (30,85 dB), y ℓ_1-AP+OP (33,52 dB). **Columna derecha**, desde arriba hacia abajo: resultado de StOMP (28,66 dB), IHT (29,10 dB) y ℓ_0-AP (32,98 dB). 52

4.1. La línea gruesa muestra la función mínimo entre $y(x) = 1$ (línea intermitente) e $y(x) = x^2$ (línea punteada). 56
4.2. **Arriba**, curvas de convergencia de IHT con un umbral bajo ($\theta = 5$) y tres diferentes valores de α. Hemos usado la imagen *House* y DT-CWT con 8 escalas. **Abajo**, lo mismo para un umbral más alto ($\theta = 60$). 59
4.3. Parábola invertida en el intervalo $[-1, 1]$, centrada en 0 y con máximo 1. Fuera de ese intervalo es constante a 0. 60
4.4. Función de una dimensión con múltiples mínimos, progresivamente suavizada hasta tener sólo uno. La línea negra continua indica el camino que une los mínimos globales a través de la escala. Hemos usado, como filtro de suavizado, una versión normalizada (en área) de $h(x)$ (Ver Figura 4.3). 61
4.5. Resultados de fidelidad-raleza usando ℓ_0-GM con $\beta = 0{,}9$ (círculos, $1{,}5 \cdot 10^2$ iteraciones) y $\beta = 0{,}99$ (línea continua, $1{,}5 \cdot 10^3$ iteraciones), comparado con IHT, usando varios umbrales fijos (líneas intermitentes, 10^5 iteraciones cada una). Se usa la imagen *House* y DT-CWT con 8 escalas. . . 63
4.6. **Arriba**, resultados de fidelidad de la aproximación rala promediados en el conjunto de prueba usando ℓ_0-GM con $\alpha = 1{,}85(\theta^2/2)$, tres diferentes valores de β y usando DT-CWT con 8 escalas. **Abajo**, calidad de la reconstrucción desde los coeficientes más altos en amplitud del vector resultante de ejecutar ℓ_0-GM para un valor muy alto de λ (muy baja raleza), y para los mismos valores de β. Las curvas punteadas corresponden con las del panel de arriba. El eje de ordenadas ha sido re-escalado para mejorar la visibilidad. 73

4.7. Resultados de aproximación rala de nuestro método de optimización no convexa (ℓ_0-GM) promediados para las imágenes del conjunto de prueba, y comparados con otros métodos vistos previamente (StOMP, IHT, ℓ_0-AP y ℓ_1-AP+OP). 74
4.8. Recorte a tamaño 64×64 de la reconstrucción de la imagen *Einstein* usando $0{,}04 \cdot N$ (2605) coeficientes activos, utilizando DT-CWT con 8 escalas, para varios métodos. **Arriba - izquierda**, resultado de StOMP, implementado como se vio en la Sección 3.4.1 (28,98 dB). **Arriba - derecha**, IHT (31,20 dB). **Centro - izquierda**, ℓ_1-AP (29,70 dB). **Centro - derecha**, ℓ_0-AP (31,97 dB). **Abajo - izquierda**, ℓ_1-AP+OP (32,38 dB). **Abajo - derecha**, ℓ_0-GM (33,28 dB). 75
4.9. **Arriba**, curvas de convergencia de IST con un umbral bajo ($\theta = 5$) y tres diferentes valores de α. Hemos usado la imagen *House* y DT-CWT con 8 escalas. **Abajo**, lo mismo para un umbral más alto ($\theta = 60$). 76
4.10. Resultados de fidelidad-raleza usando ℓ_1-GM con $\beta = 0{,}9$ (círculos, $1{,}5 \cdot 10^2$ iteraciones) y $\beta = 0{,}99$ (línea continua, $1{,}5 \cdot 10^3$ iteraciones), comparado con IHT, usando varios umbrales fijos (líneas intermitentes, 10^3 iteraciones cada una). Se usa la imagen *House* y DT-CWT con 8 escalas. . . 77
4.11. Resultados de aproximación rala promediados en el conjunto de prueba usando ℓ_1-GM con $\alpha = 1{,}85\theta$, diferentes valores de β y usando DT-CWT con 8 escalas. También se muestran el resultado correspondiente a ℓ_1-AP. 78
4.12. Iteraciones llevadas a cabo por los métodos ℓ_1-AP y ℓ_0-GM ($\alpha = 1{,}85\theta$, $\beta = 0{,}99$) para alcanzar un resultado cercano al óptimo para distintos niveles de raleza. Usamos la imagen *Barbara* y DT-CWT con 8 escalas. 79

5.1. **Arriba a la izquierda**, recorte de *Peppers* empezando en la fila 111, columna 91. **Abajo a la izquierda**, mismo recorte de la sub-banda de alta frecuencia del análisis lineal de *Peppers* con DT-CWT 8 escalas, correspondiente a -45^{o} de orientación. Previamente se ha duplicado el tamaño de esta sub-banda mediante replicado de píxeles para coincidir con el tamaño de la imagen. **Arriba a la derecha**, imagen degradada poniendo a cero, aleatoriamente, el 40 % de los píxeles. **Abajo a la derecha**, sub-banda correspondiente a la imagen degradada. 82

6.1. Ejemplo de aplicación de ℓ_1-AP y ℓ_0-AP a la eliminación de artefactos de cuantificación espacial. **Arriba - izquierda**, imagen *Einstein* original, recortada a 128 × 128 píxeles. **Arriba - derecha**, cuantificación con 3 bits observada (PSNR: 27,98 dB). **Centro - izquierda**, resultado de ℓ_1-AP usando DT-CWT con 8 escalas (30,17 dB). **Centro - derecha**, resultado de ℓ_1-AP usando Curvelets 6 escalas (30,61 dB). **Abajo - izquierda**, resultado de ℓ_0-AP usando DT-CWT 8 escalas (31,21 dB). **Abajo - derecha**, resultado de ℓ_0-AP usando Curvelets 6 escalas (31,38 dB). 96

6.2. Ejemplo de aplicación de ℓ_1-AP y ℓ_0-AP a la eliminación de artefactos de cuantificación espacial. **Arriba - izquierda**, imagen *Peppers* original, recortada a 128 × 128 píxeles. **Arriba - derecha**, cuantificación con 3 bits observada (PSNR: 28,81 dB). **Centro - izquierda**, resultado de ℓ_1-AP usando DT-CWT con 8 escalas (29,08 dB). **Centro - derecha**, resultado de ℓ_1-AP usando Curvelets 6 escalas (29,50 dB). **Abajo - izquierda**, resultado de ℓ_0-AP usando DT-CWT 8 escalas (31,06 dB). **Abajo - derecha**, resultado de ℓ_0-AP usando Curvelets 6 escalas (30,85 dB). 97

6.3. **Arriba - izquierda**, *Einstein* cuantificada con 3 bits y recortada a 128 × 128 píxeles empezando en (71, 41), (27,98 dB). **Arriba - derecha**, resultado de RRIR (30,39 dB). **Centro - izquierda**, resultado de CD (30,44 dB). **Centro - derecha**, resultado de DT+OP usando DT-CWT 8 escalas (30,72 dB). **Abajo - izquierda**, resultado de ℓ_0-AP usando DT-CWT con 8 escalas (31,21 dB). **Abajo - derecha**, resultado de ℓ_0-AP utilizando conjuntamente DT-CWT con 8 escalas y Curvelets 6 escalas, con igual factor de escala, $\sqrt{\frac{1}{2}}$, para ambos (31,93 dB). 99

6.4. **Arriba - izquierda**, *Peppers* cuantificada con 3 bits y recortada a 128 × 128 píxeles empezando en (71, 41), (28,81 dB). **Arriba - derecha**, resultado de RRIR (29,65 dB). **Centro - izquierda**, resultado de CD (29,85 dB). **Centro - derecha**, resultado de DT+OP usando DT-CWT 8 escalas (30,38 dB). **Abajo - izquierda**, resultado de ℓ_0-AP usando DT-CWT con 8 escalas (31,07 dB). **Abajo - derecha**, resultado de ℓ_0-AP utilizando conjuntamente DT-CWT con 8 escalas y Curvelets 6 escalas, con igual factor de escala, $\sqrt{\frac{1}{2}}$, para ambos (31,46 dB). 100

6.5. **Izquierda**, detalle del cielo de una imagen fotográfica de 8 bits con contraste amplificado aproximadamente 40 veces. **Derecha**, mismo detalle tras procesar con ℓ_0-AP. 101
6.6. **Línea continua - eje izquierdo**, error cuadrático medio normalizado de los píxeles estimados con respecto a valores normalizados al valor mínimo de esta curva en ordenadas, y al valor para el que ocurre este mínimo, en abscisas. **Línea intermitente - eje derecho**, valor cuadrático medio normalizado de los píxeles estimados. **Línea punteada**, desviación típica para cada valor del eje de abscisas. **Línea mixta segmentos y puntos**, valor cuadrático medio de los píxeles originales en las posiciones perdidas. Todas las curvas están promediados en nuestro conjunto de prueba, usando una máscara que elimina alrededor del 40 % de los píxeles. . 104
6.7. Ejemplo visual de interpolación de píxeles perdidos aleatoriamente. **Arriba - izquierda**, imagen *Barbara* de nuestro conjunto de prueba recortada a 128×128. **Arriba - derecha**, pérdida aleatoria de $\approx$ 80 % de los píxeles rellenados con la media de la imagen (PSNR: 14,75 dB). **Centro - izquierda**, interpolación conseguida con ℓ_1-GM (23,26 dB). **Centro - derecha**, resultado de *Fast-inpainting* (24,84 dB). **Abajo - izquierda**, resultado de ℓ_0-GM usando Curvelets 6 escalas (25,19 dB) **Abajo - derecha**, interpolación conseguida con ℓ_0-GM combinando Curvelets 6 escalas con DCT local con tamaño de bloque 32×32 y factores de escala iguales $\sqrt{0{,}5}$ (25,65 dB). 107
6.8. **Arriba - izquierda**, imagen *Barbara* observada con los píxeles perdidos rellenados con la media de la imagen (PSNR: 24,19 dB). **Arriba - derecha**, resultado de *Fast-inpainting* (32,71 dB). **Abajo - izquierda**, resultado de *EM-inpainting* usando Curvelets con 6 escalas y DCT local con tamaño de bloque 32×32 (34,14 dB) y ambos factores de escala con valor $\sqrt{0{,}5}$. **Abajo - derecha**, nuestro resultado con ℓ_0-GM usando Curvelets 6 escalas (34,92 dB). 109
6.9. **Izquierda**, detalle del resultado de *EM-inpainting* mostrado en la Figura 6.8 (34,38 dB). **Derecha**, lo mismo para el método ℓ_0-GM (35,13 dB). 110

6.10. **Arriba - izquierda**, fotografía real dañada. **Arriba - derecha**, resultado de *Fast-inpainting*. **Abajo - izquierda**, resultado de *EM-inpainting* usando Curvelets y DCT local. **Abajo - derecha**, nuestro resultado con ℓ_0-GM usando Curvelets 6 escalas y DCT local con tamaño de bloque 32×32 y ambos factores de escala a $\sqrt{0{,}5}$. 110
6.11. Comparación visual entre los métodos ℓ_1-GM y ℓ_0-GM aplicados a desmosaicado. **Arriba-Izquierda**, Detalle de la imagen 15 de la base de datos de *Eastman Kodak*, recortada a tamaño 64×64. **Arriba-Derecha**, resultado de desmosaicado con patrón de Bayer 'GB' del método ℓ_1-GM usando DT-CWT. La PSNR para los canales R, G y B respectivamente es 37,24 dB, 39,87 dB y 37,17 dB. **Abajo-Izquierda**, resultado de ℓ_0-GM usando Curvelets (39,29, 42,27 y 38,20 dB). **Abajo-Derecha**, resultado de ℓ_0-GM usando DT-CWT (39,59, 41,99 y 39,07 dB). 116
6.12. Comparación visual entre los métodos de desmosaicado. **Arriba - izquierda**, resultado del método de [149] sobre imagen 15 de la base de datos de *Eastman Kodak* (*Lighthouse*). Se ha recortado la imagen a tamaño 64×64 para mejorar la visibilidad de los artefactos. La PSNR de las respectivas bandas R,G,B es 39,81, 41,57 y 39,09 dB. **Arriba - derecha**, resultado del método de [148] (39,89, 43,87 y 39,26 dB). **Abajo - izquierda**, resultado del método de [150] (40,39, 44,14 y 39,73 dB). **Abajo - derecha**, nuestro resultado con ℓ_0-GM (39,59, 41,99, 39,07 dB). 120
6.13. **Arriba - izquierda**, resultado de aplicar interpolación por vecino más cercano (replicado) al promediado de bloques 2×2 y submuestreo de la imagen *House*, para recuperar la resolución original (PSNR: 30,52 dB). Hemos recortado una región de tamaño 128×128 para mejorar la visibilidad. **Arriba - derecha**, resultado de la interpolación bilineal (29,65 dB). **Abajo - izquierda**, resultado de ℓ_1-GM (33,50 dB). **Abajo - derecha**, resultado de ℓ_0-GM (33,53 dB). . . 124

8.1. **Top left**, *Einstein* standard image. **Top right**, random image (white Gaussian noise). **Bottom**, sum of the two images above. 133

8.2. **Left**, image obtained using the 10 % of pixels with largest deviation with respect to the global mean of the *Einstein* image. **Right**, image built using the 10 % of the coefficients with largest amplitude in the Fourier representation of the same image. 134
8.3. **Left**, sub-band of the representation of a natural image under a wavelet-like (DT-CWT) filter bank. Dark pixels represent high amplitude coefficients and light ones those with low amplitude. **Right**, same sub-band of a sparse approximation to the natural image. 135

9.1. Sparse approximation results comparison for several representations. Data have been obtained using the largest in amplitude coefficients of the linear responses and then averaging in our test set. Performance is measured in terms of PSNR given a number of coefficients. This number is normalised by the total number of pixels in the image. . . . 143
9.2. **Top-left**, highest-frequency sub-band of analysis vector of *Peppers* image using DT-CWT with 8 scales. Light and dark points correspond, respectively, with low and high amplitudes of the coefficients. The size of the sub-band has been doubled in both directions through replication of rows and columns in order to match the size of the image, and then it has been cropped to 64×64 for visibility. **Top-centre**, same sub-band, but this time non-linearly obtained using the ℓ_0-AP method (see Chapter 10). **Top-right**, result of applying a threshold in amplitude to the result in central panel (preserving 7 times less coefficients than pixels in the image). **Bottom-left**, original image, which is perfectly reconstructed by the representations corresponding to the left and central panels of the top row. **Bottom-right**, approximation obtained with the sparse coefficients corresponding to the top-right panel (35,67 dB). 144
9.3. One-dimensional p-th power of the ℓ_p-norm for different p values. 146

10.1. **Top**, graphical explanation of the ℓ_0-AP method. **Bottom**, same for ℓ_1-AP. Only a face of the ball is shown for clarity. . 161

10.2. Logarithmic plot of the approximation quality (PSNR, in dB) vs. the number of iterations for ℓ_0-AP for three images and two sparseness levels. The representation used here is DT-CWT. The number at the end of the curves is the PSNR at convergence. The numbers accompanying the tangency point (indicated by the intersection with the dotted curves) are the PSNR and the number of iterations obtained when the stopping criterion is reached. 162
10.3. Convergence curves in semi-logarithmic scale for ℓ_1-AP, using three images and two sparseness levels. Details are similar to Figure 10.2. It is also indicated the ℓ_0-norm, normalised by N, of the solution at convergence. 167
10.4. Averaged compaction results (fidelity of the approximation as PSNR, in dB) for our test set using StOMP, DT+OP, IHT and IST. **Top**, using 8-scale DT-CWT. **Bottom**, using 6-scale Curvelets. 172
10.5. Compaction results, averaged in our test set, of methods ℓ_0-AP, ℓ_1-AP, ℓ_1-AP+OP, IHT and DT+OP. **Top**, using 8-scale DT-CWT. **Bottom**, using 6-scale Curvelets. 173
10.6. Visual comparison of the methods using $0{,}0765 \cdot N$ Curvelets coefficients and the *Einstein* image, , where N is the number of pixels in the image. Results are cropped to 128×128, starting at pixel (71, 41), to improved the visibility. **Left column**, from top to bottom: original image and results of ℓ_1-AP (30,85 dB) and ℓ_1-AP+OP (33,52 dB). **Right column**, from top to bottom: results from StOMP (28,66 dB), IHT (29,10 dB) and ℓ_0-AP (32,98 dB). 179

11.1. Bold line shows the minimum between $y(x) = 1$ (dashed) and $y(x) = x^2$ (dotted). 184
11.2. **Top**, IHT convergence curves using a low threshold ($\theta = 5$) and three different α values. We have used *House* image and 8-scale DT-CWT. **Bottom**, same result for a higher threshold ($\theta = 60$). 186
11.3. 1-D smoothing function: an inverted parabola in the interval $[-1, 1]$, centred at 0 and with maximum 1. Outside that interval is 0. 187
11.4. 1-D Function with multiple minima progressively smoothed until obtaining only one. The black continuous line indicates the path joining the global optima through the scale of the smoothing kernel. We have used here as smoothing kernel a normalised (in area) version of $h(x)$ (See Figure 11.3). 189

11.5. Fidelity-sparseness results of ℓ_0-GM, using $\beta = 0{,}9$ (circles, $1{,}5 \cdot 10^2$ iterations) and $\beta = 0{,}99$ (solid, $1{,}5 \cdot 10^3$ iterations), compared to IHT, using several thresholds (dashed, 10^5 iterations). We use *House* image and DT-CWT with 8-scales. 190
11.6. **Top**, sparse approximation fidelity averaged in our test set using ℓ_0-GM with $\alpha = 1{,}85(\theta^2/2)$, three different β values and 8-scale DT-CWT. **Bottom**, quality of the reconstruction from the highest amplitude coefficients of the vector obtained using ℓ_0-GM for a very high λ value (very low sparseness), and the same β values. Dotted curves correspond to that of top panel. The vertical axis has been re-scaled to improve visibility. 199
11.7. ℓ_0-GM sparse approximation results averaged in our test set compared to other methods previously seen (StOMP, IHT, ℓ_0-AP and ℓ_1-AP+OP). 200
11.8. 64×64 crop of the reconstruction of *Einstein* image using $0{,}04 \cdot N$ (2605) active DT-CWT coefficients, for several sparse approximation methods. **Top-left**, result of StOMP, implemented as described in Section 10.4.1 (28,98 dB). **Top-right**, IHT (31,20 dB). **Centre-left**, ℓ_1-AP (29,70 dB). **Centre-right**, ℓ_0-AP (31,97 dB). **Bottom-left**, ℓ_1-AP+OP (32,38 dB). **Bottom-right**, ℓ_0-GM (33,28 dB). 201
11.9. **Top**, convergence curves for IST with a low threshold ($\theta = 5$) and three different α values. We have used *House* image and 8-scale DT-CWT. **Bottom**, same result for a higher threshold ($\theta = 60$). 202
11.10Fidelity-sparseness results of ℓ_0-GM, using $\beta = 0{,}9$ (circles, $1{,}5 \cdot 10^2$ iterations) and $\beta = 0{,}99$ (solid, $1{,}5 \cdot 10^3$ iterations), compared to IHT, using several thresholds (dashed, 10^3 iterations). We use *House* image and DT-CWT with 8-scales. 203
11.11Averaged sparse approximation results in the test set using ℓ_1-GM with $\alpha = 1{,}85\theta$, different β values and using DT-CWT with 8 scales. We also show the result of ℓ_1-AP. 204
11.12Iterations needed to provide nearly optimal result for different sparseness level using ℓ_1-AP and ℓ_1-GM ($\alpha = 1{,}85\theta$, $\beta = 0{,}99$). We use *Barbara* image and 8-scale DT-CWT. . . 205

12.1. **Top-left**, *Peppers* crop, starting at row 111, column 91. **Bottom-left**, same crop of the high frequency sub-band of the linear response to *Peppers* using 8-scales DT-CWT, corresponding to orientation -45^{o}. We have previously doubled the size of this sub-band, through pixel replication, in order to match the image size. **Top-right**, degraded image by setting to zero, randomly, 40 % of the pixels. **Bottom-right**, corresponding sub-band. 208

13.1. Example of application of ℓ_1-AP and ℓ_0-AP to de-quantizing. **Top-left**, original *Einstein* image, cropped to 128×128 pixels. **Top-right**, 3-bits observed quantisation (PSNR: 27,98 dB). **Centre-left**, ℓ_1-AP result using 8-scale DT-CWT (30,17 dB). **Centre-right**, ℓ_1-AP result using 6-scale Curvelets (30,61 dB). **Bottom-left**, ℓ_0-AP result using 8-scale DT-CWT (31,21 dB). **Bottom-right**, ℓ_0-AP result using 6-scale Curvelets (31,38 dB). 220

13.2. Example of application of ℓ_1-AP and ℓ_0-AP to de-quantizing. **Top-left**, original *Peppers* image, cropped to 128×128. **Top-right**, 3-bits observed quantisation (PSNR: 28,81 dB). **Centre-left**, ℓ_1-AP result using 8-scale DT-CWT (29,08 dB). **Centre-right**, ℓ_1-AP result using 6-scale Curvelets (29,50 dB). **Bottom-left**, ℓ_0-AP result using 8-scale DT-CWT (31,06 dB). **Bottom-right**, ℓ_0-AP result using 6-scale Curvelets (30,85 dB). 221

13.3. **Top-left**, *Einstein* quantised with 3 bits and cropped to 128×128, (PSNR: 27,98 dB). **Top-right**, RCIR result (30,39 dB). **Centre-left**, CD result (30,44 dB). **Centre-right**, DT+OP result using DT-CWT with 8 scales (30,72 dB). **Bottom-left**, ℓ_0-AP result using DT-CWT with 8 scales (31,21 dB). **Bottom-right**, ℓ_0-AP result using jointly 8-scale DT-CWT and 6-scale Curvelets, with equal scale factor, $\sqrt{\frac{1}{2}}$ (31,93 dB). 222

13.4. **Top-left**, *Peppers* quantised with 3 bits and cropped to 128×128, (PSNR: 28,81 dB). **Top-right**, RCIR result (29,65 dB). **Centre-left**, CD result (29,85 dB). **Centre-right**, DT+OP result using DT-CWT with 8 scales (30,38 dB). **Bottom-left**, ℓ_0-AP result using 8-scale DT-CWT (31,07 dB). **Bottom-right**, ℓ_0-AP result using jointly 8-scale DT-CWT and 6-scale Curvelets, with equal scale factor, $\sqrt{\frac{1}{2}}$ (31,46 dB). 223
13.5. **Left**, detail of the sky of a photographic 8-bits image with contrast amplified approximately 40 times. **Right**, same detail after processing with ℓ_0-AP. 224
13.6. **Bold line - left axis**, Mean Square Error of the estimated pixels normalised to the minimum value of this curve, in ordinates, and to the value for which this minimum occurs, in abscissas. **Dashed line - right axis**, Normalised Mean Square Value of the estimated pixels. **Dotted line**, typical deviation for each value in the horizontal axis. **Dashed - dotted line**, Mean Square Value of the original pixels in the missing positions. All curves are averaged in our test set, using a random mask where approximately 40 % of the pixels are lost. 227
13.7. Visual interpolation example of randomly missing pixels. **Top-left**, *Barbara* image, cropped to 128×128. **Top-right**, missing ≈ 80 % of the pixels and filling them with the global mean (PSNR: 14,75 dB). **Centre-left**, interpolation made by ℓ_1-GM (23,26 dB). **Centre-right**, result from *Fast-inpainting* (24,84 dB). **Bottom-left**, ℓ_0-GM result using Curvelets with 6 scales (25,19 dB) **Bottom-right**, interpolation made by ℓ_0-GM combining 6-scale Curvelets and LDCT with block size 32×32, and equal scale factors, $\sqrt{0{,}5}$ (25,65 dB). 230
13.8. **Top-left**, *Barbara* image where value of missing pixels is the global mean of the observed ones (PSNR: 24,19 dB). **Top-right**, *Fast-inpainting* result (32,71 dB). **Bottom - left**, *EM-inpainting* result using 6-scale Curvelets and LDCT with block size 32×32 (34,14 dB). **Bottom-right**, ℓ_0-GM result using 6-scale Curvelets (34,92 dB). 231
13.9. **Left**, detail of the result of *EM-inpainting* shown in Figure 13.8 (PSNR: 34,38 dB). **Right**, same for the method ℓ_0-GM (35,13 dB). 232

13.10 **Top-left**, damaged photographic image. **Top-right**, *Fast-inpainting* result. **Bottom-left**, *EM-inpainting* result using Curvelets and LDCT. **Bottom-right**, our result using ℓ_0-GM with 6-scale Curvelets and LDCT using 32×32 blocks and both scale factors to $\sqrt{0{,}5}$. 233
13.11 Visual example of the comparison between ℓ_1-GM and ℓ_0-GM applied to de-mosaicing. **Top-left**, 64×64 detail of image 15 of *Eastman Kodak* database. **Top-right**, result of de-mosaicing a Bayer mosaic with patter 'GB' using ℓ_1-GM with DT-CWT. PSNR for channels R, G and B is 37,24 dB, 39,87 dB and 37,17 dB. **Bottom-left**, ℓ_0-GM result using Curvelets (39,29, 42,27 and 38,20 dB). **Top-right**, ℓ_0-GM result using DT-CWT (39,59, 41,99 and 39,07 dB). 238
13.12 Visual comparison between de-mosaicing methods. **Top-left**, result of method in [149] with the image 15 of *Eastman Kodak* database. We have cropped the image to 64×64 to improve the visibility of artifacts. PSNR of R,G and B channels is, respectively, 39,81, 41,57 and 39,09 dB. **Top-right**, [148] result (39,89, 43,87 and 39,26 dB). **Bottom-left**, [150] result (40,39, 44,14 and 39,73 dB). **Bottom-right**, ℓ_0-GM result (39,59, 41,99, 39,07 dB). 241
13.13 **Top-left**, nearest neighbour interpolation (replication) of the resizing of the sub-sampling of the 2×2 averaged blocks *House* image (PSNR: 30,52 dB). We have cropped to 128×128 to improve visibility. **Top-right**, result of bilinear interpolation (29,65 dB). **Bottom-left**, result of ℓ_1-GM (33,50 dB). **Bottom-right**, result of ℓ_0-GM (33,53 dB). 245

A.1. Conjunto de imágenes de prueba utilizadas en esta Tesis. . . 254

B.1. Test set of images used in this Thesis. 256

Índice de cuadros

3.1. 47
3.2. 48
3.3. 49

4.1. 66
4.2. 67

6.1. 98
6.2. 106
6.3. 106
6.4. 117
6.5. 118
6.6. 119
6.7. 123

10.1. 174
10.2. 175
10.3. 176

11.1. 193
11.2. 194

13.1. 219
13.2. 229
13.3. 229
13.4. 237
13.5. 239
13.6. 240
13.7. 244

Capítulo 1

Introducción

1.1. Introducción y objetivos

El cerebro humano se ha adaptado a lo largo de su prolongada evolución, así como durante nuestro aprendizaje personal, para tratar los estímulos visuales de forma eficiente [1]. Debido a ello, hay una fuerte conexión entre el origen físico de estos estímulos y la estructura del sistema visual humano. Además, la visión es el más desarrollado de los sentidos del ser humano, en términos de la cantidad de información que permite adquirir y procesar por unidad de tiempo.

Por otra parte, los humanos siempre hemos necesitado transmitir información a nuestros semejantes. Para ello contamos con algunas herramientas limitadas al momento y lugar preciso donde se emite el mensaje, como el lenguaje fonético. Sin embargo, también existe la necesidad de comunicarse con un número mayor de personas, incluso aunque no estén presentes en el momento de emitir el mensaje. Esto se consiguió, en primer lugar, precisamente a través de estímulos visuales, como el lenguaje ideográfico y las pinturas. Véanse como ejemplos el arte prehistórico o la escritura simbólica.

Obtener una imagen de nuestro entorno consiste en proyectar el espacio de tres dimensiones en el que vivimos sobre una superficie de dos dimensiones, reproduciendo las formas de los objetos y sus detalles. Bajo el término *imagen natural*, denominamos a aquellas imágenes que se captan, típicamente mediante la fotografía, del mundo real, de forma que son similares a la información visual que suelen captar nuestros ojos. Lo sorprendente es que, para la correcta interpretación del mensaje, no es necesario que las imágenes naturales sean proyecciones perfectas de la realidad. Nuestro sistema visual puede detectar y reconocer los objetos representados aunque estén distorsionados, naturalmente hasta

cierto punto. Esta es una capacidad exclusiva de un sistema visual muy avanzado que explota de manera masiva esa información.

Hoy en día estamos asistiendo a una revolución tecnológica sin precedentes. Cada vez se procesa más información, que tiene que llegar a más gente. Una consecuencia natural del papel dominante de la visión en nuestra percepción, es que una de las áreas que más se está viendo afectada por esta revolución es la de las imágenes digitales. Durante los últimos años se ha producido un enorme desarrollo de las técnicas de captación, procesado, transmisión y almacenamiento de estas imágenes, que ya han sustituido a las analógicas como principal vehículo de representación. Gracias a las enormes posibilidades que ofrece la tecnología digital, las herramientas disponibles para manipular imágenes se han multiplicado de forma exponencial. Cada vez se exige más calidad sin sacrificar la velocidad de procesamiento, y, en consecuencia, es necesario un esfuerzo renovado tanto en el ámbito de la captación como en el del procesamiento posterior. Además, debido a la creciente importancia de las comunicaciones digitales, es cada vez más importante ahorrar ancho de banda, por lo que se busca maximizar la calidad visual para un soporte de información (número de bits) dado.

Al principio, problemas tales como la codificación de imágenes para compresión, el realce, la eliminación de ruido y artefactos molestos, la recuperación de información perdida, el reconocimiento de bordes, esquinas o formas en la imagen, etc. se aproximaban de forma heurística, mediante técnicas más o menos *ad-hoc*. Sin embargo, cada vez queda menos duda de la importancia de desarrollar buenos modelos sobre las imágenes que nos permitan una aplicación general a gran variedad de tareas.

La mayoría de estas aplicaciones tienen su reflejo en la visión humana. Son tareas que el cerebro realiza continuamente. Así que, para desarrollar un buen modelo, conviene preguntarse lo siguiente. ¿Cómo puede nuestro cerebro discriminar la información relevante en una imagen distorsionada?

Es obvio que cualquier imagen arbitraria no tiene por qué representar objetos del mundo que nos rodea, pues estos tienen una estructura típica que los hace reconocibles. Esta estructura se refleja en las imágenes naturales [1, 2], las cuales, típicamente, están compuestas de bordes localizados y zonas suaves relativamente amplias, posiblemente con alguna textura. El panel superior izquierdo en la Figura 1.1 es un ejemplo de una imagen natural típica. Por otro lado, las imágenes aleatorias, como la que se ve en el panel superior derecho de la Figura 1.1, no tienen, en general, ninguna estructura. Debido a la ingente cantidad de estímulos visuales procesados, y aprendidos a procesar durante la evolución, nuestro cerebro es capaz de distinguir muy claramente la imagen original que subyace tras una versión degradada de la misma (ver el panel inferior de la Figura 1.1, formado

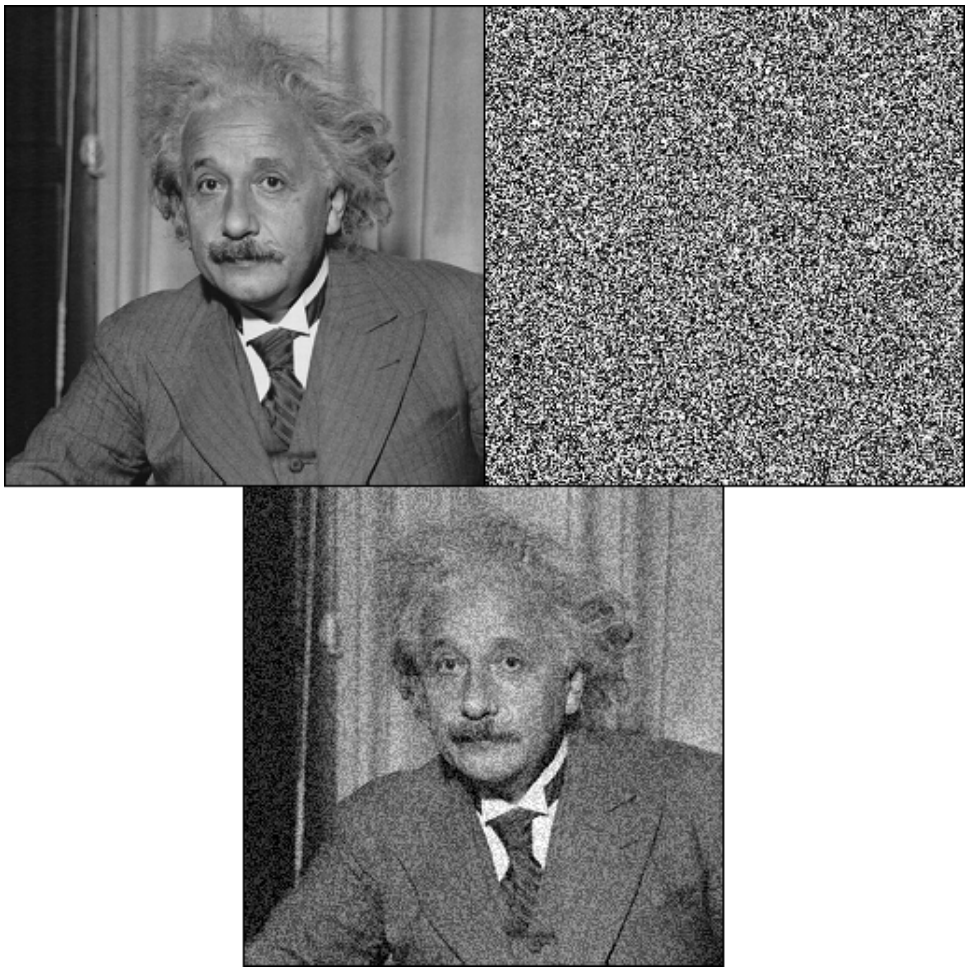

Figura 1.1: **Arriba - izquierda**, *imagen estándar* Einstein. **Arriba - derecha**, *imagen aleatoria (ruido blanco gaussiano).* **Abajo**, *suma de las imágenes anteriores.*

mediante una mezcla aditiva de los dos anteriores). Por consiguiente, si queremos realizar de manera automática este tipo de procesamiento, es de vital importancia contar con un buen conocimiento *a priori* sobre la estructura típica de las imágenes naturales, como muchos autores han resaltado (por ejemplo, [3, 4, 5]).

Uno de los criterios considerados normalmente cuando se evalúa la eficiencia de un sistema neurológico es maximizar la relación entre la cantidad de información y el número de neuronas necesarias para representarla [1, 5]. Igualmente, podemos plantearnos el mismo objetivo cuando tratamos con imágenes naturales. Si conseguimos representarlas con el menor número posible de números, estaremos facilitando no sólo su almacenamiento, sino también la consecución de descripciones estadísticas

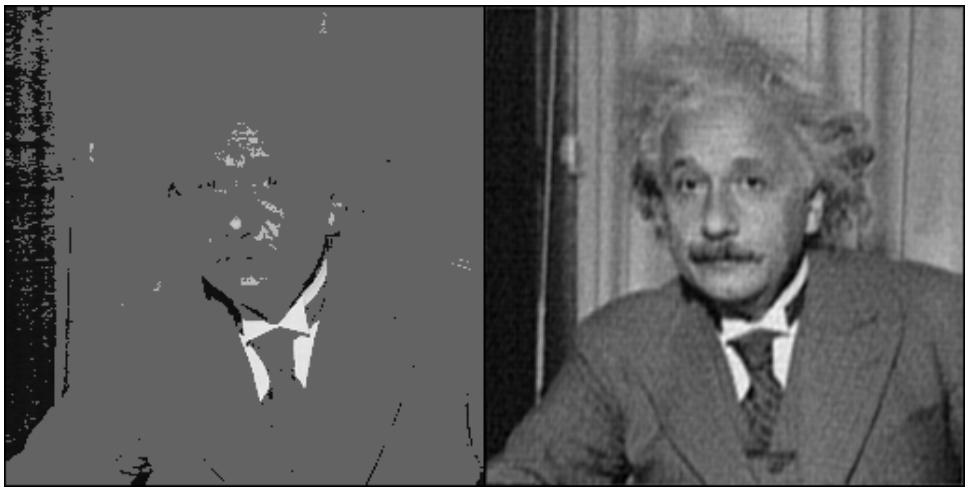

Figura 1.2: **Izquierda**, *imagen obtenida usando el* 10 % *de los píxeles de mayor desviación respecto a la media en la imagen* Einstein. **Derecha**, *imagen construida utilizando el* 10 % *de los coeficientes de mayor amplitud en la representación de Fourier de la misma imagen.*

más potentes. Por tanto, estaremos aumentando el rendimiento de las aplicaciones de restauración.

Lamentablemente, a pesar de la estructura típica descrita, el ingente tamaño del conjunto de imágenes naturales y la fuerte dependencia estadística que existe entre los píxeles cercanos entre sí hacen que el modelado sea una tarea demasiado compleja para llevarse a cabo en el dominio de los píxeles. La capacidad de acumular información en pocos elementos se puede ver considerablemente aumentada si transformamos los píxeles de la imagen a algún nuevo dominio. Volvamos al ejemplo mostrado en el panel izquierdo de la Figura 1.1. Si sólo nos quedamos con el 10 % de los píxeles de mayor desviación respecto a la media, obtenemos la imagen del panel izquierdo de la Figura 1.2, donde, como podemos ver, la mayoría de las características de la imagen original apenas son reconocibles. Por otro lado, el panel de la derecha de la Figura 1.2 muestra la imagen formada con el 10 % de las frecuencias cuyos coeficientes tienen más amplitud en el dominio de Fourier. Aunque se ha obtenido usando el mismo número de coeficientes, la segunda imagen está más próxima a la original que la primera, tanto en términos objetivos como subjetivos.

Esta propiedad también facilita, como hemos dicho, la descripción estadística de las imágenes naturales. Por ejemplo, típicamente éstas tienen amplias zonas de textura suave, por lo que la energía de su representación de Fourier estará concentrada en las bajas frecuencias. Pensemos en una degradación que altere de manera uniforme todas las frecuencias (por ejemplo, ruido blanco gaussiano). Aquellas que son dominantes se verán

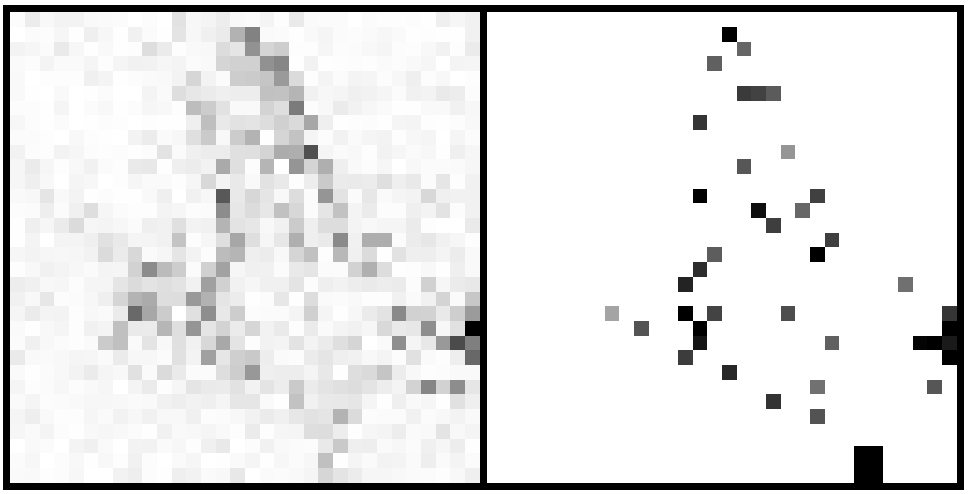

Figura 1.3: **Izquierda**, *sub-banda de la representación de la imagen* Peppers *bajo un banco de filtros de tipo ondícula (DT-CWT). Los píxeles oscuros representan coeficientes de alta amplitud y los claros aquellos con baja amplitud.* **Derecha**, *la misma sub-banda de una aproximación rala a la imagen.*

relativamente poco afectadas. Intuitivamente, vemos que eliminando las frecuencias de baja amplitud, muy afectadas pero poco importantes para reconstruir la imagen original, estaremos reduciendo significativamente la cantidad de ruido de la observación mientras mantenemos una alta fidelidad con la imagen original. Esta observación nos lleva a la consecuencia de que, cuánto más se concentre la energía en pocos coeficientes, la eliminación de aquellos de baja amplitud resultará en una mayor reducción de la cantidad de ruido de la imagen que si la energía se distribuye más uniformemente.

La RAE[1] define la raleza (*sparseness*, en inglés) como la propiedad de aquellas cosas cuyos componentes, partes o elementos están separados más de lo común en su clase. En esta Tesis hemos hecho un uso extendido de esta definición, interpretando el término como un concepto continuo en base a la concentración de la mayor parte de la energía de una señal discreta en una proporción pequeña de coeficientes. Para medir el grado de concentración utilizamos alguna norma del vector de la representación. En la Figura 1.3 puede verse un ejemplo de dos distribuciones de coeficientes en dos dimensiones. Mientras que el panel de la derecha muestra una distribución rala (*sparse*, en inglés), el de la izquierda reparte más la energía entre los coeficientes.

Además del dominio de Fourier, existen otras representaciones que potencian aún más el procesamiento eficiente y facilitan la descripción

[1]Real Academia Española.

estadística de las imágenes. Por ejemplo, las representaciones lineales basadas en filtros paso-banda multi-escala, denominadas genéricamente ondículas (*wavelets*, en inglés) están especialmente bien adaptadas para representar diversas propiedades de las imágenes naturales, como la invariancia a escala y la existencia de estructuras localmente orientadas. Se ha visto que las respuestas a este tipo de filtros de las imágenes naturales producen distribuciones típicamente ralas [3, 6].

Las representaciones redundantes, que son aquellas con más coeficientes que píxeles en la imagen, permiten un análisis y procesamiento de las imágenes más potente, comparadas con las representaciones críticamente muestreadas, porque aquellas favorecen la extracción de características locales relevantes [7, 5]. Además, siendo invariantes a la traslación, rotación, fase, etc. [8, 9], ofrecen mejore resultados de restauración (e.g., [7, 8]).

También se ha observado que se puede aumentar aún más la capacidad expresiva potencial utilizando representaciones redundantes, es decir, aquellas que usan más coeficientes que píxeles tiene la imagen en el dominio original (ver, por ejemplo, [10]). Además, siendo normalmente invariantes a la traslación, las representaciones redundantes proporcionan mejores resultados en diferentes aplicaciones de procesamiento de imágenes (ver, por ejemplo, [7, 11, 5]).

No obstante, la transformación directa a un dominio redundante no favorece el aumento de la raleza de la representación con respecto a los dominios críticamente muestreados (ver, por ejemplo [12], o el Capítulo 2 de esta Tesis). Necesitamos, adicionalmente, métodos no lineales que aumenten esta raleza. Definimos un diccionario como un conjunto de vectores, también llamados átomos. El *problema de representación rala* se define como encontrar la expresión de una señal de entrada como combinación lineal de tan pocos vectores de un diccionario dado como sea posible. En esta Tesis, trataremos con una variante usual, llamada *problema de aproximación rala*, que permite cierta tolerancia en la fidelidad a la señal de entrada. Así, podemos obtener representaciones muy sencillas con diferentes niveles de aproximación, muy útiles para codificación y para restauración de imágenes. El principal inconveniente que tradicionalmente se ha encontrado para resolver estos problemas es su tremenda complejidad [13], lo que se consideró, durante mucho tiempo, como un obstáculo demasiado severo en la práctica. Sin embargo, en los últimos años, y gracias al incremento espectacular en la velocidad de computación y a la mayor conciencia de la importancia de la raleza en el procesamiento de señales, estos métodos han sido objeto de gran interés. Así, se han desarrollado técnicas más o menos eficientes que encuentran soluciones sub-óptimas (por ejemplo [14, 13, 15, 16, 17, 18, 19]); comúnmente para utilizar la raleza como conocimiento *a*

priori en tareas de restauración (por ejemplo [20, 16, 21, 22, 23]).

Las tendencias actuales de este tipo de métodos siguen tres estrategias principales. La más antigua de ellas se basa en aplicar técnicas voraces (*greedy*, en inglés) para expresar las imágenes de forma incremental, utilizando aquellos vectores que mejor aproximen la parte de la imagen aún no representada (ver, por ejemplo, [3, 24, 25]). El principal problema de estas técnicas es que, siendo en general muy ineficientes, se quedan a menudo atrapadas en mínimos locales poco favorables en términos de compactación de energía. Hay técnicas voraces basadas en estrategias distintas de selección, como [26], donde, en un trabajo previo a esta Tesis, nosotros seleccionábamos los coeficientes significativos umbralizando directamente el vector de la representación observado.

La siguiente estrategia, en orden de aparición, está basada en reducir la complejidad del problema cambiando la búsqueda de raleza estricta por la minimización de la suma de los valores absolutos de los coeficientes de la representación. Esto resulta en la minimización de una función convexa. De hecho, se denomina *problema de relajación convexa*. Su resolución es, en general, una solución local a los problemas de representación o aproximación rala. Las primeras aproximaciones a este problema usaban algoritmos numéricos de optimización, como gradientes conjugados o métodos de punto interior (ver, por ejemplo, [27, 13, 28]). Sin embargo, estas técnicas han resultado a menudo ineficientes, sobre todo en problemas de muchas dimensiones como son, en general, los que interesan en procesamiento de imágenes.

Finalmente, en los últimos años se han venido desarrollando una serie de métodos eficientes y que están dando buenos resultados prácticos tanto en compactación como en su aplicación a problemas de restauración. Están basados en la combinación iterativa de operaciones lineales con operaciones de umbralización. Dependiendo de la umbralización utilizada, pueden dar soluciones directas al problema de aproximación rala (como en [15, 17, 29]), o bien resolver aproximaciones a él como, por ejemplo, el problema de relajación convexa (ver, por ejemplo, [16, 30, 31, 19]). Estos métodos han demostrado ya dar resultados satisfactorios en algunas tareas de restauración (por ejemplo, [16, 21, 22]) En el Capítulo 2 realizaremos una revisión más extensa de la literatura existente en este tema.

Se ha puesto mucho esfuerzo en encontrar las restricciones bajo las cuales los heurísticos voraces y la relajación convexa consiguen encontrar la solución óptima a los problemas de representación o aproximación rala (ver, entre otros, [32, 33, 34, 35]). Pero estas condiciones parecen ser demasiado restrictivas para que se cumplan en situaciones prácticas de procesamiento de imágenes naturales, usando representaciones típicas y niveles prácticos

de raleza. Esto se ha apuntado, por ejemplo, en [36].

1.2. Contribución de esta Tesis

En esta Tesis se derivan dos métodos iterativos, aunque relativamente eficientes, para resolver el problema de aproximación rala. Además de minimizar directamente la cuasi-norma[2] ℓ_0 para un error de aproximación dado, también hemos desarrollado la versión resultante de ambos métodos si se usa como criterio a minimizar la suma de los valores absolutos de los coeficientes (norma ℓ_1), como proponen los métodos de relajación convexa.

El primero de los métodos presentados se basa en reformular el problema de aproximación rala como encontrar, dados p y R, la mejor aproximación a la imagen dentro de la bola ℓ_p de radio R. Nuestra solución utiliza proyecciones alternas [37, 38] entre la bola ℓ_p y el conjunto de vectores que representan perfectamente la imagen. Nos centraremos en los casos $p = 0$, para el que obtendremos soluciones sub-óptimas, y $p = 1$, para el que se encontrará el óptimo global. Se pueden encontrar métodos similares en la literatura, tanto usando $p = 0$, donde se ha derivado de forma heurística en [17], como usando $p = 1$ [39, 40]. Llamamos ℓ_p-AP a este método.

El segundo método está basado en re-expresar la función de coste del problema de aproximación rala, que es discontinua y no restringida, para obtener una forma continua y restringida equivalente a ella. Realizando descenso en la dirección opuesta al gradiente de esta nueva función de coste, obtenemos una versión generalizada del método de umbralización dura iterativa (IHT, de *Iterative Hard Thresholding*, en inglés) [41]. Esta derivación nos sirve para demostrar que el punto fijo de las iteraciones de este método corresponde a un mínimo local del problema. A continuación, mostramos el método que proponemos, consistente en realizar descenso de gradiente sobre versiones cada vez menos suavizadas de la nueva función de coste. Este método, que denominamos ℓ_0-GM, coincide con otros usados anteriormente de forma heurística [42, 17, 21]. No obstante, esta es la primera vez que se deriva como solución a un problema de optimización. Estudiaremos también el mismo método aplicado al problema de relajación convexa, que denominamos ℓ_1-GM.

Hemos seguido una metodología muy rigurosa al plantear nuestros métodos como solución a problemas formulados como optimización de un criterio estándar bien definido. Sin embargo, en vez de buscar garantías de que los métodos desarrollados alcancen el óptimo global bajo ciertas

[2] Aunque la norma ℓ_p no es estrictamente una norma cuando $0 \leq p < 1$, en esta Tesis usamos este término para cualquier valor de p, en aras de la sencillez.

restricciones, como hacen otros autores (por ejemplo, [32, 33, 34]), en esta Tesis hemos buscado obtener buenos resultados en condiciones prácticas de procesamiento de imágenes. Frente a la asunción extendida de que los métodos basados en maximizar la raleza estricta son intratables, hemos comprobado que nuestros métodos sub-óptimos basados en minimización de la norma ℓ_0 ofrecen resultados muy superiores en la práctica a aquellos basados en minimizar criterios alternativos convexos (como la norma ℓ_1). Especialmente, ℓ_0-GM ofrece un rendimiento de compactación excelente, como otras técnicas parecidas basadas en ajuste dinámico del umbral, y superior no sólo a nuestro primer método, sino a las estrategias voraces usadas en la práctica y a las técnicas de relajación convexa. De hecho, veremos que asintóticamente alcanza un comportamiento muy cercano al óptimo, cuando el número de coeficientes activos en la representación se aproxima al número de píxeles de la imagen.

Además, el interés de nuestras aproximaciones se incrementa considerablemente al estudiar la aplicación de los métodos desarrollados a diferentes problemas de restauración de imágenes. Mostraremos resultados de muy alta calidad en una gran variedad de aplicaciones como eliminación de artefactos de cuantificación espacial (*de-quantizing*, en inglés), recuperación de píxeles perdidos (*in-painting*), interpolación espacial-cromática para mosaicos en cámaras digitales (*de-mosaicing*), o super-resolución estática. Que nosotros sepamos, es la primera vez que este tipo de métodos se aplica a la eliminación de artefactos de cuantificación espacial.

El contenido del documento se divide en los siguientes capítulos. En el Capítulo 2 se formula el problema de aproximación rala, motivándolo por la necesidad de aumentar la compactación de energía conseguida mediante transformaciones lineales. También se analizan con detalle las principales estrategias tradicionalmente usadas para solucionar este problema. A continuación, el Capítulo 3 desarrolla el primer método propuesto en esta Tesis, ℓ_p-AP. Nos centramos en los casos $p = 0$ y $p = 1$, y comparamos su rendimiento entre ellos y con respecto a otros métodos existentes en la literatura. En el Capítulo 4 derivamos el método IHT y demostramos que el punto fijo de sus iteraciones es un mínimo local del problema de aproximación rala. A continuación obtenemos el segundo método propuesto, denominado ℓ_0-GM. También se muestra la derivación análoga para $p = 1$, resultando el método ℓ_1-GM. En la sección de resultados se compara el rendimiento de ℓ_0-GM con el resto de métodos, y también se presentan las ventajas prácticas de ℓ_1-GM sobre otros métodos que obtienen el óptimo global al problema de relajación convexa. En el Capítulo 12 estudiaremos cómo adaptar los métodos presentados a problemas de restauración. Por

último, en el Capítulo 6 mostraremos los resultados de aplicación a la restauración de imágenes bajo varias degradaciones diferentes (ver arriba). El Capítulo 7 concluye esta Tesis.

Capítulo 2

El problema de aproximación rala

El problema de aproximación rala puede definirse como minimizar una medida del error cometido al aproximar una imagen como combinación lineal de un número limitado de átomos extraídos de un conjunto redundante (diccionario). Mediremos esta distorsión de forma cuantitativa utilizando el error cuadrático medio (MSE, de *Mean Square Error*, en inglés). En este capítulo mostramos que, en un dominio redundante, existen infinitas formas distintas de representar una imagen. Tradicionalmente se ha escogido la solución de mínima energía, porque es fácil de calcular linealmente. Sin embargo, los métodos no lineales tienen un potencial mucho mayor a la hora de compactar la energía en pocos coeficientes. Este tipo de métodos han sido ampliamente usados y se han mostrado muy útiles para tareas de restauración.

En la Sección 2.1 motivaremos el uso de métodos no lineales para la obtención de representaciones ralas en dominios redundantes. A continuación formularemos el problema de aproximación rala en la Sección 2.2, para describir los métodos más importantes que se han usado hasta ahora para su resolución en la Sección 2.3. Por último, en la Sección 2.4 analizaremos las condiciones bajo las cuales el problema de aproximación rala puede ser resuelto de forma óptima utilizando optimización convexa y heurísticos voraces.

2.1. Raleza de análisis y raleza de síntesis

Matemáticamente, representar una imagen como combinación lineal de los vectores de un diccionario significa resolver un sistema de ecuaciones lineales con más ecuaciones que incógnitas. Existen, por lo tanto, infinitas

soluciones. ¿Cómo escoger entre ellas? La solución más común es la de mínima norma euclídea, que es una solución lineal, y por tanto fácil y rápida de calcular. De hecho, las transformaciones "inversas"típicamente usadas en procesamiento de imágenes para representarlas minimizan la norma euclídea (a través de una pseudo-inversa). Sin embargo, como hemos discutido en la Introducción, hay buenas razones para buscar soluciones que concentren la energía en tan pocos coeficientes como sea posible. Estas soluciones expresan las imágenes, posiblemente permitiendo cierto nivel de error, como combinación lineal de un número menor de vectores del diccionario que las soluciones lineales. Aunque algunos autores han observado que las respuestas lineales de los bancos de filtros de ondículas a las imágenes naturales ya concentran la mayor parte de la energía en relativamente pocos coeficientes (ver, por ejemplo, [3, 43, 44, 6]), lo cierto es que la solución de mínima norma euclídea tiende a repartir la energía entre los coeficientes tanto como sea posible, por lo que no es adecuada para la representación rala. A continuación ilustramos que la compactación de la energía en diccionarios redundantes es mucho mayor en ciertas transformaciones no lineales de la imagen.

La Figura 2.1 muestra la fidelidad (en dB) a la imagen original obtenida al aproximarla usando diferentes representaciones para un amplio rango de niveles de raleza, es decir, de número de vectores involucrados en la aproximación. Los datos están promediados para las cinco imágenes de nuestro conjunto de prueba (ver Apéndice A). La fidelidad a la imagen original se mide utilizado la relación señal-ruido pico (PSNR, de *Peak Signal-to-Noise Ratio*, en inglés), que se define como $10 \cdot \log_{10}(\frac{\rho^2}{MSE})$ y se mide en decibelios (dB). El parámetro ρ es la amplitud máxima de los coeficientes de las señales involucradas. Como aquí tratamos con imágenes monocromas de 8 bits, en nuestro caso $\rho = 255$. En esta Tesis, normalmente representamos el número de coeficientes activos normalizado por el número total de píxeles de la imagen. Cada curva de la Figura 2.1 ha sido obtenida reconstruyendo la imagen usando los coeficientes de mayor amplitud de cada transformación lineal, para diferentes niveles de raleza. Tres de las representaciones usadas están críticamente muestreadas (píxeles, Fourier y Ondículas de Haar), mientras que la cuarta es redundante (DT-CWT, siglas de Ondículas Complejas de Árbol Dual o *Dual-Tree Complex Wavelet*, en inglés [45]).

Podemos ver cómo la calidad de la aproximación para un número dado de funciones elementales crece cuando transformamos los píxeles al dominio de Fourier, y crece aún más cuando usamos ondículas críticamente muestreadas. Desafortunadamente, el rendimiento de la representación lineal redundante sufre una brusca caída, debido a que hay muchos

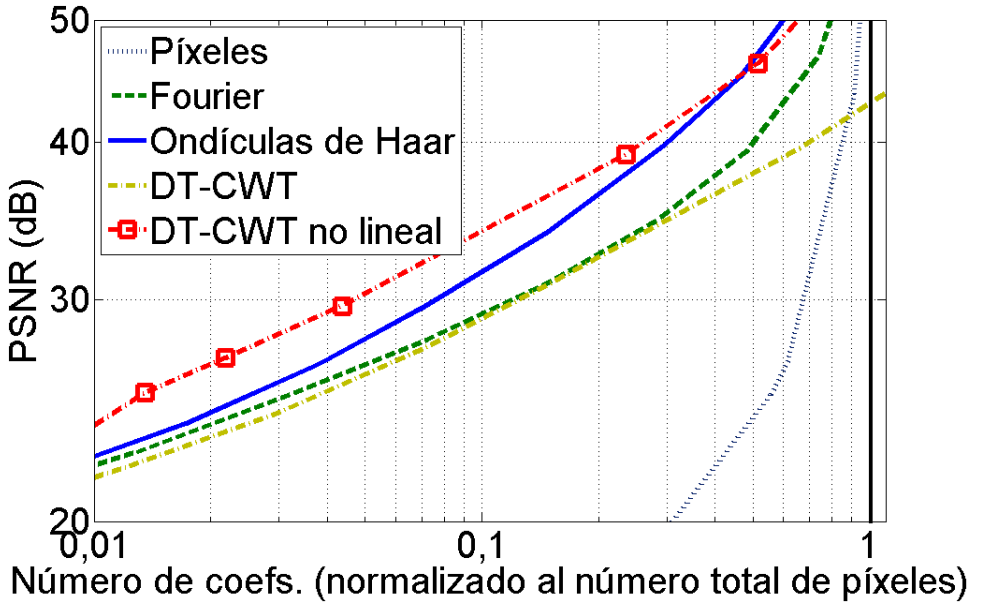

Figura 2.1: *Comparación de resultados de aproximación rala obtenidos usando los coeficientes de mayor amplitud de las respuestas lineales de diferentes representaciones y promediados en nuestro conjunto de prueba. El rendimiento se mide en función de la PSNR obtenida usando un cierto número de coeficientes, que aparece normalizado por el número total de píxeles en la imagen.*

coeficientes que se originan como respuesta a la misma característica de la imagen, lo que produce un descenso en la raleza. Entonces, como ya hemos apuntado, si se quiere incrementar el rendimiento de la aproximación con respecto a las ondículas críticamente muestreadas, se necesita utilizar un mecanismo no lineal de selección de átomos. En la Figura 2.1 también se muestra el resultado obtenido usando DT-CWT con el mejor método propuesto en esta Tesis (ℓ_0-GM, ver Capítulo 4). Se puede comprobar que proporciona una gran mejora en compactación sobre las aproximaciones obtenidas desde las representaciones lineales. La necesidad de mecanismos no lineales de selección para obtener representaciones que concentren más la energía ya ha sido tratada por varios autores, como por ejemplo [28, 13, 46, 47, 48, 49].

La Figura 2.2 ilustra el efecto de usar este tipo de mecanismos de selección transformando la imagen *Peppers* con DT-CWT. El panel superior izquierdo muestra los coeficientes de una sub-banda de la representación lineal con de la imagen. En el panel central se muestran los coeficientes de una representación obtenida de manera no lineal (utilizando ℓ_p-AP, ver Capítulo 3) con objeto de aumentar la concentración de la energía en pocos coeficientes. En el primero se ve una distribución menos rala que

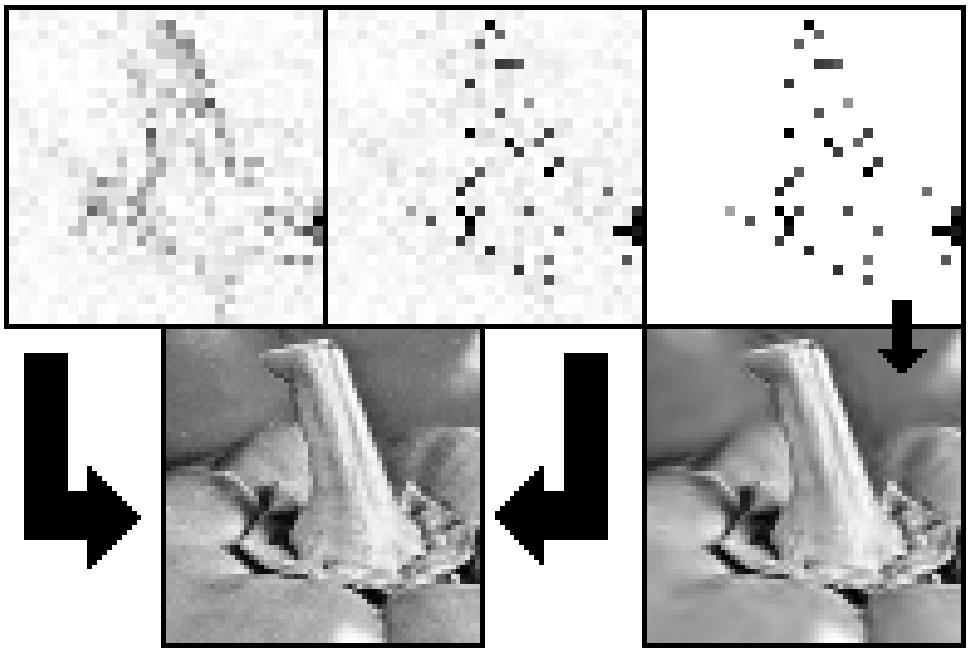

Figura 2.2: **Arriba-izquierda**, *sub-banda de la escala de más alta frecuencia de la representación de análisis de* Peppers *usando DT-CWT con* 8 *escalas. Los puntos claros y oscuros corresponden respectivamente con amplitudes pequeñas y grandes de los coeficientes. El tamaño de la sub-banda ha sido duplicado en ambas direcciones mediante replicado de filas y columnas para coincidir con el tamaño de la imagen y luego se ha recortado a* 64×64 *para conseguir mejor visibilidad.* **Arriba-centro**, *la misma sub-banda, pero esta vez obtenida no linealmente con el método* ℓ_0*-AP propuesto en esta Tesis (ver Capítulo 3).* **Arriba-derecha**, *resultado de aplicar un umbral a las amplitudes del resultado del panel central (preservando* 7 *veces menos coeficientes que píxeles tiene la imagen).* **Abajo-izquierda**, *imagen original, que es reconstruida perfectamente por los coeficientes que corresponden con los paneles izquierdo y central de la fila superior.* **Abajo-derecha**, *aproximación obtenida con los coeficientes ralos del panel superior-derecho (*35,67 *dB).*

en el segundo, que disminuye fuertemente las respuestas simultáneas a una misma característica. Esta mejor compactación hace posible que, utilizando solamente una pequeña proporción del número total de coeficientes (alrededor del 7 veces menos coeficientes que número de píxeles de la imagen, en este caso), se mantenga una alta calidad en la reconstrucción (35,67 dB en este ejemplo, véase panel derecho).

2.2. Formulación del problema de aproximación rala

Formularemos a continuación el problema de aproximación rala. Sea $\mathbf{\Phi}$ una matriz de tamaño $N \times M$ con $M > N$ y $rango(\mathbf{\Phi}) = N$, que representa el operador de síntesis de un marco ajustado de Parseval[1] (*Parseval tight-frame*, en inglés). Esta matriz es nuestro diccionario, de forma que cada una de sus columnas es un átomo (vector). N es el número de píxeles en el dominio original y M el número de coeficientes en el dominio transformado. Entonces, para una imagen dada, $\mathbf{x} \in \mathbb{R}^N$, el sistema lineal de ecuaciones:

$$\mathbf{\Phi a} = \mathbf{x}, \tag{2.1}$$

tiene infinitas soluciones en $\mathbf{a} \in \mathbb{R}^M$. Si deseamos estrechar el cerco sobre una solución concreta, debemos añadir criterios adicionales. Así que podemos introducir una función $f(\mathbf{a})$ que evalúe la idoneidad de cada solución, de forma que el problema queda planteado como:

$$\hat{\mathbf{a}}^f = \arg \min_{\mathbf{a} \in \mathbb{R}^M} f(\mathbf{a}) \text{ s.a. } \mathbf{\Phi a} = \mathbf{x}. \tag{2.2}$$

De entre las posibles opciones para $f(\mathbf{a})$, a menudo se ha utilizado la p-ésima potencia de la norma ℓ_p, que para un valor dado de p se define como $\|\mathbf{a}\|_p^p = \sum_{i=1}^{M} |a_i|^p$. En la Figura 2.3 se muestra la forma de esta función, en su versión unidimensional, para varios valores de p. Como hemos dicho, la norma comúnmente más usada ha sido, sin duda, la euclídea [50], $p = 2$, dando lugar a la solución de mínima energía, $\mathbf{a}^{LS}$. Esta solución es especialmente fácil de calcular para marcos de Parseval. De hecho, si $\mathbf{\Phi}^T = \mathbf{\Phi}^T[\mathbf{\Phi}\mathbf{\Phi}^T]^{-1}$ es la pseudoinversa de $\mathbf{\Phi}$, que corresponde con el operador de análisis del marco de Parseval, entonces tenemos que $\mathbf{a}^{LS} = \mathbf{\Phi}^T\mathbf{x}$. No obstante, como hemos visto en la sección anterior, esta solución no es adecuada en términos de maximizar la raleza. Idealmente, ésta se mide con la norma ℓ_0, que se expresa por extensión de la definición de norma como el número de coeficientes distintos de cero en el vector. Así, el problema de representación rala queda planteado como:

$$\hat{\mathbf{a}}^0 = \arg \min_{\mathbf{a}} \|\mathbf{a}\|_0 \text{ s.a. } \mathbf{\Phi a} = \mathbf{x}. \tag{2.3}$$

Sin embargo, los diccionarios redundantes típicamente usados en procesamiento de imágenes no permiten representaciones de las imágenes

[1] Una transformación lineal con un marco ajustado de Parseval conserva la norma euclídea del vector original. Utilizaremos el término marco de Parseval por sencillez.

naturales donde la mayoría de los coeficientes sean realmente cero. Es más realista buscar representaciones que concentren la mayor parte de la energía en la menor proporción posible de coeficientes, de forma que la mayoría de ellos tengan amplitudes relativamente pequeñas. Este tipo de distribuciones, como hemos ejemplificado en la sección anterior, tienen la propiedad de que unos pocos coeficientes de alta amplitud aproximan la imagen de forma aceptable para muchas aplicaciones. Es por ello que muchos autores prefieren relajar la restricción de la Ecuación (2.3), formulando el problema de aproximación rala como:

$$\hat{\mathbf{a}}^0(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}, \tag{2.4}$$

donde $\lambda \in \mathbb{R}^*$ es un número real positivo que controla la importancia relativa entre los términos de raleza y fidelidad de la función de coste, de forma que mientras mayor sea su valor, la solución tendrá menos error, a costa de reducir la raleza. La Ecuación (2.4) es equivalente o bien a minimizar $\|\mathbf{a}\|_0$ para un error cuadrático dado (que depende de λ) o a minimizar el error cuadrático para una norma ℓ_0 de la aproximación dada (que también depende de λ). La Ecuación (2.3) es un caso particular de la Ecuación (2.4), cuando λ tiende a infinito. Como ejemplo, la sub-banda mostrada en el panel central de la Figura 2.2 puede entenderse como perteneciente a la solución de (2.4) cuando λ es infinito, donde se reconstruye perfectamente la imagen sacrificando la raleza. Por otro lado, la sub-banda del panel derecho correspondería a la solución con un valor de λ más pequeño, y que utiliza menos coeficientes a costa de reducir la calidad de la reconstrucción de la imagen.

2.3. El problema de aproximación rala en la literatura

Encontrar el óptimo global del problema de aproximación rala es un problema combinatorial, y, por tanto, NP-complejo [13, 51, 46, 34]. Su solución requiere explorar cada combinación posible de columnas de $\mathbf{\Phi}$, y resolver por mínimos cuadrados, eligiendo el que ofrezca el menor MSE. Es por esto que siempre se ha buscado atacar el problema a través de alguna aproximación más eficiente. Algunas de estas primeras aproximaciones se restringían a diccionarios concretos (por ejemplo, paquetes de ondículas o funciones trigonométricas localizadas) donde es posible extraer la base ortonormal que represente más eficientemente a la imagen [43]. Pero durante las últimas dos décadas se han popularizado métodos más efectivos y generales para encontrar soluciones aproximadas tanto al problema (2.3)

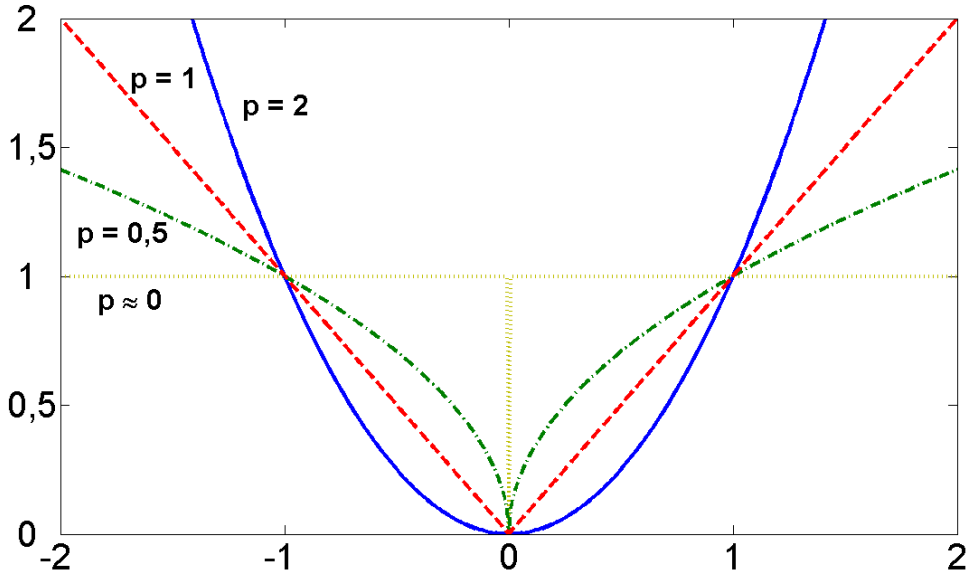

Figura 2.3: *Forma de la p-ésima potencia de la norma ℓ_p en una dimensión para varios valores de p.*

como al (2.4). Podemos clasificar estas técnicas en tres grandes grupos: heurísticos voraces, métodos basados en relajación convexa y métodos basados en umbralización iterativa. A continuación haremos un repaso a la literatura más importante existente sobre estos tres tipos de técnicas.

2.3.1. Heurísticos voraces

Las primeras estrategias para obtener soluciones locales a la Ecuación (2.4) con diccionarios generales fueron derivadas de forma heurística. De entre ellas, la más utilizada proviene de la observación de que es conveniente, para predecir un vector dado y dado un determinado diccionario, seleccionar aquél vector que tenga máxima correlación con el vector objetivo (ver, por ejemplo, [52]).

Esta familia de algoritmos es bien conocida y ampliamente usada. De hecho, estos métodos han sido re-inventados en varios campos. En modelado estadístico, se les llama regresión múltiple por pasos (*forward stepwise regression*, en inglés), y se han usado desde los años 60 (ver, por ejemplo, [53, 54] y sus referencias). Cuando se usa en procesamiento de señal, se usan los términos *Matching Pursuit* (MP) [14] y *Orthogonal Matching Pursuit* (OMP) [24], entre otros. En teoría de aproximación se refieren a ellos como Algoritmos Voraces (*Greedy*, en inglés) [55, 56, 57, 58]. Una extensa revisión de estos métodos, aplicados a aproximación no lineal,

puede encontrarse en [59].

En nuestro contexto, MP es el método voraz más básico. Se implementa a través de un conjunto de índices, I, que indica los átomos del diccionario que han sido seleccionados para formar la aproximación, y un residuo, $\mathbf{r}$, o parte de la imagen aún no representada. El conjunto de índices se inicializa vacío, $I^{(0)} = \emptyset$, y el residuo a la imagen completa, $\mathbf{r}^{(0)} = \mathbf{x}$. En cada iteración $k+1$, se actualiza la base seleccionada añadiendo aquél átomo con máxima correlación con el residuo:

$$I^{(k+1)} = I^{(k)} \bigcup \{i : \langle \phi_i, \mathbf{r}^{(k)} \rangle \geq \langle \phi_j, \mathbf{r}^{(k)} \rangle, \forall j \neq i, j \notin I^{(k)}\},$$

donde $\langle \cdot, \cdot \rangle$ indica el producto interno de dos vectores. La estimación se actualiza como:

$$\hat{\mathbf{x}}^{(k)} = \sum_{j=1}^{k} \langle \phi_{i(j)}, \mathbf{r}^{(j)} \rangle \phi_{i(j)},$$

donde $i(j)$ representa el índice escogido en la iteración i. El siguiente paso es actualizar el residuo:

$$\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} - \langle \phi_i, \mathbf{r}^{(k)} \rangle \phi_i.$$

Las iteraciones paran o bien cuando se haya alcanzado el nivel de error deseado, o bien cuando se hayan seleccionado el número de términos requeridos. El principal problema que presenta MP es que, dado el conjunto de átomos seleccionados en cada iteración, las amplitudes de los coeficientes no están optimizadas para representar de forma óptima la imagen. OMP añade un paso intermedio para optimizar por mínimos cuadrados los coeficientes correspondientes a los átomos seleccionados en cada iteración. Dado el subconjunto de índices seleccionado en la iteración k, $I^{(k)}$, definimos la matriz $\mathbf{\Phi}_{I^{(k)}}$, de tamaño $N \times k$, formada por las columnas ϕ_i de $\mathbf{\Phi}$ tales que $i \in I^{(k)}$. Entonces, en cada iteración de OMP, tras la actualización del conjunto de índices, se resuelve el problema:

$$\hat{\mathbf{a}}_I^{(k)} = \arg \min_{\mathbf{a}_I \in \mathbb{R}^k} \|\mathbf{x} - \mathbf{\Phi}_{I^{(k)}} \mathbf{a}_I\|_2.$$

La actualización del residuo será ahora: $\mathbf{r}^{(k)} = \mathbf{x} - \mathbf{\Phi}_{I^{(k)}} \hat{\mathbf{a}}_I^{(k)}$. Este método no sólo da mejores resultados de compactación, sino que converge más rápidamente que MP [24]. Sin embargo, como ocurre a menudo con las estrategias voraces, OMP se queda atrapado en óptimos locales que con frecuencia no son lo suficientemente favorables para el problema planteado. Se han propuesto múltiples modificaciones heurísticas a OMP que se basan,

sobre todo, en búsquedas a través de árboles jerarquizados o recursivas que intentan explorar el mayor número de posibles elecciones en cada paso (ver, por ejemplo, [60, 61, 62, 63]). Otra desventaja de OMP, que es incluso más acusada en las variantes mencionadas, es que la selección de un único coeficiente en cada paso del algoritmo resulta impracticable en términos de tiempo de computación para los diccionarios usados más a menudo en procesamiento de imágenes. Existen otro tipo de variantes eficientes de OMP que seleccionan más de un coeficiente en cada paso, ya sea con un tamaño de paso fijo (por ejemplo, [64]) o variable (por ejemplo, en [65, 25]). En esta Tesis, denominaremos este tipo de métodos bajo el nombre del que quizá sea el más extendido de ellos, *Stagewise OMP* (StOMP) [25].

OMP ha sido aplicado con diferente éxito a varias aplicaciones, por ejemplo en eliminación de ruido [23, 66], codificación de video [67, 68, 69], compresión de imágenes en color [70] o separación de señales de audio [71, 58]. Sin embargo, todos estas aplicaciones de OMP o bien necesitan pocos pasos del algoritmo o bien utilizan diccionarios ortogonales entre sí. Las únicas técnicas voraces aplicables en la mayoría de situaciones prácticas para el caso de diccionarios redundantes típicamente usados en procesamiento de imágenes son de tipo StOMP.

En un trabajo previo a esta Tesis [26], presentamos un método para eliminar artefactos de cuantificación espacial mediante la búsqueda de una aproximación rala de la observación. En el método resultante, los coeficientes significativos se seleccionaban directamente umbralizando las amplitudes del vector $\mathbf{a}^{LS}$, lo que puede verse como una estrategia voraz.

2.3.2. El problema de relajación convexa y *Basis Pursuit*

Como hemos visto, utilizar la norma euclídea para evaluar las posibles soluciones (aproximadas o no) al sistema de ecuaciones lineales (2.1) no es eficiente en términos de conseguir soluciones ralas. En el otro extremo, la norma ℓ_0 no es convexa y ni siquiera continua, lo que convierte al problema en difícil y costoso. Este inconveniente, en la práctica, no queda bien resuelto por las estrategias voraces. ¿Y si intentamos encontrar un camino intermedio que nos permita aprovechar las ventajas de uno y otro enfoque? La respuesta para algunos casos nos la puede proporcionar la norma ℓ_1. Como no es diferenciable, favorece las soluciones ralas; pero, debido a que es convexa, se puede encontrar con relativa facilidad el óptimo global del problema. Por otro lado, como las técnicas resultantes permiten optimizar todos los coeficientes a la vez, se pueden obtener niveles de aproximación inalcanzables en la práctica para OMP. Llamaremos a esto el problema de

relajación convexa, que queda entonces formulado, de forma análoga a la Ecuación (2.4), como:

$$\hat{\mathbf{a}}^1(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}. \tag{2.5}$$

Este problema se ha formulado frecuentemente, desde los años 50, en términos de programación lineal (LP). Por ejemplo, en [27] se plantea un método *Simplex* para obtener representaciones de mínima norma ℓ_1 formulando el problema como un LP. Sin embargo, no fue hasta que el incremento en la velocidad de los microprocesadores y en la capacidad de memoria permitió el uso de técnicas más avanzadas, cuando se empezó a explotar de manera sistemática esta técnica.

Los métodos *Basis Pursuit* (BP) y *Basis Pursuit Denoising* (BPDN) [13, 46] resuelven respectivamente el problema de representación y el de aproximación mediante la resolución de un LP equivalente a la Ecuación (2.5), utilizando Métodos de Punto Interior (*Interior-Point Methods*, en inglés). Este trabajo ha tenido tanta difusión que hoy en día se utiliza asiduamente los términos BP y BPDN como sinónimos del problema de relajación convexa.

La aparición de BP y BPDN estuvo precedida de dos avances significativos: 1) el sorprendente descubrimiento de que se puede estimar casi óptimamente una función suave degradada resolviendo un problema de relajación convexa utilizando la base de ondículas apropiada y un parámetro λ relacionado con la varianza del ruido [72]; y 2) la aparición de LASSO, que propuso la relajación convexa como planteamiento para resolver el problema de selección de subconjuntos en regresión lineal [73]. La técnica conocida como *Least Angle Regression* [65] fue adaptada más tarde para resolver la formulación LASSO. El gran inconveniente de esta y otras técnicas (por ejemplo, FOCUSS [28]) es que necesitan manejar explícitamente la matriz. Esto no es practicable en muchas ocasiones en procesamiento de imágenes, donde se manejan matrices que habitualmente tienen cientos de miles de filas y columnas. Sin embargo, sí tenemos disponibles formas muy eficientes para realizar productos de la matriz con un vector para las representaciones típicamente usadas en procesamiento de imágenes. Tanto BP como los métodos iterativos que veremos en la siguiente sección aprovechan estas herramientas. Una breve pero completa historia de las diferentes aproximaciones al problema de relajación convexa puede encontrarse en [74].

Aparte de la codificación, las principales áreas donde se han aplicado este tipo de métodos son la regresión estadística [73, 28] y la eliminación de ruido [13, 46], aunque es posible encontrar otras aplicaciones como, por ejemplo, la recuperación de píxeles perdidos [75].

Recientemente se ha generado mucha expectación con una nueva aplicación para el problema de relajación convexa, bajo el nombre de Sensores Compresivos (*Compressed Sensing*, en inglés, ver, por ejemplo [76, 77]). Esta técnica está basada en la observación de que un número relativamente pequeño de proyecciones aleatorias de una señal rala son suficientes para recuperar desde ellas una buena aproximación de esta señal [78]. Se ha planteado como una potente alternativa al teorema tradicional de muestreo de Nyquist, cuando la señal puede expresarse como combinación lineal de pocos elementos.

Es también posible utilizar cuasi-normas intermedias ($0 < p < 1$) para regularizar el problema (2.2) (ver, por ejemplo, [79, 16]). Aunque estas normas no llevan a funciones de coste convexas, y por lo tanto es difícil obtener el óptimo global, varios autores han resaltado el hecho de que la distribución marginal de los coeficientes de la transformación lineal de una imagen natural bajo un banco de ondículas se modela apropiadamente con este tipo de normas (ver, por ejemplo, [80, 20]). Este acercamiento al problema de aproximación rala es muy interesante, pero su estudio queda fuera de los objetivos de esta Tesis.

2.3.3. Umbralización iterativa

Ya se ha comentando que los métodos voraces no son del todo satisfactorios para aplicarlos a imágenes naturales, porque o bien son demasiado costosos en términos de tiempo de computación, o bien quedan atrapados en mínimos locales poco favorables. Por otra parte, los métodos aplicables a procesamiento de imágenes basados en resolver el problema de relajación convexa mediante técnicas clásicas de optimización requieren demasiado tiempo de computación para imágenes de tamaño usual.

Durante los últimos años se han desarrollado nuevas técnicas eficientes para resolver el problema (2.4) basadas en aplicar iterativamente operaciones de umbralización combinadas con operaciones lineales. Por umbralización entendemos aquellas operaciones que decrecen la amplitud de los coeficientes de la representación, posiblemente dejando a cero aquellos que estén por debajo de un umbral en amplitud. Llamamos a estas técnicas Métodos de Umbralización Iterativa (*Iterative Shrinkage Methods*, en inglés)[2].

Los métodos de umbralización iterativa han sido ampliamente usados. Por ejemplo, [81] ya utilizaba técnicas similares para segmentación de imágenes. En [8], se demuestra que, en presencia de observaciones

[2]Otro posible nombre es Iteraciones de Landweber Umbralizadas (*Thresholded Landweber Iteration*, en inglés) [40].

perturbadas por ruido blanco gaussiano, el sencillo método de aplicar una umbralización (dura o suave) a una transformación ortogonal lineal de la imagen consigue, para algunos modelos de señal, un resultado óptimo en sentido de error cuadrático medio.

La optimalidad de estas umbralizaciones depende de la ortogonalidad de la matriz del diccionario[3]. Pero la umbralización, por sí sola, no ofrece soluciones óptimas para el caso de usar transformaciones redundantes. Aún así, se ha practicado con asiduidad para todo tipo de representaciones redundantes (ver [82] y sus referencias). La umbralización, dentro de un esquema iterativo, puede ofrecer soluciones aproximadas de una forma bastante eficiente.

Más recientemente, otros autores [80, 83, 84] simplificaron las ideas de [72, 8] y, a partir de una formulación bayesiana para hallar el *Máximo A Posteriori* (MAP), le dieron al problema la estructura similar de la Ecuación (2.2), cuando se minimiza la p-ésima potencia de una norma ℓ_p:

$$\hat{\mathbf{a}}^0(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_p^p + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}. \tag{2.6}$$

El método de umbralización iterativa, que se aplica frecuentemente para resolver este problema, puede describirse según las siguientes iteraciones:

$$\mathbf{a}^{(k+1)} = S_p\left(\mathbf{a}^{(k)} + \mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi a}^{(k)}), \theta\right), \tag{2.7}$$

donde $S_p(\mathbf{a}, \theta)$ indica una cierta operación de umbralización sobre un vector $\mathbf{a}$ con un umbral θ, que en nuestro caso es función del parámetro λ. La proyección lineal que sirve como argumento a la umbralización proviene de minimizar el término de fidelidad de la Ecuación (2.6) para un vector $\mathbf{a}$ dado, y es la proyección ortogonal del vector sobre el espacio afín de reconstrucción perfecta (ver Sección 3.1).

El primer método del que tenemos conocimiento que propone una técnica de umbralización iterativa para el problema (2.6) fue [79, 16], que utiliza normas ℓ_p con $0 < p \leq 1$ con una formulación basada en *Expectation-Maximization* (EM). Más tarde, en [30] se deriva, para el caso $p = 1$, el mismo método a través de términos extra añadidos a la función de coste sin cambiar la localización de sus mínimos. También [82] deriva un algoritmo similar para el caso $p = 1$ desde una perspectiva diferente, buscando maximizar la raleza de la representación. El método introducido en estos trabajos se basa en alternar una umbralización suave con una proyección lineal. Comúnmente, se le denomina como método de umbralización suave iterativa (de *Iterative Soft-Thresholding*, IST).

[3] Además depende también del uso de la norma euclídea para el término de error.

Corresponde a iterar con (2.7) usando la umbralización suave, definida como $S_1(\mathbf{a}, \theta) = \mathbf{b}$, donde:

$$b_i = \begin{cases} sign(a_i) \cdot (|a_i| - \theta), & |a_i| > \theta \\ 0, & |a_i| \leq \theta. \end{cases} \tag{2.8}$$

Aquí, $sign(\cdot)$ indica la función signo. El valor del umbral resulta en $\theta = \frac{1}{2\lambda}$. La convergencia del método fue demostrada en [30, 39]. Estas técnicas también se han usado para separar los componentes morfológicos de la señal, utilizando varios diccionarios con la propiedad de que cada uno de ellos es susceptible de representar de forma rala un tipo distinto de señales de entrada (Análisis en Componentes Morfológicos, o MCA, de *Morphological Component Analysis*, en inglés) [48, 21]. Otros autores también han derivado métodos similares desde distintas perspectivas (por ejemplo, [85]). Ver [86] para una revisión de los métodos basados en umbralización iterativa.

Existen otros algoritmos iterativos diferentes a IST, como el descrito en [87], que usa la umbralización suave como una operación de descenso en la dirección opuesta al gradiente, utilizando una búsqueda lineal para hallar el tamaño de paso óptimo en cada iteración. Otros ejemplos son [83], que aplica umbralización suave en representaciones redundantes dentro de una formulación variacional para eliminar ruido y para compresión; o la aplicación de la técnica genérica de optimización llamada Mínimos Cuadrados Re-ponderados Iterativos (*Iterative Re-weighted Least Squares*, en inglés) [88], que reformula la Ecuación (2.5) como un problema de programación cuadrática (requiere manipulación directa de la transformada en forma de una matriz de grandes proporciones). Queremos destacar de entre ellos el método de descenso de gradiente proyectado propuesto recientemente en [40], que cambia el umbral fijo por un umbral adaptado en cada iteración, para que la norma ℓ_1 del vector umbralizado sea constante. Este es un método similar al derivado en el Capítulo 3 de esta Tesis, usando la norma ℓ_1. Además, aplica después un tamaño de paso optimizado para el descenso de gradiente, con el objetivo de acelerar la convergencia.

Se puede derivar un método alternativo a IST utilizando umbralización dura, al que se conoce como IHT (de Umbralización Dura Iterativa, o *Iterative Hard Thresholding*, en inglés). En este caso, la operación utilizada en las iteraciones (2.7) es $S_0(\mathbf{a}, \theta) = \mathbf{b}$, con:

$$b_i = \begin{cases} a_i, & |a_i| > \theta \\ 0, & |a_i| \leq \theta. \end{cases} \tag{2.9}$$

Y el valor del umbral resulta en $\theta = \lambda^{-\frac{1}{2}}$. El primero en proponer este método fue [15], aunque se planteaba como un heurístico sin justificación

teórica. Además, se plantea de forma heurística en otros trabajos, como [41]. En [29] se ha derivado utilizando funciones surrogadas y, además, se ha demostrado su convergencia a un mínimo local del problema de aproximación rala[4].

En [17] se plantea un método similar, pero que en vez de usar un umbral fijo, lo que fija es el número de coeficientes que quedan activos tras cada umbralización. Este es un método similar al derivado en el Capítulo 3 de esta Tesis, usando la norma ℓ_0, pero aquí lo obtenemos como solución a un problema de optimización. Una mejora reciente a este método [89] utiliza diferentes umbrales para cada sub-banda de la representación en función de la energía de cada una.

Se han propuesto soluciones alternativas a las umbralizaciones blanda y dura. Por ejemplo, la umbralización firme (*Firm Shrinkage*, en inglés) [90] trata de mejorar los resultados obtenidos, en [72], usando ambas umbralizaciones. En [91, 92] se presenta una formulación variacional para los métodos IST, IHT y para el método similar utilizando umbralización firme.

Varios autores han comparado el rendimiento de usar umbralización suave o dura. Salvo en casos excepcionales, como alguno de los vistos en [84], la gran mayoría de estos autores han experimentado una superioridad en la práctica de la umbralización dura [84, 36, 93, 94, 95].

Se ha experimentado una gran mejora en el rendimiento general de este tipo de algoritmos cuando se utilizan umbrales dinámicos que decrecen en cada iteración. Por ejemplo, es una de las ideas inherentes a los métodos de punto próximo introducidos en [42]. Estos métodos resuelven iterativamente una sucesión de problemas del tipo (2.5) utilizando valores crecientes de λ. La adaptación dinámica de IST la encontramos propuesta para MCA [21, 22, 96] pero sin justificación teórica. También de forma heurística, [40] plantea la posibilidad de aumentar el radio de la bola ℓ_1 en la que proyecta cada operación de umbralización suave en cada iteración. En cuanto a versiones heurísticas del método IHT dinámico, las encontramos en [97, 15, 17]. En [98] se ha desarrollado un método basado en sustituir la norma ℓ_0 por una función equivalente continua. Utiliza una función gaussiana que lleva a un algoritmo diferente a ℓ_0-GM.

A pesar de su reciente introducción, los métodos basados en umbralización iterativa han demostrado ya ser muy potentes para numerosas aplicaciones. Por ejemplo, [22] utiliza IST y IHT para recuperación de píxeles perdidos en la imagen. También hay varios trabajos que atacan el

[4]Este trabajo es paralelo a la derivación del método y demostración de que el punto fijo es un mínimo local que nosotros planteamos en el Capítulo 4 y que fue publicado en [12].

problema clásico de restauración (emborronado más ruido) con IST (por ejemplo, [31, 99, 100]). Además, pueden encontrarse otras aplicaciones, como el tratamiento de imágenes médicas [101] o la codificación de video [102]. Por último, es de destacar la aplicación de este tipo de técnicas a Sensores Compresivos [103, 78].

2.4. Condiciones de equivalencia al minimizar las normas ℓ_1 y ℓ_0

En las secciones anteriores hemos repasado las técnicas más comunes para solucionar el problema de aproximación rala de la Ecuación (2.4). Hemos visto que la solución global es muy difícil de hallar en la práctica, y se han propuesto principalmente tres tipos distintos de aproximaciones al problema: métodos voraces, métodos basados en relajación convexa a través de técnicas clásicas de optimización y métodos basados en umbralización iterativa. La siguiente pregunta es: ¿Cómo de buenas son las soluciones que nos ofrecen estos métodos? En esta sección vamos a repasar unos sorprendentes resultados que demuestran que, bajo ciertas condiciones, tanto los métodos voraces (en concreto, OMP) como los métodos que resuelven de forma óptima el problema de relajación convexa alcanzan el óptimo global para el problema de *representación* rala, y obtienen un MSE proporcional al nivel de ruido al resolver el problema de *aproximación* rala.

Para estos resultados es vital el concepto de coherencia mutua de una matriz. Éste se define como $M(\mathbf{\Phi}) = \sup\{\langle \phi_i, \phi_j \rangle; \forall i \neq j\}$. Existe una restricción más fuerte, asociada a otra medida diferente de la riqueza del diccionario, denominada $Spark(\mathbf{\Phi})$ o rango de Kruskal. Se define como el número mínimo de columnas de la matriz que forman un conjunto linealmente dependiente. Se ha establecido la siguiente relación entre la $Spark(\mathbf{\Phi})$ y la coherencia mutua: $Spark(\mathbf{\Phi}) \geq \frac{1}{M(\mathbf{\Phi})}$.

El primer paso para establecer unas estudiar las condiciones bajo las cuales resolver el problema (2.5) ofrecía la solución global al problema (2.4) se dio al demostrar que, si una solución es suficientemente rala, es el único óptimo global del problema (2.3) de representación rala (ver [104, 105] y también [106, 107, 108, 109, 110]). Estos resultados eran interesantes porque permitían disponer de una manera sencilla de comprobar si la solución a la que se llegaba con los diferentes métodos aplicados era la óptima. La condición a comprobar se establece en $\|\hat{\mathbf{a}}^0\|_0 < \frac{Spark(\mathbf{\Phi})}{2}$. Pero aún faltaba tener resultados generales que estableciesen cuando un método llegaba efectivamente al óptimo global.

En [13, 46] se mostraba empíricamente (usando funciones discretas

pequeñas de una dimensión) que la solución al problema de relajación convexa es más rala que la solución de mínima norma euclídea. Definiendo la coherencia mutua, para dos matrices de igual tamaño, como $M(\mathbf{\Phi}_A, Phimtx_B) = \sup\{|\phi_a; \phi_b| : \phi_a \in \mathbf{\Phi}_A, \phi_b \in \mathbf{\Phi}_B\}$, en [105] se demuestra que, si la solución al problema de representación rala, cuando $\mathbf{\Phi}$ está formado por la concatenación de dos diccionarios mutuamente incoherentes (aquellos que dan un valor pequeño de $M(\mathbf{\Phi}_A, \mathbf{\Phi}_B)$), cumple que $\|\hat{\mathbf{a}}^0\|_0 < \frac{1}{2}(1 + \frac{1}{M(\mathbf{\Phi})})$, entonces es única y puede obtenerse mediante la minimización de la norma ℓ_1 de la representación. Posteriormente, [111] mejora esa cota situándola en $\|\hat{\mathbf{a}}^0\|_0 < \frac{0{,}9142}{M(\mathbf{\Phi})}$. Este resultado se puede extender también a diccionarios redundantes [112, 113, 114]. En estos trabajos se redujo la cota de unicidad de la solución para diccionarios generales a $\|\hat{\mathbf{a}}^0\|_0 < \frac{Spark(\mathbf{\Phi})}{2}$, lo que es una cota dos veces menos restrictiva que el límite establecido para que los métodos basados en relajación convexa alcancen ese mínimo global. Finalmente, [32] relajó las condiciones al demostrar que si una señal tiene una representación con menos de τN coeficientes con amplitud mayor que cero, donde $\tau > 0$ es un factor de proporcionalidad real, entonces la solución al problema de relajación convexa es única e igual a la solución del problema de representación rala. Sin embargo, no queda claro como hallar ρ para cada diccionario en concreto.

Pero en la mayoría de las situaciones prácticas no es razonable asumir que los coeficientes observados representen perfectamente a la señal. Por eso, es más interesante el escenario donde una señal ideal tiene una aproximación rala, pero sólo observamos una versión degradada por ruido blanco aditivo de la misma. En [34] se estudia el hecho de que los algoritmos basados en relajación convexa pueden generar buenas aproximaciones ralas en diccionarios redundantes en las mismas condiciones en las que obtienen la solución óptima para el problema de representación rala (descritas en [112, 113, 114]). Bajo estas cotas de raleza para la señal original, y si el diccionario tiene la propiedad de ser mutuamente incoherente, entonces los algoritmos basados en aproximación convexa son globalmente estables. Es decir, que el error cometido es proporcional al nivel de ruido existente incluso bajo el efecto de cantidades arbitrarias de ruido. También se demuestra que, bajo ciertas condiciones, el soporte de los resultados de estos métodos está contenido dentro de la selección ideal existente para la señal original sin ruido. Similares resultados fueron derivados también en [114, 74]. Referimos a [49] para tener una visión más completa de estos trabajos.

En cuanto a las técnicas voraces, en [33, 34] se demuestra que OMP encuentra la solución global en las mismas condiciones que BP para el problema de aproximación rala, con la diferencia de que OMP es localmente estable. Es decir, bajo cierta pequeña cantidad de ruido se puede recuperar

la representación rala ideal con un error que crece de forma como mucho proporcional al nivel de ruido. Sin embargo, en [115, 116] se muestra que, en la práctica, OMP consigue mejores resultados, y además más rápidamente. Referimos a [116] para ver más detalladamente estos resultados. Ño existen, de momento, resultados similares que demuestren bajo qué condiciones los métodos basados en umbralización iterativa alcanzan el óptimo global para los problemas de representación o aproximación rala.

Capítulo 3

Aproximación rala usando proyecciones alternas

En este capítulo se presenta un método de optimización sencillo y robusto que obtiene de forma no lineal una solución sub-óptima para el problema de aproximación rala de la Ecuación (2.4). La estrategia consiste en, dado un marco de Parseval que transforma los píxeles de la imagen a un dominio transformado sobre-completo, y valores para dos parámetros, p y R, buscar el vector de norma ℓ_p igual a R que mejor aproxime la imagen, en sentido del MSE de la reconstrucción. El método está basado en aplicar proyecciones ortogonales alternas[1] sobre el conjunto de vectores del dominio transformado de norma ℓ_p igual o menor a R, y sobre el conjunto de vectores que representan perfectamente a la imagen. Demostraremos que este método, al que llamamos ℓ_p-AP (por Proyecciones Alternas, *Alternated Projections* en inglés), converge al óptimo global de la función de coste cuando $p \geq 1$, y a un óptimo local si $0 \leq p < 1$. Aquí nos centraremos en los casos $p = 0$ y $p = 1$. Mostraremos que, incluso siendo sub-óptimo, ℓ_0-AP mejora claramente los resultados de ℓ_1-AP (que es equivalente a otros métodos de tipo *Basis Pursuit*). También veremos como reajustar los coeficientes de la solución dada por ℓ_1-AP, a través de una optimización por mínimos cuadrados de los coeficientes activos. Obtenemos así unos resultados de compactación ligeramente superiores a los de ℓ_0-AP. Por último, veremos que que ℓ_0-AP mejora los resultados de otras estrategias existentes, como las técnicas voraces o los métodos de umbralización iterativa basados en aplicar umbrales fijos. El método ℓ_0-AP fue ya usado, aunque derivado de forma heurística, en [17], mientras que ℓ_1-AP aparece en [40], desarrollado de forma paralela e independiente a

[1] Aquí utilizamos el término *proyección ortogonal* en un sentido amplio, para referirnos a cualquier proyección de mínima distancia euclídea.

nuestro trabajo.

En la Sección 3.1 describimos el método ℓ_p-AP, explicando con detalle los casos particulares $p = 0$ y $p = 1$. En la Sección 3.2 explicamos un método que, dado un conjunto de índices, encuentra el vector que tiene ese conjunto como soporte y aproxima con menor MSE a la imagen. Tras esto, se describen los detalles de implementación en la Sección 3.3, para discutir el resultado de los experimentos de compactación en la Sección 3.4. La Sección 3.5 concluye el capítulo.

3.1. El método ℓ_p-AP

Por motivos de claridad en la exposición, empezaremos recordando la formulación para el problema de aproximación que minimiza una norma general ℓ_p:

$$\hat{\mathbf{a}}^p(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_p^p + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}, \tag{3.1}$$

donde $\|\mathbf{a}\|_p = (\sum_{i=1}^{M} |a_i|^p)^{\frac{1}{p}}$ denota la norma ℓ_p de $\mathbf{a}$. Si asumimos que damos un valor determinado a λ, entonces $\hat{\mathbf{a}}^p(\lambda)$ tendrá una determinada norma ℓ_p, que notamos[2] $R(\lambda)$. Entonces, resolver la Ecuación (3.1) para un valor de λ dado es equivalente a minimizar el error de aproximación para una determinada norma de la solución, $\|\hat{\mathbf{a}}^p(\lambda)\|_p = R$:

$$\hat{\mathbf{a}}^p(\lambda) = \hat{\mathbf{a}}^p(R) = \arg\min_{\mathbf{a}\in\mathbb{R}^M} \|\mathbf{\Phi a} - \mathbf{x}\|_2^2 \text{ s.a. } \|\mathbf{a}\|_p^p = R. \tag{3.2}$$

Una bola ℓ_p de radio R, centrada en el origen, es el conjunto de todos aquellos vectores con norma ℓ_p menor o igual que R, $B_p(R) = \{\mathbf{a} \in \mathbb{R}^M : \|\mathbf{a}\|_p^p \leq R\}$. Resolvemos entonces aquí el problema:

$$\hat{\mathbf{a}}^p(R) = \arg\min_{\mathbf{a}\in B_p(R)} \|\mathbf{\Phi a} - \mathbf{x}\|_2. \tag{3.3}$$

Aunque, estrictamente, $\hat{\mathbf{a}}^p(R) \neq \hat{\mathbf{a}}^p(\lambda)$ (por incluir los vectores de norma menor que R como admisibles), en la práctica, como veremos mas adelante usaremos un método que nos asegura, bajo ciertas condiciones, que la solución cumple la restricción de la optimización de la Ecuación (3.2).

La Ecuación (3.3) puede resolverse a través de varias técnicas. Nosotros hemos elegido utilizar el método de las proyecciones alternas [37, 38] debido a su sencillez y a sus propiedades de convergencia. Este método consiste en proyectar ortogonalmente de forma alternada entre dos o más conjuntos

[2] A lo largo de esta Tesis, eliminamos la dependencia de R en λ por claridad.

hasta alcanzar la convergencia. Cuando los conjuntos son convexos y tienen intersección, el método converge hacia la proyección ortogonal del vector de partida sobre la intersección de los conjuntos involucrados. Cuando los conjuntos son convexos pero tienen intersección vacía entre ellos, el método converge a un ciclo límite de mínima distancia entre ellos. Cuando uno o más de los conjuntos no son convexos, el ciclo límite al que se converge es un mínimo local de la distancia entre ellos[3]. Véase [117] para una discusión más completa sobre las propiedades de convergencia cuando se usan conjuntos no convexos.

Para aplicar el método de las proyecciones alternas, tenemos que definir dos conjuntos. El primero de ellos será el conjunto de soluciones a la Ecuación (2.1), definido como $S(\mathbf{\Phi}, \mathbf{x}) = \{\mathbf{a} \in \mathbb{R}^M : \mathbf{\Phi a} = \mathbf{x}\}$. Es un sub-espacio afín del espacio $\mathbb{R}^M$, y por lo tanto es convexo. El segundo conjunto es la bola ℓ_p de radio R, centrada en el origen, $B_p(R)$, para unos valores de p y R dados. Este conjunto es convexo sólo si $p \geq 1$. Asumiremos que el vector inicial tiene una norma ℓ_p mayor que la deseada (como sucede en la práctica) asegurándonos así de que la solución a la que lleguemos estará sobre la frontera de la bola ℓ_p y que, como $B_p(R)$ es un conjunto cerrado, se cumple la restricción de la optimización de la Ecuación (3.2).

Denotamos $P_C^{\perp}(\mathbf{v})$ a la proyección ortogonal de un vector $\mathbf{v}$ sobre un conjunto C dado. La proyección ortogonal de $\mathbf{a}$ sobre el sub-espacio afín $S(\mathbf{\Phi}, \mathbf{x})$ de reconstrucción perfecta de $\mathbf{x}$ se puede hallar fácilmente, siendo:

$$P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\mathbf{a}) = \mathbf{a} + \mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi a}). \tag{3.4}$$

Este resultado se puede interpretar en términos de añadir al vector $\mathbf{a}$ la diferencia entre la solución de mínima norma euclídea, $\mathbf{a}^{LS} = \mathbf{\Phi}^T\mathbf{x}$, y el vector de análisis de la reconstrucción usando $\mathbf{a}$ ($\mathbf{\Phi}^T\mathbf{\Phi a}$).

Por otro lado, la expresión de la proyección ortogonal sobre $B_p(R)$, $P^{\perp}_{B_p(R)}(\mathbf{a})$, depende, obviamente, del valor concreto de p. Exploraremos en detalle los casos $p = 0$ y $p = 1$ en los siguientes apartados.

Finalmente, el método ℓ_p-AP se implementa a través de las siguientes iteraciones:

$$\begin{aligned} \hat{\mathbf{a}}^p(R)^{(0)} &= P^{\perp}_{B_p(R)}(\mathbf{a}^{LS}), \\ \hat{\mathbf{a}}^p(R)^{(k+1)} &= P^{\perp}_{B_p(R)}(P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\hat{\mathbf{a}}^p(R)^{(k)})). \end{aligned}$$

Hemos elegido parar las iteraciones cuando $\|\hat{\mathbf{a}}^p(R)^{(k+1)} - \hat{\mathbf{a}}^p(R)^{(k)}\|_2 < \delta$ para un $\delta > 0$ (ver los detalles de la implementación en la Sección 3.3). Veremos a continuación con detalle los casos $p = 0$ y $p = 1$.

[3]En este caso, puede ocurrir que la proyección ortogonal sobre uno de los conjuntos no convexos no sea única, pero este es un problema teórico sin consecuencias prácticas.

3.1.1. ℓ_0-AP

3.1.1.1. Proyección sobre la bola ℓ_0 de radio dado

Cuando $p = 0$, es directo derivar que $P^{\perp}_{B_0(R)}(\mathbf{a})$ es una operación que aplica un umbral duro conservando los R coeficientes más grandes en amplitud:

$$P^{\perp}_{B_0(R)}(\mathbf{a}) = \mathbf{a}^h,$$

donde:

$$a_i^h = \begin{cases} a_i, & |a_i| > \theta_h(\mathbf{a}, R) \\ 0, & |a_i| \leq \theta_h(\mathbf{a}, R). \end{cases}$$

Aquí, $\theta_h(\mathbf{a}, R)$ es el umbral más pequeño entre aquellos que preservan los $R - n_0$ mayores coeficientes en amplitud, siendo n_0 el entero no negativo más pequeño que garantiza que existe una solución. Así, $n_0 = 0$ si no hay amplitudes repetidas en el intervalo de interés. De acuerdo con la definición previa, en la práctica el umbral se establece a la amplitud del $R + 1$-ésimo coeficiente de mayor amplitud en el vector $\mathbf{a}$.

Este método también puede verse como un caso particular del método descrito en [17], pero con la diferencia de que, en aquel trabajo, el método propuesto no estaba justificado como un método formal de optimización.

3.1.1.2. Esquema y convergencia de ℓ_0-AP

El panel superior de la Figura 3.1 muestra una ilustración del método ℓ_0-AP de pequeñas dimensiones, lo que permite visualizar mejor su comportamiento ($N = 2$, $M = 3$, $R = 1$).

A continuación demostraremos que este método converge a un óptimo local, en el dominio de la imagen, del MSE de la reconstrucción para los vectores pertenecientes a la bola ℓ_0. Sustituyendo según la Ecuación (3.4):

$$\|\mathbf{a} - P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\mathbf{a})\|_2 = \|\mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{a})\|_2 = \|\mathbf{x} - \mathbf{\Phi}\mathbf{a}\|_2, \tag{3.5}$$

donde el último paso es cierto porque $\mathbf{\Phi}^T$ es un marco de Parseval. Dado que $\hat{\mathbf{a}}^0(R)$ es un mínimo local en $B_0(R)$ de la distancia a $S(\mathbf{\Phi}, \mathbf{x})$, entonces existe un $\delta > 0$ tal que para todo $\mathbf{a} \in B_0(R)$, si $\|\mathbf{a} - \hat{\mathbf{a}}^0(R)\|_2 < \delta$ entonces $\|\mathbf{a} - P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\mathbf{a})\|_2 \geq \|\hat{\mathbf{a}}^0(R) - P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\hat{\mathbf{a}}^0(R))\|_2$. Usando (3.5) tenemos que $\|\mathbf{x} - \mathbf{\Phi}\mathbf{a}\|_2 \geq \|\mathbf{x} - \mathbf{\Phi}\hat{\mathbf{a}}^0(R)\|_2$. Esto es, que $\hat{\mathbf{a}}^0(R)$ es un mínimo local en $\mathbf{a}$, y dentro del conjunto $B_0(R)$, de la distancia euclídea entre $\mathbf{\Phi}\mathbf{a}$ y $\mathbf{x}$.

En cuanto a las propiedades de convergencia, hemos observado que el método evoluciona rápidamente hacia la solución durante las primeras

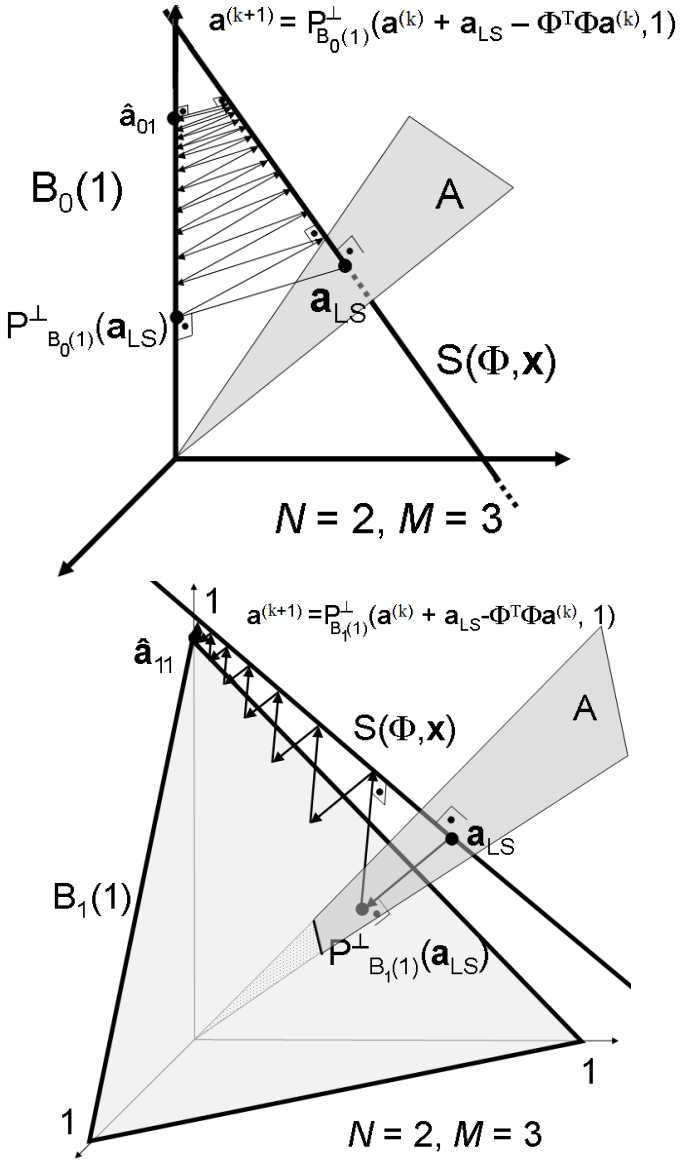

Figura 3.1: **Arriba**, *explicación gráfica del método ℓ_0-AP.* **Abajo**, *lo mismo para ℓ_1-AP. Sólo se muestra una cara de la bola ℓ_1 para aumentar la claridad.*

iteraciones, y luego se va reduciendo la velocidad de convergencia, tal y como se muestra en la Figura 3.2 para las imágenes *Barbara*, *Boat* y *House* de nuestro conjunto de prueba (ver Apéndice A). Vemos que la velocidad de convergencia del método también depende del grado de raleza impuesto (cuanto más raleza, más rapidez de convergencia). En particular, en esta Tesis estamos interesados en explorar el comportamiento de los métodos en la convergencia, y esto ha requerido realizar algunos miles de iteraciones

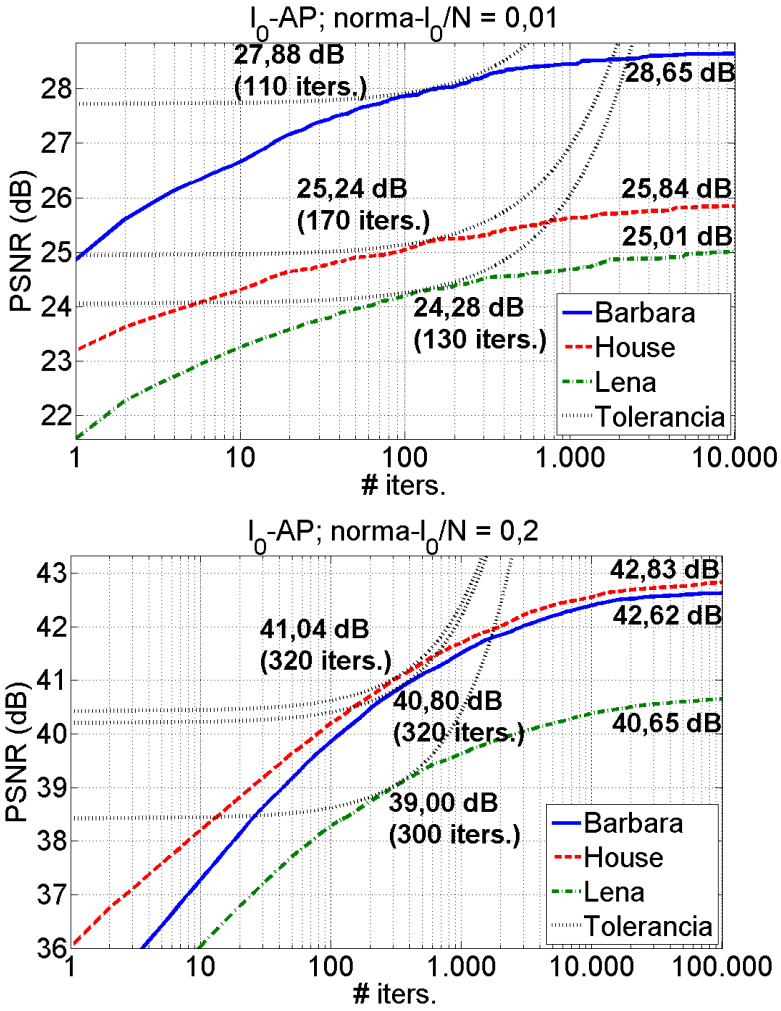

Figura 3.2: *Gráfica en ejes logarítmicos de la calidad en la aproximación (PSNR, en dB) en función del número de iteraciones para ℓ_0-AP usando tres imágenes y dos niveles de raleza. La representación usada es DT-CWT. El número al final de las curvas es la PSNR en la convergencia. Los números que acompañan al punto de tangencia (indicado por las curvas punteadas) son la PSNR y el número de iteraciones al cumplir el criterio de parada.*

para cada experimento. Sin embargo, en una implementación práctica, se pueden realizar muchas menos iteraciones con resultados satisfactorios. En este trabajo hemos establecido el criterio de parada basándonos en el incremento de la PSNR en cada tramo de 10 iteraciones. La curva punteada es la correspondiente a la tasa de crecimiento usada como tolerancia (sería una línea recta si no estuviese en coordenadas logarítmica). En el apartado 3.3.2 se pueden ver más detalles sobre el criterio de parada.

3.1.2. ℓ_1-AP

3.1.2.1. Proyección sobre la bola ℓ_1 de radio dado

En el caso $p = 1$, se puede demostrar que la proyección ortogonal de un vector $\mathbf{a}$ sobre la bola ℓ_1 de radio R dado, que notamos como $\mathbf{a}^s = P^{\perp}_{B_1(R)}(\mathbf{a})$, es una operación de umbralización suave. Esto ya ha sido demostrado anteriormente, por ejemplo en [39, 40], pero nuestra demostración alternativa aportará además un método iterativo para hallar el valor de umbral asociado a esta operación.

Asumimos primero que $\|\mathbf{a}\|_1 > R$, porque si no la proyección sobre $B_1(R)$ sería la identidad. Además, haremos uso de la propiedad de conservación del signo entre los coeficientes de cualquier vector y su proyección sobre una bola ℓ_p, de forma que tenemos que $sign(\mathbf{a}^s) = sign(\mathbf{a})$. El problema se reduce entonces a proyectar el vector formado por los componentes $\{|a_1|, |a_2|, ..., |a_M|\}$, que notamos $\mathbf{a}^{abs}$, sobre el hiper-cuadrante positivo de la bola $B_1(R)$. Una vez obtenida esa proyección, restauraremos el signo de cada elemento para obtener la proyección de $\mathbf{a}$ sobre $B_1(R)$.

El hiper-cuadrante positivo de $B_1(R)$ se puede definir como la intersección de dos conjuntos convexos. El primero de ellos será el conjunto de todos aquellos vectores cuyos componentes sumen R:

$$F(R) = \{\mathbf{b} \in \mathbb{R}^M : \sum_{i=1}^{M} b_i \leq R\}.$$

Dado un vector $\mathbf{c} \in \mathbb{R}^M$, la expresión de la proyección ortogonal sobre este conjunto es:

$$P^{\perp}_{F(R)}(\mathbf{c}) = \mathbf{c} - \delta,$$

donde $\delta = \frac{\sum_{i=1}^{M}(c_i)-R}{M}$ si $\sum_{i=1}^{M} c_i > R$ y 0 en caso contrario.

El segundo conjunto es el hiper-cuadrante positivo del espacio vectorial M-dimensional completo:

$$G^+ = \{\mathbf{b} \in \mathbb{R}^M : \forall\ i = \{1, ..., M\}, b_i \geq 0\}.$$

La proyección ortogonal sobre este conjunto se define como:

$$P^{\perp}_{G^+}(\mathbf{c}) = \mathbf{D}\mathbf{c},$$

donde $\mathbf{D}$ es una matriz diagonal, de tamaño $M \times M$, tal que $d_{ii} = 1$ si $c_i > 0$ y 0 en caso contrario.

Siguiendo la teoría de proyecciones alternas, la proyección ortogonal del vector $\mathbf{a}$ sobre la intersección de $F(R)$ y G^+, que notamos como $\mathbf{a}^{pro} = P^{\perp}_{F(R)\bigcap G^+}(\mathbf{a})$, se define como:

$$\mathbf{a}^{pro} = \lim_{n\to\infty}[P^{\perp}_{G^+}(P^{\perp}_{F(R)}(\cdots n \cdots P^{\perp}_{G^+}(P^{\perp}_{F(R)}(\mathbf{a}^{abs}))\cdots n \cdots))], \tag{3.6}$$

La proyección ortogonal de $\mathbf{a}$ sobre $B_1(R)$ se obtiene finalmente como:

$$\mathbf{a}^s = sign(\mathbf{a}) \cdot \mathbf{a}^{pro}. \tag{3.7}$$

A continuación demostraremos que la expresión obtenida en la Ecuación (3.7) es una umbralización suave. Primero, para $k = \{1, ..., n\}$, notamos como $\delta^{(k)}$ al sustraendo correspondiente a la k-ésima aplicación de la proyección ortogonal sobre $F(R)$ en la Ecuación (3.6). Además, notamos $\mathbf{D}^{(k)}$ a la máscara aplicada en la k-ésima aplicación de la proyección ortogonal sobre G^+. Entonces:

$$\mathbf{a}^{pro} = \lim_{n\to\infty}[\mathbf{D}^{(n)}(\cdots\mathbf{D}^{(2)}(\mathbf{D}^{(1)}(\mathbf{a}^{abs} - \delta^{(1)}) - \delta^{(2)})\cdots - \delta^{(n)})].$$

Esto puede expresarse como:

$$\mathbf{a}^{pro} = \mathbf{a}^{abs} - \mathbf{d}, \tag{3.8}$$

donde cada elemento d_i se define como:

$$d_i = \begin{cases} \theta_s(\mathbf{a}, R), & |a_i| > \theta_s(\mathbf{a}, R) \\ |a_i|, & |a_i| \leq \theta_s(\mathbf{a}, R), \end{cases}$$

y donde $\theta_s(\mathbf{a}, R) = \sum_{k=1}^{n} \delta^k$. En consecuencia, si sustituimos la expresión de la Ecuación (3.8) en la Ecuación (3.7) obtenemos:

$$\mathbf{a}^s = sign(\mathbf{a}) \cdot (\mathbf{a}^{abs} - \mathbf{d}),$$

que es la definición de una operación de umbralización suave. Es decir, $\mathbf{a}^s = S_1(\mathbf{a}, \theta_s(\mathbf{a}, R))$ (ver Ecuación (2.8)).

Esta demostración nos ha proporcionado, además, un método basado en proyecciones alternas para, dado $\mathbf{a}$, encontrar el valor del umbral que nos lleva al valor deseado de la norma ℓ_1 tras la umbralización. Este método comenzaría por eliminar el signo de $\mathbf{a}$, proyectar el resultado sobre la intersección de $F(R)$ y G^+ utilizando proyecciones alternas, y finalmente devolver el signo original a cada elemento de esa proyección. Hemos experimentado que este método converge linealmente en pocas iteraciones. A continuación, desarrollamos un método cuyas iteraciones requieren menos cálculo, de forma que el método final es más sencillo de implementar.

Primeramente, vamos a expresar R en función del umbral[4] $\theta_s(\mathbf{a}, R)$. Para ello, definimos el conjunto de índices que corresponden a coeficientes de $\mathbf{a}$ con amplitudes superiores a un umbral θ: $\Upsilon(\mathbf{a}, \theta) = \{i \in \{1, ..., M\} : |a_i| > \theta\}$. Entonces, podemos escribir:

$$\begin{aligned} R &= \sum_{\Upsilon(\mathbf{a},\theta_s)} (|a_i| - \theta_s) \\ R &= \left(\sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i| \right) - card(\Upsilon(\mathbf{a}, \theta_s)) \cdot \theta_s, \end{aligned}$$

donde $card(\cdot)$ indica la cardinalidad de un conjunto. Podemos expresar la ecuación anterior como:

$$\theta_s = \frac{\left(\sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i|\right) - R}{card(\Upsilon(\mathbf{a}, \theta_s))}. \tag{3.9}$$

El término de la derecha depende de θ_s, pero podemos resolver esta ecuación iterativamente usando las siguientes iteraciones:

$$\begin{aligned} \theta_s^{(0)} &= 0, \\ \theta_s^{(k+1)} &= \frac{\left(\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i|\right) - R}{card(\Upsilon(\mathbf{a}, \theta_s^{(k)}))}. \end{aligned} \tag{3.10}$$

Las iteraciones terminan cuando $\|\theta_s^{(k+1)} - \theta_s^{(k)}\|_2$ está por debajo de un umbral de tolerancia (ver apartado 3.3.2 para más detalles sobre el criterio de parada).

A continuación, demostramos que las iteraciones (3.10) convergen a θ_s, haciendo notar primero que $R(\theta_s)$ es una función estrictamente decreciente, y, en consecuencia, también lo es $\theta_s(R)$. Esto implica que la Ecuación (3.9) tiene una solución única en θ_s. Si encontramos $\theta_s^{(k+1)} = \theta_s^{(k)}$ entonces ese valor satisface la Ecuación (3.9), por lo que sabemos que si las iteraciones convergen entonces lo hacen a la única solución θ_s. Entonces, para probar la convergencia a θ_s, es suficiente demostrar que la sucesión $\theta_s^{(k)}$ converge. Esto puede hacerse demostrando que, 1) $\theta_s^{(k)}$ es monótonamente creciente, y 2) está acotada superiormente por θ_s. Esto es lo que hacemos a continuación.

Comenzamos observando que $\theta_s^{(0)} = 0 \leq \theta_s$. Asumiendo que $\theta_s^{(k)} \leq \theta_s$, entonces:

$$\sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} |a_i| \leq \sum_{\Gamma(\mathbf{a},\theta_S^{(k)},\theta_s)} \theta_s,$$

[4]Por claridad en la notación, en la derivación que sigue hemos eliminado la dependencia de θ_s sobre $\mathbf{a}$ y R.

donde $\Gamma(\mathbf{a}, \theta_1, \theta_2) = \{i \in \{1, ..., M\} : \theta_1 < |a_i| \leq \theta_2\}$. De aquí obtenemos lo siguiente:

$$\begin{aligned}
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - \sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i| &\leq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s - \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s, \\
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\leq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s, \\
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\leq card(\Upsilon(\mathbf{a}, \theta_s^{(k)})) \cdot \theta_s, \\
\frac{\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R}{card(\Upsilon(\mathbf{a}, \theta_s^{(k)}))} &\leq \theta_s, \\
\theta_s^{(k+1)} &\leq \theta_s.
\end{aligned}$$

A continuación vemos que, como $\sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} |a_i| \geq \sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} \theta_s^{(k)}$, y como también $\sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s \geq \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s^{(k)}$, entonces:

$$\sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} |a_i| + \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s \geq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s^{(k)},$$

y son ciertas las siguientes desigualdades:

$$\begin{aligned}
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - \sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i| + \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s &\geq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s^{(k)}, \\
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\geq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s^{(k)}, \\
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\geq card(\Upsilon(\mathbf{a}, \theta_s^{(k)})) \cdot \theta_s^{(k)}, \\
\frac{\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R}{card(\Upsilon(\mathbf{a}, \theta_s^{(k)}))} &\geq \theta_s^{(k)}, \\
\theta_s^{(k+1)} &\geq \theta_s^{(k)}.
\end{aligned}$$

En consecuencia, la sucesión es monótonamente creciente, con lo que la prueba está completa.

3.1.2.2. Esquema y convergencia de ℓ_1-AP

La Figura 3.1 (panel inferior) ilustra el comportamiento de ℓ_1-AP con $N = 2$, $M = 3$, y $R = 1$. Sólo se muestra una cara de $B_1(1)$ para mejorar la visualización de la figura.

Es fácil demostrar que ℓ_1-AP proporciona el mínimo global para la distancia euclídea, en el dominio de la imagen, entre las reconstrucción desde los vectores de $B_1(R)$ y la imagen $\mathbf{x}$. Primero notamos que $\hat{\mathbf{a}}^1(R)$ es el mínimo global en $B_1(R)$ de la distancia euclídea a $S(\mathbf{\Phi}, \mathbf{x})$ (porque los dos conjuntos son convexos). Entonces, para todo $\mathbf{a} \in B_1(R)$, tenemos que $\|\mathbf{a} - P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\mathbf{a})\|_2 \geq \|\hat{\mathbf{a}}^1(R) - P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\hat{\mathbf{a}}^1(R))\|_2$. Aplicando la Ecuación (3.4) y siendo $\mathbf{\Phi}^T$ un marco de Parseval, obtenemos que $\|\mathbf{x} - \mathbf{\Phi a}\|_2 \geq \|\mathbf{x} - \mathbf{\Phi}\hat{\mathbf{a}}^1(R)\|_2$. Esto es, que $\mathbf{\Phi}\hat{\mathbf{a}}^1(R)$ es el mínimo global, para todo $\mathbf{a} \in B_1(R)$, de la distancia euclídea a de $\mathbf{\Phi a}$ a $\mathbf{x}$.

La Figura 3.3 ilustra las propiedades de convergencia de ℓ_1-AP. La interpretación es similar a la de la Figura 3.2. Como no hay soluciones locales que evitar, la convergencia es más regular que con ℓ_0-AP y el número de iteraciones necesitado para converger es menor. Hemos incluido un ejemplo, en el panel inferior, donde se consigue reconstrucción perfecta.

3.2. Minimización del error cuadrático para una selección de coeficientes dada

A medida que avanzan las iteraciones del método ℓ_0-AP, la selección de coeficientes que se realiza se vuelve cada vez más y más estable, de modo que la solución final se convierte en óptima en sentido de mínimos cuadrados, para esa selección[5]. Como ya fue indicado en [17], en el límite, cuando se fija el número de coeficientes activos, los dos conjuntos (el sub-espacio vectorial generado por los átomos seleccionados y el sub-espacio afín de reconstrucción perfecta) involucrados son convexos, y las iteraciones convergen al óptimo global, para esa selección, de forma lineal.

Sin embargo, no ocurre lo mismo si utilizamos una norma ℓ_p genérica, porque la proyección sobre la bola ℓ_p no es, en general, aplicar un umbral duro. Como nuestro objetivo final es resolver el problema de aproximación rala de la Ecuación (2.4), debemos usar algún método para mejorar la calidad de la aproximación para un conjunto dado de funciones seleccionadas de alguna forma. Aquí usamos uno basado en proyecciones alternas que ya ha sido previamente usado por numerosos autores, como [13, 15, 26, 34, 100].

Dado un conjunto I de R índices extraídos de $\{1, ..., M\}$, definimos $\mathbf{\Phi}_I$ como una matriz $N \times R$ formada por las columnas ϕ_i de $\mathbf{\Phi}$ tales que $i \in I$. Entonces, queremos encontrar:

$$\hat{\mathbf{a}}_I = \arg \min_{\mathbf{a}_I \in \mathbb{R}^R} \|\mathbf{\Phi}_I \mathbf{a}_I - \mathbf{x}\|_2,$$

[5]Notar que el método todavía es sub-óptimo porque la selección de funciones elementales no es óptima en general.

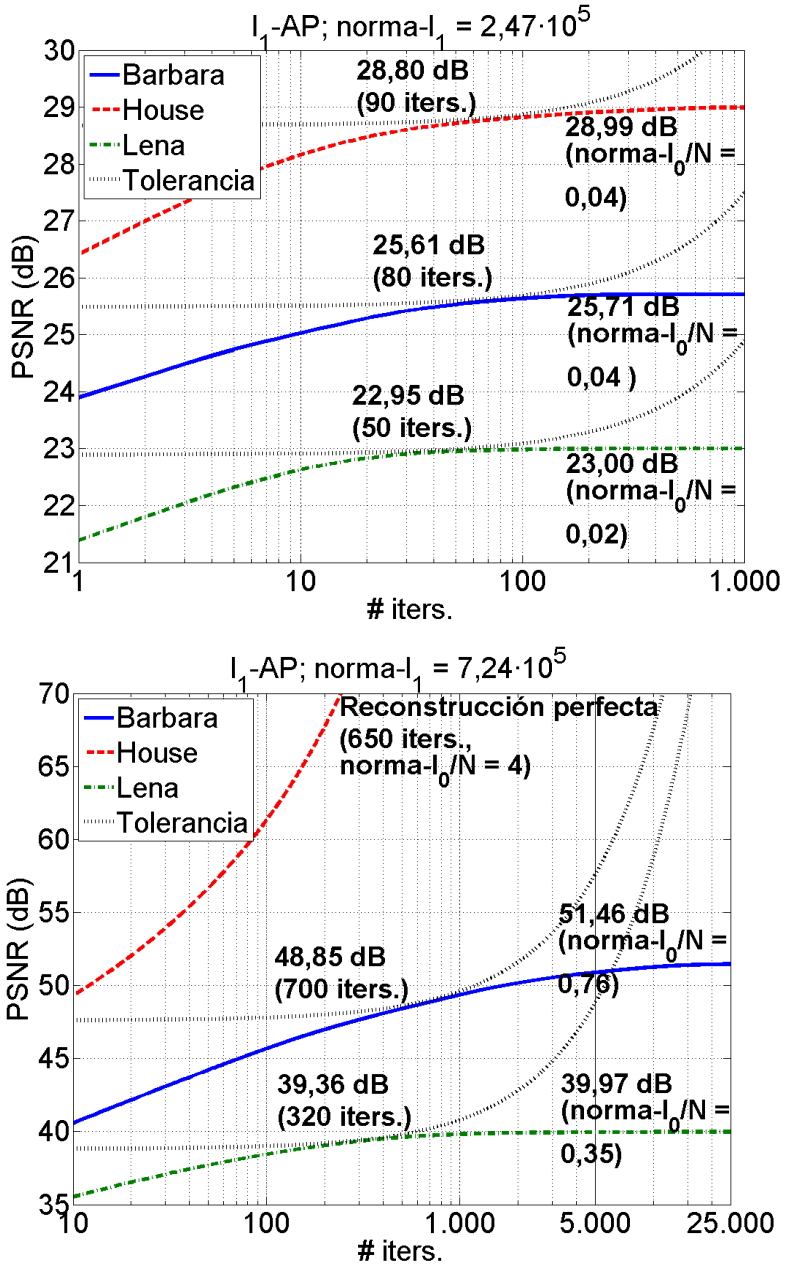

Figura 3.3: *Curvas de convergencia en escala semi-logarítmica para el método ℓ_1-AP usando tres imágenes y dos niveles de raleza. Los detalles son similares a los de la Figura 3.2. También está indicada la norma ℓ_0, normalizada por N, de la solución en la convergencia.*

que se traduce en $\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^\sharp \mathbf{x}$, donde $\mathbf{\Phi}_I^\sharp$ es la pseudo-inversa de $\mathbf{\Phi}_I$. Nótese que $\mathbf{\Phi}_I^\sharp = \mathbf{\Phi}_I^T[\mathbf{\Phi}_I\mathbf{\Phi}_I^T]^{-1}$ si $R > N$ y $\mathbf{\Phi}_I^\sharp = [\mathbf{\Phi}_I^T\mathbf{\Phi}_I]^{-1}\mathbf{\Phi}_I^T$ si $R \leq N$. Cuando tratamos con imágenes, el tamaño de $\mathbf{\Phi}_I$ convierte el cálculo de la pseudo-

inversa de forma directa en una tarea completamente impracticable. Lo que se hace en vez de eso es seguir el siguiente esquema iterativo:

$$\begin{aligned} \mathbf{a}^{(0)} &= \mathbf{D}_I \mathbf{\Phi}^T \mathbf{x}, \\ \mathbf{a}^{(k+1)} &= \mathbf{D}_I [\mathbf{a}^{(k)} + \mathbf{\Phi}^T (\mathbf{x} - \mathbf{\Phi} \mathbf{a}^{(k)})]. \end{aligned} \tag{3.11}$$

donde $\mathbf{D}_I$ es una matriz diagonal de tamaño $M \times M$ tal que $d_{ii} = 1$ si $i \in I$ y 0 en caso contrario. En el Apéndice C demostramos que este método efectivamente resuelve la pseudo-inversa en $\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^\sharp \mathbf{x}$.

3.3. Implementación

3.3.1. Representaciones

Para probar los métodos, inicialmente utilizamos cuatro marcos de Parseval diferentes, a saber: DT-CWT [45], Curvelets [118], Pirámide Orientable [7] y una versión redundante de las ondículas de Haar [119]. De entre ellas elegimos las dos que dan el mejor rendimiento medio en términos de compactación de energía. Estos son DT-CWT y Curvelets[6]. El factor de redundancia de DT-CWT es 4, y de Curvelets es $\approx 7{,}2$.

Con el objetivo de realizar un tratamiento homogéneo de ambas representaciones, hemos dividido los coeficientes DT-CWT en su parte real e imaginaria. Por otro lado, para optimizar la aproximación en tasas de raleza extremadamente altas, hemos insertado, en ambas representaciones, una escala extra compuesta de la media global de la imagen. De esta forma nos adaptamos al hecho de que normalmente la mejor aproximación a una imagen natural, utilizando sólo un coeficiente, es quedarse con la media global.

El código MATLAB® para DT-CWT está disponible en [120]. También hemos usado código MATLAB® para la implementación de Curvelets (*CurveLab 2.0* [121]).

3.3.2. Convergencia y criterio de parada

El criterio de convergencia para el método ℓ_p-AP se traduce en nuestra implementación en el uso de dos constantes de tolerancia. La primera controla el incremento en PSNR de la estimación para decidir si se ha alcanzado o no la convergencia. De forma empírica, hemos elegido parar cuando, tras 10 iteraciones, este incremento es menor que 0,02 dB. Este

[6] Hemos llevado a cabo los experimentos con todas las representaciones, y los resultados con todas ellas son cualitativamente similares a los presentados.

criterio de parada se puede ver dibujado como curvas punteadas en las Figuras 3.2 y 3.3. Estas curvas serían rectas tangentes a las curvas de convergencias de los métodos si el eje de abscisas se mostrase en escala lineal. Hemos experimentado que este criterio da, típicamente, diferencias con respecto a la PSNR en la convergencia menores de 1 dB en el rango de alta raleza y menores que 2 dB en el rango de baja raleza. Estas diferencias son incluso más pequeñas para ℓ_1-AP (favoreciendo, de esta forma, a este último método en la comparación).

Nótese que, si el radio de la bola ℓ_p usada es grande, el método consigue reconstrucción perfecta de la imagen. En este caso, el incremento en PSNR es, en concordancia con la teoría, lineal. Para detectar esta situación hemos usado un segundo criterio de tolerancia, que controla el incremento de PSNR tras cada 10 iteraciones. Las iteraciones se detienen cuando la diferencia entre los dos últimos incrementos es menor que una constante (10^{-6} para ℓ_0-AP y 10^{-4} para ℓ_1-AP).

La búsqueda del umbral en cada iteración de ℓ_0-AP la realizamos a través de una búsqueda por sección áurea. Esto requiere un parámetro extra de tolerancia que controla el tamaño del intervalo de búsqueda. Hemos utilizado el valor 0,1 para este parámetro. Para ℓ_1-AP, hemos utilizado el método descrito en el apartado 3.1.2, utilizando como criterio de parada la diferencia entre el radio deseado para la bola ℓ_1 y el obtenido en cada iteración. Hemos experimentado que, eventualmente, el método iterativo utilizado para buscar el umbral en ℓ_1-AP proporciona exactamente el radio de la bola ℓ_1 requerido. Sin embargo, para reducir computación en la práctica, hemos elegido 0,1 también como valor de esta tolerancia.

3.4. Resultados y discusión

En los siguientes experimentos pretendemos comparar la capacidad de nuestros métodos para compactar la energía en pocos coeficientes, comparando con otros algoritmos referentes en este área. Se ha explorado un un amplio rango de raleza para cada método, y se han usado las imágenes de nuestro conjunto de prueba.

Hemos usado una escala logarítmica para el eje de ordenadas en nuestras figuras, a pesar de que la PSNR es ya una medida logarítmica. Creemos que esto, aunque no sea usual, está justificado en este caso porque mejora enormemente la visualización de los resultados. En cuanto al muestreo de las curvas, cada marcador corresponde a una medida promediada del método correspondiente en nuestro conjunto de prueba, y hemos interpolado linealmente los valores intermedios.

3.4.1. Comparación de algunos métodos previos

Nuestros primeros experimentos comparan estrategias de aproximación rala comúnmente usadas. Nos hemos planteado los dos siguientes objetivos: a) comparar el uso de umbrales duros y suaves en algoritmos de umbralización iterativos; y b) comparar las estrategias directa y acumulativa para la selección de funciones de la representación en algoritmos voraces.

Con respecto al primer objetivo, hemos implementado el método de umbralización iterativa en sus versiones IHT e IST, como fueron descritos en el apartado 2.3.3. Recordemos que estos métodos iteran entre la umbralización y la proyección sobre el espacio afín de reconstrucción perfecta (Ecuación (3.4)), utilizando un umbral fijo. Hemos usado el mismo criterio de parada que con ℓ_p-AP (ver apartado 3.3.2). De esta forma, nuestra implementación de estos métodos sólo se diferencia de nuestra implementación de ℓ_0-AP y ℓ_1-AP en que aquellos usan un umbral fijo y éstos un radio de la bola ℓ_p fijo.

Para comparar los métodos heurísticos voraces entre sí, hemos implementado StOMP [25] y el método que presentamos en [26], y al que llamamos aquí DT+OP (de Umbralización Directa más Optimización, *Direct Thresholding plus Least Squares-Optimization* en inglés). Para elegir el umbral utilizado por StOMP, hemos establecido previamente cuántos coeficientes serán elegidos en cada iteración del algoritmo. Por su parte, DT+OP aplica el umbral directamente (y sólo una vez) sobre la representación lineal de la imagen para cada muestra. Ambos métodos usan las Ecuaciones (3.11) para optimizar la calidad de la reconstrucción obtenida tras cada umbralización. Aquí también hemos utilizado los mismos criterios de parada descritos para ℓ_p-AP.

La Figura 3.4 muestra gráficamente los resultados de este experimento. El panel de arriba muestra los resultados obtenidos con DT-CWT, usando 8 escalas, y el de abajo con Curvelets usando 6. Esta figura muestra que el rendimiento de la umbralización dura iterativa es mejor que el de la umbralización suave iterativa para niveles medio-altos de raleza. En niveles bajos, el número de mínimos locales aumenta considerablemente y entonces IHT tiene más probabilidades de quedar atrapado en mínimos locales poco favorables. También se puede ver que los resultados obtenidos con DT+OP mejoran a nuestra implementación de StOMP, excepto en el rango de muy alta raleza, que tiene poca importancia práctica. Esto indica que la selección directa de coeficientes parece más adecuada que la acumulación de ellos en base a la correlación con el residuo. Entre los métodos comparados, IHT tiene el mejor rendimiento. Anteriormente, varios autores han apuntado que la umbralización dura favorece la compactación de energía frente a la suave [84, 93, 94, 18], pero no se han presentado comparaciones sistemáticas

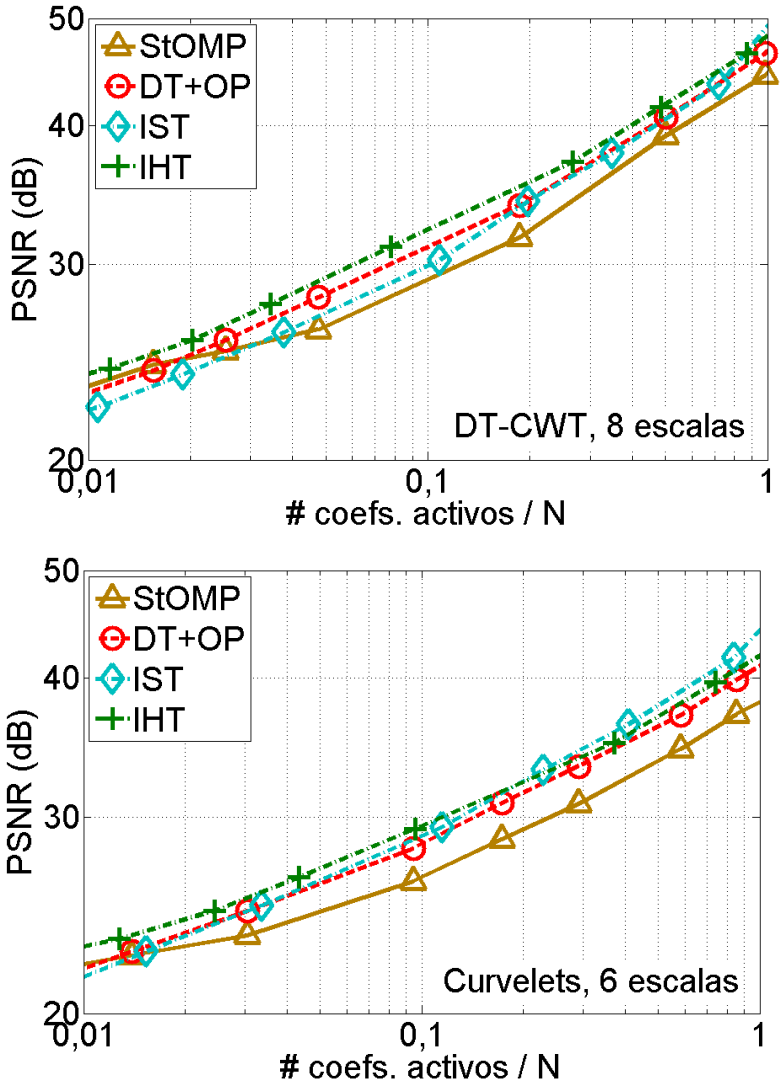

Figura 3.4: *Resultados de compactación promediados en el conjunto de prueba para StOMP, DT+OP, IHT e IST.* **Arriba**, *usando DT-CWT con* 8 *escalas.* **Abajo**, *usando Curvelets con* 6 *escalas.*

utilizando imágenes naturales.

3.4.2. Comparación de ℓ_0-AP con ℓ_1-AP y métodos previos

Nuestro segundo experimento compara ℓ_0-AP con ℓ_1-AP. Debido a que las amplitudes de los coeficientes del resultado de ℓ_1-AP no son óptimas en cuanto a calidad de reconstrucción para el soporte obtenido, también

comparamos con el resultado de optimizar por mínimos cuadrados estos coeficientes, usando las iteraciones de la Ecuación (3.11). Etiquetamos a este método como ℓ_1-AP+OP. Además, hemos incluido IHT y StOMP como representantes de las estrategias de umbralización iterativa con umbral fijo y los métodos voraces, respectivamente.

La Figura 3.5 muestra el resultado de este experimento. El panel de arriba muestra los resultados de DT-CWT, con 8 escalas, y el de abajo los de Curvelets con 6. Podemos ver como ℓ_0-AP mejora claramente a ℓ_1-AP, incluso aunque este último esté minimizando óptimamente la norma ℓ_1 para cada nivel de raleza. También vemos que ℓ_1-AP+OP mejora drásticamente los resultados de ℓ_1-AP, incluso superando ligeramente a ℓ_0-AP. Esto muestra que la selección de coeficientes hecha por ℓ_1-AP es mejor, en general, que la de ℓ_0-AP, especialmente en el rango de baja raleza. Esto, al igual que antes, parece una consecuencia natural de que ℓ_0-AP se queda atrapado en óptimos locales poco favorables, cuyo número crece rápidamente al decrecer el nivel de raleza.

También se puede ver que ℓ_0-AP mejora significativamente los resultados de IHT y StOMP. Es interesante ver que fijar el radio de la bola ℓ_p en cada iteración resulta ser mucho mejor que fijar el umbral en cada iteración. En el caso de los métodos de relajación convexa, tanto IST como ℓ_1-AP encuentran, eventualmente, el óptimo global, pero estos resultados demuestran que

Las tablas 3.1 y 3.2 muestran los resultados numéricos de la curvas de la Figura 3.5.

La Figura 3.6 compara de forma visual los métodos reconstruyendo la imagen *Einstein*[7] usando $0{,}0765 \cdot N$ coeficientes Curvelets. Desde arriba hacia abajo, la columna de la izquierda muestra la imagen original, el resultado de ℓ_1-AP (30,85 dB) y el de ℓ_1-AP+OP (33,52 dB). Véase la gran mejora visual obtenida al post-optimizar los coeficientes seleccionados. La columna de la derecha muestra StOMP (28,66 dB), IHT (29,10 dB) y ℓ_0-AP (32,98 dB). Aunque está más de medio dB por debajo en PSNR, no hay una diferencia significativa visual entre el resultado de ℓ_0-AP y el de ℓ_1-AP+OP. Además, aunque no se muestra aquí, esta diferencia se vuelve más pequeña para valores menores de PSNR.

Como hemos apuntado antes, el método ℓ_0-AP es equivalente al presentado en [17] cuando se usa un número fijo de coeficientes en cada iteración y no se usan heurísticos añadidos. Los autores referidos aplican el umbral en cada iteración sobre la magnitud de cada coeficiente

[7]Para todos los experimentos de esta Tesis, hemos eliminado el reborde negro de *Einstein* replicando las filas y columnas adyacentes. Esto la convierte en una imagen natural más representativa.

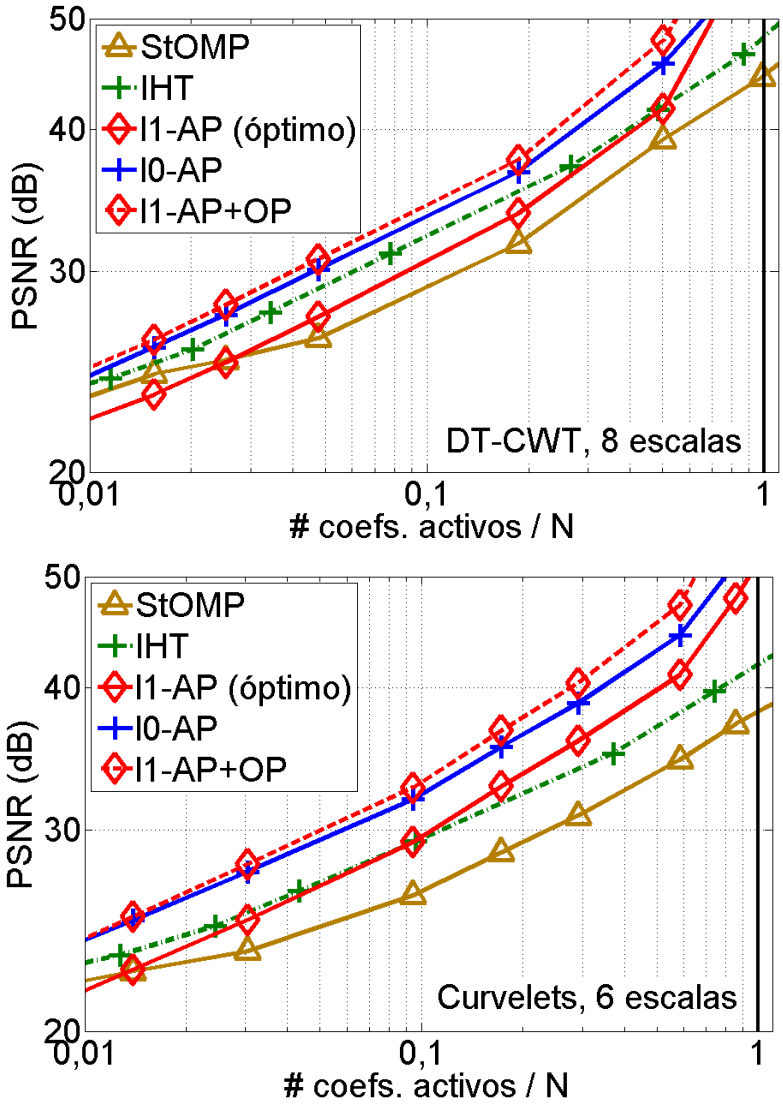

Figura 3.5: *Resultados de compactación, promediados en el conjunto de prueba, de los métodos ℓ_0-AP, ℓ_1-AP, ℓ_1-AP+OP, IHT y DT+OP.* **Arriba**, *usando DT-CWT con 8 escalas.* **Abajo**, *usando Curvelets con 6 escalas.*

complejo. Entonces, para comparar con propiedad sus resultados con los nuestros, hemos doblado el número de coeficientes seleccionados dado en sus resultados. Comparamos usando 5 escalas DT-CWT como hacen estos autores. Usando un número fijo de 24000 coeficientes seleccionados con la imagen *Lena*[8] de tamaño 512×512 nuestro resultado es superior al suyo en 2,02 dB (39,09 contra 37,07 dB).

[8]Agradecemos al Prof. Nick Kingsbury por la ayuda prestada para replicar este experimento.

DT-CWT/Curvelets		R/N		
Imagen	Método	0.00305	0.00944	0.02914
Barbara	StOMP	25,75/24,02	27,91/27,26	35,99/32,56
	IHT	28,18/25,98	33,25/30,51	40,58/34,30
	ℓ_1-AP	26,39/25,75	30,21/31,02	38,65/38,38
	ℓ_0-AP	*29,23/28,61*	*33,38/34,29*	*41,76/41,51*
	ℓ_1-AP+OP	**29,95/29,10**	**34,12/34,94**	**43,09/43,08**
House	StOMP	28,64/25,52	30,81/29,23	37,88/34,22
	IHT	31,19/28,64	34,76/32,69	40,25/37,77
	ℓ_1-AP	29,60/27,98	32,79/33,32	39,56/39,35
	ℓ_0-AP	*32,09/30,45*	*35,18/35,63*	*43,00/41,87*
	ℓ_1-AP+OP	**32,61/31,19**	**35,79/36,69**	**44,41/43,78**
Boat	StOMP	24,17/22,46	26,35/24,70	32,53/29,23
	IHT	25,46/24,07	30,07/27,62	34,69/31,17
	ℓ_1-AP	24,10/23,41	27,53/27,00	34,56/33,62
	ℓ_0-AP	*26,48/26,15*	*30,43/29,97*	*38,00/36,73*
	ℓ_1-AP+OP	**26,92/26,19**	**31,05/30,24**	**39,08/38,02**
Lena	StOMP	24,95/23,10	27,31/25,88	34,42/30,46
	IHT	27,09/24,86	*32,17*/28,63	39,38/32,54
	ℓ_1-AP	25,50/24,78	28,92/28,72	36,89/35,69
	ℓ_0-AP	*27,72/27,10*	31,63/*31,27*	*40,29/38,66*
	ℓ_1-AP+OP	**28,49/27,64**	**32,50/32,15**	**41,41/40,40**
Peppers	StOMP	24,48/22,97	26,85/25,66	33,14/29,69
	IHT	25,80/24,53	*31,57*/28,41	38,43/32,28
	ℓ_1-AP	24,36/24,47	28,46/28,80	35,99/34,89
	ℓ_0-AP	*27,43/26,85*	31,47/*30,88*	*38,81/37,35*
	ℓ_1-AP+OP	**27,82/27,43**	**32,26/32,13**	**40,12/39,17**

Cuadro 3.1: *Comparación detallada de los métodos usando DT-CWT con* 8 *escalas y Curvelets con* 6 *escalas en nuestro conjunto de prueba. Los números en negrita indican el método que proporciona la mejor aproximación para cada imagen y nivel de raleza. Los números en cursiva indican el segundo mejor método. Cada columna corresponde con un número de coeficientes, cuyo valor aparece normalizado por N. Las valores de raleza de las columnas normalizadas corresponden con* 2001, 6189 *y* 19096 *coeficientes activos respectivamente. Hemos extraído directamente los valores de PSNR de los experimentos, excepto para los de IHT, que han sido interpolados linealmente.*

En [17] también se presenta una versión dinámica que incrementa el número de coeficientes complejos usados en cada iteración (desde 12000 a 24000 en 30 iteraciones en el experimento que describen). En este caso obtienen 38,68 dB en la aproximación, todavía 0,41 dB por

DT-CWT/Curvelets		R/N		
Imagen	Método	0.05851	0.08552	1.4873
Barbara	StOMP	42,88/36,91	45,02/38,56	49,86/43,03
	IHT	44,73/39,69	47,69/42,78	53,44/47,16
	ℓ_1-AP	45,39/44,16	50,24/50,20	$>$ **100**/$>$ **100**
	ℓ_0-AP	*48,31/47,37*	*51,76/53,00*	*61,73/64,86*
	ℓ_1-AP+OP	**52,03/51,33**	**56,88/61,48**	$>$ **100**/$>$ **100**
House	StOMP	42,78/37,52	45,49/41,29	51,65/44,32
	IHT	45,85/42,02	50,01/43,95	55,98/47,94
	ℓ_1-AP	46,22/43,78	50,09/50,94	$>$ **100**/$>$ **100**
	ℓ_0-AP	*50,92/46,52*	*54,56/53,96*	*67,38/63,13*
	ℓ_1-AP+OP	**53,18/49,17**	**57,05/60,94**	$>$ **100**/$>$ **100**
Boat	StOMP	37,73/32,67	40,20/35,00	45,56/38,01
	IHT	40,61/36,07	44,29/38,27	52,50/42,61
	ℓ_1-AP	40,71/38,73	45,86/45,29	$>$ **100**/57,46
	ℓ_0-AP	*45,50/42,54*	*50,22/49,66*	*63,29/58,70*
	ℓ_1-AP+OP	**47,76/45,14**	**52,90/55,64**	$>$ **100**/**71,97**
Lena	StOMP	41,07/34,39	43,31/36,93	48,48/40,95
	IHT	43,65/38,07	46,15/41,61	51,57/45,47
	ℓ_1-AP	43,71/41,30	48,14/49,70	$>$ **100**/$>$ **100**
	ℓ_0-AP	*47,54/44,74*	*51,16/51,02*	*61,60/60,94*
	ℓ_1-AP+OP	**50,67/47,47**	**55,09/58,18**	$>$ **100**/$>$ **100**
Peppers	StOMP	39,10/33,41	41,91/36,31	47,61/39,61
	IHT	42,38/37,22	45,52/40,31	51,60/44,74
	ℓ_1-AP	41,92/39,75	49,84/46,17	$>$ **100**/58,62
	ℓ_0-AP	*45,76/43,04*	*52,13/50,40*	*63,78/61,45*
	ℓ_1-AP+OP	**48,12/45,66**	**56,05/56,30**	$>$ **100**/**72,91**

Cuadro 3.2: *Continuación de la Tabla 3.1. Los valores de raleza de las columnas normalizadas corresponden a* 38342, 56048 *y* 97471 *coeficientes activos respectivamente.*

debajo de nuestro resultado. Sin embargo, es fácil comprobar que esta diferencia se debe a la flexibilidad añadida que provee a nuestro esquema el uso independiente de las partes reales e imaginaria de los coeficientes complejos. Realmente, si no separamos los coeficientes complejos en nuestra implementación de ℓ_0-AP, nuestro resultado baja 1,31 dB por debajo de su versión dinámica (37,37 contra 38,68 dB). Los resultados óptimos deberían obtenerse usando umbrales dinámicos y separando las partes reales e imaginarias (ver Capítulo 4). El método en [89] mejora, utilizando heurísticos, los resultados de ℓ_0-AP.

3.4.3. Tiempo de computación

El tiempo que tarda cada iteración en ejecutarse en todos los métodos está dominado por una operación de análisis y otra de síntesis de la representación utilizada. Además, la búsqueda del umbral en ℓ_0-AP y ℓ_1-AP también se lleva un tiempo significativo. Otros métodos como DT+OP y IHT no requieren esa búsqueda de umbral, por lo que son relativamente más rápidos. Aún así, el tiempo consumido por los métodos depende en mayor medida del número medio de iteraciones que se han de llevar a cabo hasta que se cumplen los criterios de convergencia. En la Tabla 3.3 se muestra que ℓ_0-AP requiere más iteraciones para converger que ℓ_1-AP. Esta diferencia es mayor debido al uso de una restricción más fuerte para detectar la convergencia hacia reconstrucción perfecta (ver apartado 3.3.2).

Es importante apuntar, como hicimos en la Sección 3.1, que la mayoría de las aplicaciones no requieren tantas iteraciones. En este capítulo hemos favorecido optimizar la calidad para poder comparar con propiedad las cotas de rendimiento de cada método, más que alcanzar un compromiso práctico entre coste y calidad. De todas formas, como también nosotros hemos experimentado (ver Capítulo 4), las estrategias heurísticas que aplican umbrales dinámicos (por ejemplo, [97, 15, 17, 21, 40]) han demostrado ser intrínsecamente más rápidas que aquellas basadas en fijar la selección de coeficientes, el umbral o número de coeficientes seleccionados.

En nuestros experimentos, hemos usado un procesador Intel®, $Core^{TM}2$ Duo a 1.66 GHz con 2 GB de RAM. Como ejemplo de tiempos de ejecución sobre imágenes de 256×256, ℓ_0-AP tarda sobre 7 minutos en parar usando DT-CWT y sobre 1 hora usando Curvelets. Por otro lado, ℓ_1-AP se lleva sobre 3 minutos usando DT-CWT y 30 minutos usando Curvelets. De nuevo, estos tiempos no son representativos de una aplicación real, para la que se requerirán normalmente muchas menos iteraciones.

	♯ Iteraciones	
Métodos	DT-CWT	Curvelets
IHT	180	231
DT+OP	188	174
ℓ_1-AP	263	360
ℓ_1-AP+OP	333	440
ℓ_0-AP	495	920

Cuadro 3.3: *Número de iteraciones promediado sobre nuestro conjunto de imágenes de prueba usando DT-CWT con* 8 *escalas y Curvelets con* 6 *escalas para los diferentes métodos comparados.*

3.5. Conclusiones

En este capítulo se ha presentado un método de optimización, que llamamos ℓ_p-AP, basado en minimizar el error cuadrático medio de la reconstrucción de una imagen desde un vector de la representación con un marco de Parseval, dada una norma ℓ_p máxima para ese vector en esa representación. Dados p y R, el método consiste en proyectar ortogonalmente de forma alterna entre la bola ℓ_p de radio R, centrada en el origen, y el conjunto de vectores desde los cuales se reconstruye perfectamente la imagen. Hemos demostrado que se consigue un óptimo global cuando $p \geq 1$, y un óptimo local cuando $0 \leq p < 1$. Nos hemos centrado en los casos $p = 0$ y $p = 1$. El caso ℓ_0-AP se traduce en iteraciones utilizadas previamente, de forma heurística, en [17]. Por otro lado, ℓ_1-AP es similar al método desarrollado, en paralelo a nuestro trabajo, en [39, 40].

Hemos mostrado, a través de experimentos sistemáticos, que ℓ_0-AP mejora claramente la compactación de energía con respecto a ℓ_1-AP, para imágenes naturales con representaciones piramidales ampliamente usadas, el cual es óptimo para resolver el problema de relajación convexa. Además, este comportamiento se repite para todas las representaciones estudiadas. Este resultado demuestra que las condiciones de obtención del óptimo global para el problema de aproximación rala usando la relajación convexa no se cumplen usando imágenes naturales y representaciones típicas. Sin embargo, se puede mejorar el resultado de ℓ_1-AP optimizando *a posteriori* por mínimos cuadrados las amplitudes de los coeficientes seleccionados. Así, hemos comprobado que la selección de coeficientes activos de ℓ_1-AP es, en general, mejor que la de ℓ_0-AP. Sin embargo, en el siguiente capítulo veremos que esta selección también está lejos de ser óptima.

También hemos visto que ℓ_0-AP mejora a los métodos de umbralización iterativa basados en umbral fijo y a la versión de StOMP implementada. Se necesitarían pruebas más exhaustivas para establecer la superioridad de ℓ_0-AP sobre las técnicas voraces en general, pero el enorme coste computacional de las versiones más estrictas de estas técnicas impiden esta comparación. Entre los métodos existentes con antelación, la umbralización iterativa proporciona claramente el mejor rendimiento en cuanto a compactación. En cuanto a los métodos voraces, hemos experimentado un mejor comportamiento de la selección de coeficientes por umbralización directa que de la estrategia acumulativa seguida por los métodos de tipo OMP.

Aunque no se compara en detalle aquí, sino en el siguiente capítulo, los métodos basados en ajuste dinámico del umbral claramente tienen, en la actualidad, el mayor potencial de compactación. Pero estos métodos no han sido, hasta ahora, formulados matemáticamente, como hemos hecho

aquí con ℓ_p-AP. Es fácil adaptar ℓ_0-AP para incrementar iterativamente el número de coeficientes (como en [17]), pero nos hemos centrado aquí en un modelo de optimización completamente justificado en la teoría. Un hecho adicional es que, para algunas tareas de restauración (como, por ejemplo, eliminación de artefactos de cuantificación espacial, ver Capítulo 6) hemos experimentado que la mayor compactación de energía no siempre implica un mejor rendimiento en restauración.

Otra ventaja adicional de nuestro método es que usamos menos parámetros que otros métodos similares [15, 17, 21]. Sin embargo, todavía requiere establecer un radio para la bola ℓ_p. Esta desventaja se elimina con el método presentado en el siguiente capítulo.

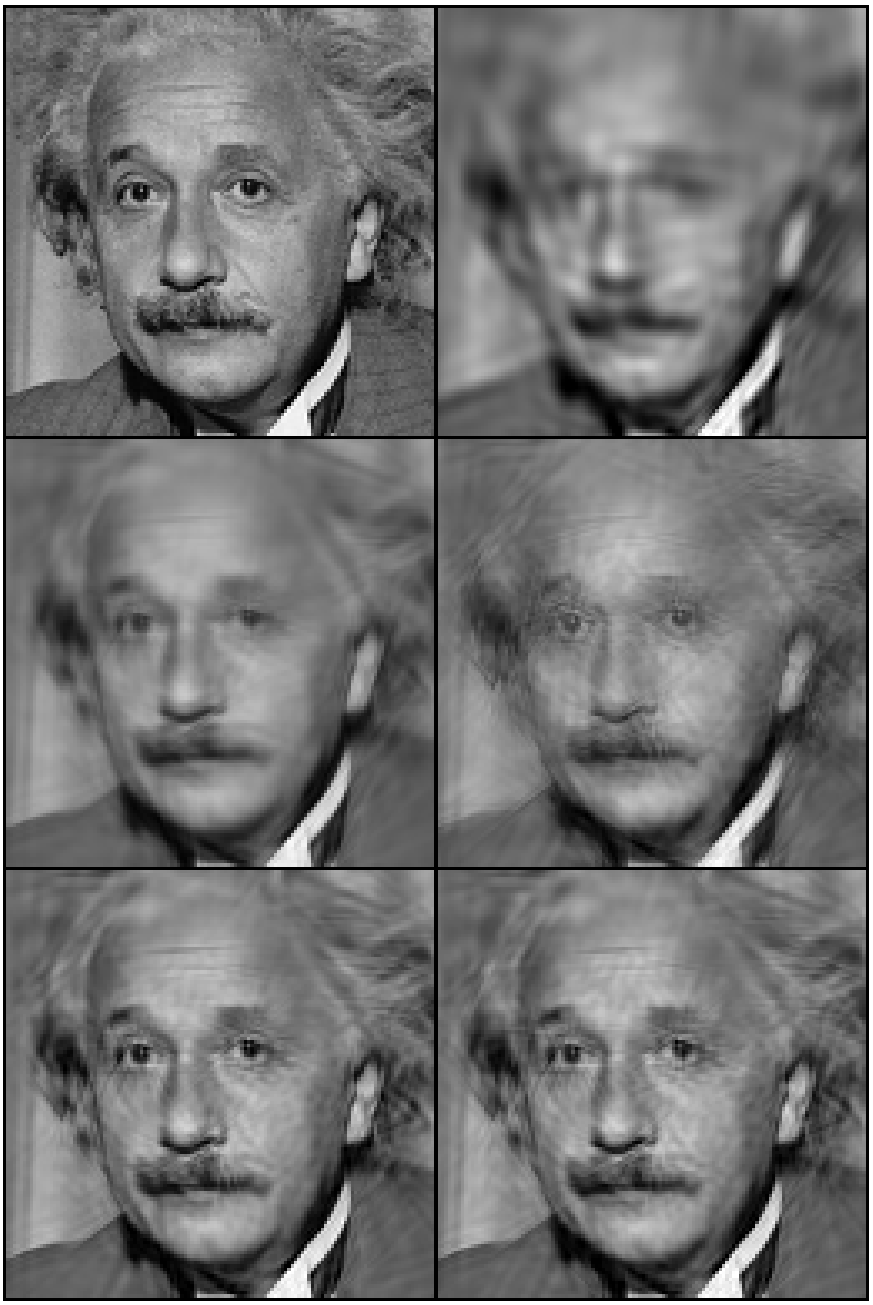

Figura 3.6: *Comparación visual de los métodos usando* $0{,}0765 \cdot N$ *coeficientes Curvelets en la imagen Einstein, donde* N *es el número de píxeles de la imagen. Los resultados están recortados a tamaño* 128×128*, comenzando en el píxel (*71, 41*), para aumentar la visibilidad.* **Columna izquierda**, *desde arriba hacia abajo: imagen original, resultado de* ℓ_1*-AP (*30,85 *dB), y* ℓ_1*-AP+OP (*33,52 *dB).* **Columna derecha**, *desde arriba hacia abajo: resultado de StOMP (*28,66 *dB), IHT (*29,10 *dB) y* ℓ_0*-AP (*32,98 *dB).*

Capítulo 4

Aproximación rala aplicando descenso de gradiente

En este capítulo se deriva matemáticamente otro método para resolver el problema de aproximación rala, que es más potente y eficiente que el descrito en el capítulo anterior, pero mantiene la ventaja de ser una solución a un problema de optimización explícito. Su diseño comienza desde la siguiente pregunta: ¿Es posible hacer descenso en la dirección opuesta al gradiente en el criterio a minimizar en la Ecuación (2.4)? La respuesta, debido a la naturaleza discontinua de la norma ℓ_0 es "no directamente". Sin embargo, mostraremos cómo se puede escribir un criterio equivalente que nos permita calcular la dirección del gradiente. Obtendremos entonces una versión generalizada del método IHT y plantearemos una demostración original de que el punto fijo de sus iteraciones es un mínimo local de la función de coste que tenemos entre manos.

Además, para evitar quedar atrapados en mínimos locales poco favorables, aplicaremos una técnica de enfriamiento determinista (*deterministic annealing*, en inglés) similar a otros algoritmos de optimización global no convexa [122, 123, 21]. Llamamos ℓ_0-GM al método resultante. Mostraremos a través de experimentos que ℓ_0-GM es competitivo con el actual estado de la técnica en cuanto a compactación de energía, mejorando tanto a ℓ_0-AP como a nuestra versión de ℓ_1-AP optimizada por mínimos cuadrados (ℓ_1-AP+OP).

Realizaremos una derivación análoga que nos llevará a derivar el método IST a través del descenso en la dirección opuesta al gradiente de una función que es equivalente al criterio a minimizar en el problema de relajación convexa (Ecuación (2.5)). También derivaremos una variante convexa de ℓ_0-GM, a la que llamaremos ℓ_1-GM. Mostraremos que consigue resultados comparables al resto de métodos de relajación convexa, y describiremos los

casos prácticos en donde debe usarse.

El método ℓ_0-GM se basa en umbralización dinámica. La idea de reducir el umbral conforme se realizan más iteraciones no es nueva [15, 17, 21, 93, 94, 19]. Sin embargo, que nosotros sepamos, es la primera vez que se deriva formalmente un método basado en estas ideas como solución directa al problema de aproximación rala. Además, hasta donde llega nuestro conocimiento, nadie ha analizado con cierta profundidad las razones por las que esta solución es tan favorable.

En este capítulo, comenzaremos en la Sección 4.1 por reformular la función de coste del problema de aproximación rala (discontinua y no restringida) de forma continua y restringida. En la Sección 4.2 derivaremos el método IHT generalizado como solución local del problema de aproximación rala. Después justificaremos el uso de umbrales descendientes en la Sección 4.3. En la Sección 4.4 se darán los detalles de implementación de ℓ_0-GM y en la Sección 4.5 se discutirán los experimentos realizados comparando capacidad de compactación de energía de ℓ_0-GM frente a los métodos analizados en el capítulo anterior. Por último, derivaremos los métodos IST y ℓ_1-GM en la Sección 4.6, y los comparamos con ℓ_1-AP. La Sección 4.7 concluye el capítulo.

4.1. Formulación continua de la función de coste

Por conveniencia, repetimos aquí la formulación del problema de aproximación rala de la Ecuación (2.4):

$$\hat{\mathbf{a}}^0(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}. \tag{4.1}$$

La función de coste asociada no sólo no es convexa sino que tampoco es continua. Esto impide que se pueda calcular directamente su gradiente. A continuación vamos a derivar una nueva función continua y restringida equivalente, sobre la que sí podremos calcular el gradiente. Vamos a partir de la siguiente formulación:

$$(\hat{\mathbf{a}}, \hat{\mathbf{b}}) = \arg\min_{\mathbf{a},\mathbf{b}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2 \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}, \tag{4.2}$$

y demostramos la igualdad $\hat{\mathbf{a}} = \hat{\mathbf{a}}^0(\lambda)$. En primer lugar expresamos la Ecuación (4.2) como:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda \min_{\mathbf{b}}\{\|\mathbf{b} - \mathbf{a}\|_2^2 \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}\}. \tag{4.3}$$

Vemos que la minimización interna es la proyección ortogonal de $\mathbf{a}$ sobre el subespacio afín $S(\mathbf{\Phi}, \mathbf{x})$ de reconstrucción perfecta de $\mathbf{x}$. La expresión de esta proyección fue definida en la Ecuación (3.4), y repetimos aquí su expresión por conveniencia:

$$P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\mathbf{a}) = \mathbf{a} + \mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{a}).$$

Sustituyendo en la Ecuación (4.2) obtenemos:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{\Phi}^T(\mathbf{\Phi}\mathbf{a} - \mathbf{x})\|_2^2\}.$$

Dado que $\mathbf{\Phi}$ es un marco de Parseval, se llega finalmente a:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{\Phi}\mathbf{a} - \mathbf{x}\|_2^2\} = \hat{\mathbf{a}}^0(\lambda),$$

tal y como queríamos demostrar. A continuación, para obtener una formulación con una función de coste continua y restringida que sólo dependa de $\mathbf{b}$, partimos de la Ecuación (4.2) e intercambiamos las variables de la minimización con respecto a la Ecuación (4.3):

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2\} \text{ s.a. } \mathbf{\Phi}\mathbf{b} = \mathbf{x}\}.$$

Es fácil ver que minimizar la función de coste para el vector $\mathbf{a}$ en este caso es equivalente a minimizar independientemente para cada índice. Expresamos la función de coste como $c(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{M} c'(a_i, b_i)$, donde:

$$c'(a, b) = \begin{cases} 1 + \lambda(b - a)^2, & |a| > 0 \\ \lambda b^2, & |a| = 0. \end{cases}$$

Dado $\mathbf{b}$, tenemos que si el valor $\tilde{a}_i(b_i)$ que minimiza $c'(a_i, b_i)$ no es cero, entonces $\tilde{a}_i(b_i) = b_i$, y $c'(\tilde{a}_i(b_i), b_i) = 1$. Así, tenemos que:

$$c(\tilde{\mathbf{a}}(\mathbf{b}), \mathbf{b}) = \sum_{i=1}^{M} \min(1, \lambda b_i^2).$$

La Figura 4.1 muestra un ejemplo en una dimensión de este mínimo (usando $\lambda = 1$). Dado algún valor de λ, notamos θ al valor que cumple $\lambda\theta^2 = 1$. Por consiguiente:

$$\theta = \lambda^{-\frac{1}{2}},$$

y tenemos que:

$$\tilde{a}_i(b_i) = \begin{cases} b_i, & |b_i| > \theta \\ 0, & |b_i| \leq \theta. \end{cases}$$

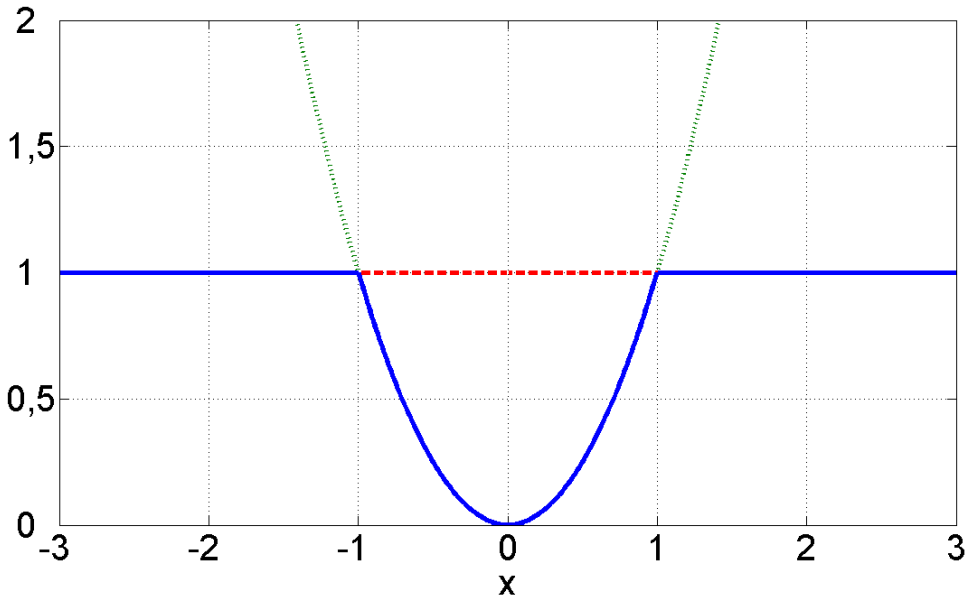

Figura 4.1: *La línea gruesa muestra la función mínimo entre* $y(x) = 1$ *(línea intermitente) e* $y(x) = x^2$ *(línea punteada).*

Esto es una operación de umbralización dura con umbral θ, que notamos $\tilde{\mathbf{a}}(\mathbf{b}) = S_0(\mathbf{b}, \theta)$. Sustituyendo $\mathbf{a}$ por $S_0(\mathbf{b}, \theta)$ en la Ecuación (4.2):

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\|S_0(\mathbf{b}, \theta)\|_0 + \lambda\|\mathbf{b} - S_0(\mathbf{b}, \theta)\|_2^2 \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}.$$

Al evaluar este criterio en cada coeficiente de $\mathbf{b}$, uno de los dos términos (el de fidelidad o el de raleza) se anula. Así, podemos expresar lo mismo de la siguiente forma:

$$\begin{aligned} \hat{\mathbf{b}} &= \arg\min_{\mathbf{b}}\{C_0(\mathbf{b}, \theta) \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}, \qquad (4.4) \\ \hat{\mathbf{a}} &= S_0(\hat{\mathbf{b}}, \theta), \end{aligned}$$

donde:

$$C_0(\mathbf{b}, \theta) = \sum_{i=1}^{M} \text{mín}\left(1, \left(\frac{b_i}{\theta}\right)^2\right). \qquad (4.5)$$

4.2. Minimización local con norma ℓ_0: IHT

El gradiente de la nueva función de coste, sin restringir, es $\nabla C_0(\mathbf{b}, \theta) = \mathbf{c}$, donde:

$$c_i = \begin{cases} 0, & |b_i| > \theta \\ \frac{2}{\theta^2} b_i, & |b_i| \leq \theta. \end{cases}$$

Esto se puede expresar más fácilmente como:

$$\nabla C_0(\mathbf{b}, \theta) = \frac{2}{\theta^2} \left(\mathbf{b} - S_0(\mathbf{b}, \theta)\right),$$

La proyección de este gradiente sobre el subespacio afín de reconstrucción perfecta, $S(\mathbf{\Phi}, \mathbf{x})$, es:

$$\nabla^{S(\mathbf{\Phi},\mathbf{x})} C_0(\mathbf{b}, \theta) = (\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi})\nabla C_0(\mathbf{b}, \theta).$$

Las iteraciones del método de descenso de gradiente son:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \alpha \nabla^{S(\mathbf{\Phi},\mathbf{x})} C_0(\mathbf{b}^{(k)}, \theta).$$

Como esta proyección es el componente del gradiente en el espacio nulo de $\mathbf{\Phi}$, $\mathbf{b}^{(k)}$ siempre tiene reconstrucción perfecta, sin importar el valor de α que usemos. Sustituyendo con la expresión del gradiente tenemos:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \frac{2\alpha}{\theta^2}(\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi}) \left(\mathbf{b}^{(k)} - S_0(\mathbf{b}^{(k)}, \theta)\right).$$

Una condición necesaria y, en nuestro caso, suficiente para llegar a un mínimo local de la función de coste es que:

$$\nabla^{S(\mathbf{\Phi},\mathbf{x})} C(\mathbf{b}^*, \theta) = \mathbf{0}.$$

Esta será la condición de convergencia de las iteraciones anteriores. Esto significa que, si estas iteraciones convergen, lo hacen a un mínimo local de la función de coste de la Ecuación (4.5).

Nótese que la elección de $\alpha = \alpha_0 = \frac{1}{2\lambda} = \frac{\theta^2}{2}$ minimiza la función de coste no restringida de la Ecuación (4.5) para un paso de descenso, resultando en:

$$\mathbf{b}^{(k+1)} = S_0(\mathbf{b}^{(k)}, \theta) + \mathbf{\Phi}^T \left(\mathbf{x} - \mathbf{\Phi} S_0(\mathbf{b}^{(k)}, \theta)\right).$$

La expresión previa es el mismo método de umbralización dura iterativa, IHT, que ya describimos en el apartado 2.3.3.

Hemos mostrado que este procedimiento proporciona, al converger, un mínimo local en el criterio de aproximación rala clásico de la Ecuación (4.1). Sin embargo, en general, la elección del valor de α que minimiza en un sólo paso la función de coste sin restricciones (α_0) no es óptima en términos de velocidad de convergencia. Hemos experimentado que se puede obtener una convergencia más rápida si se usa $\alpha \sim 1{,}85\alpha_0$.

Recientemente, hemos conocido que en [29] también demostraron, de forma paralela e independiente a nuestro trabajo[1], que el punto de

[1]Este trabajo fue publicado en Abril de 2007, mientras que el nuestro apareció en Agosto del mismo año [12].

convergencia de las iteraciones del método IHT es un mínimo local de la función de coste. Sin embargo, ellos también demuestran que el método converge si los autovalores de $(\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi})$ están entre 0 y 1, donde $\mathbf{I}$ es la matriz identidad de tamaño $M \times M$.

La Figura 4.2 muestra algunas curvas de convergencia usando umbrales fijos y diferentes valores de α y usando como representación DT-CWT con 8-escalas[2], cuyo factor de redundancia es 4. Podemos ver que, a pesar de la llamativa sencillez de este método, hacer descenso de gradiente para un valor de λ hasta la convergencia es demasiado costoso en términos computacionales. Por añadidura, sabemos que el mínimo local obtenido con este método es claramente peor que el obtenido con el método presentado en el anterior capítulo (ℓ_0-AP, ver Sección 3.4).

4.3. Minimización global con norma ℓ_0: ℓ_0-GM

A continuación proponemos una alternativa eficiente a IHT y ℓ_0-AP inspirada en técnicas deterministas de optimización global y que reduce dramáticamente el coste computacional, incrementando además significativamente la capacidad de compactación de energía en pocos coeficientes.

La función de coste de la Ecuación (4.5) se puede reescribir como:

$$C_0(\mathbf{b}, \theta) = \sum_{i=1}^{M} (1 - h(b_i/\theta)), \tag{4.6}$$

donde se ha definido $h(x) = \text{máx}(1 - x^2, 0)$ como el arco de parábola invertida centrada en cero y que va desde -1 a 1, alcanzando su valor máximo en cero con una amplitud de 1. Se puede ver la forma de esta función en la Figura 4.3. Podemos reescribir, entonces, el problema de optimización (4.4) de la siguiente manera:

$$\begin{aligned} \hat{\mathbf{b}} &= \arg\max_{\mathbf{b}} C'(\mathbf{b}, \theta), \\ \hat{\mathbf{a}} &= S_0(\hat{\mathbf{b}}, \theta), \\ C'(\mathbf{b}, \theta) &= \sum_{i=1}^{M} h(b_i/\theta) = M - C_0(\mathbf{b}, \theta). \end{aligned}$$

[2]Salvo indicación expresa, esta será la representación usada a lo largo de todo el capítulo. Hemos experimentado que otras representaciones, como por ejemplo Curvelets, ofrecen un comportamiento cualitativamente similar.

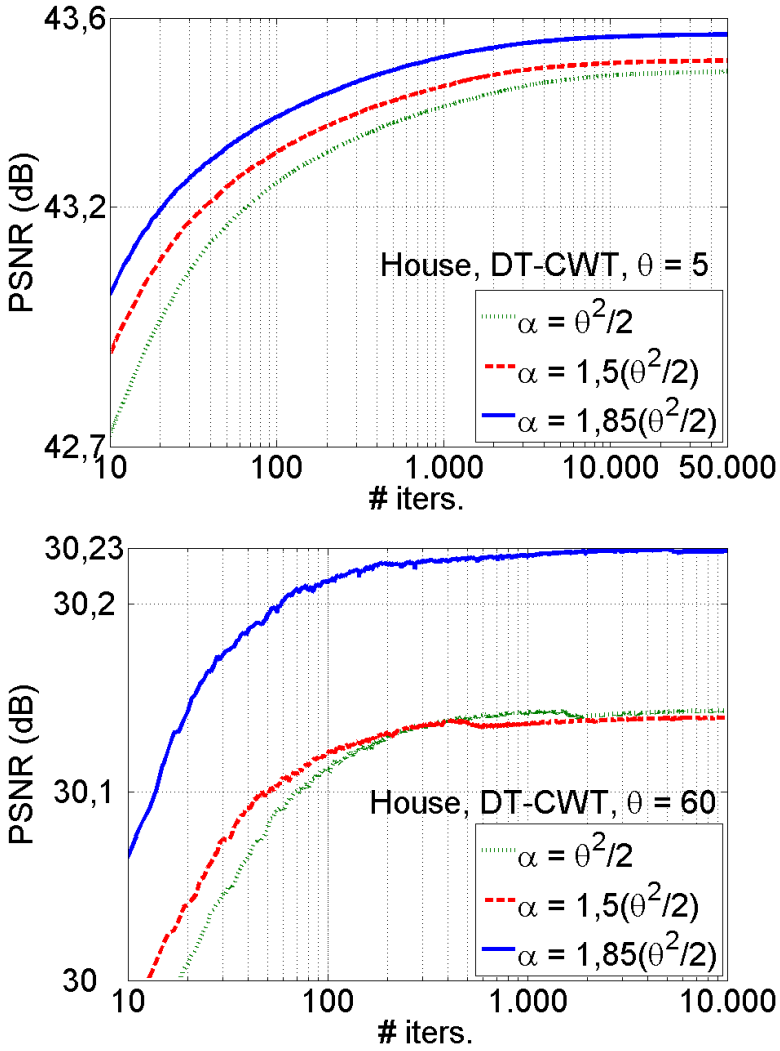

Figura 4.2: **Arriba**, *curvas de convergencia de IHT con un umbral bajo ($\theta = 5$) y tres diferentes valores de α. Hemos usado la imagen* House *y DT-CWT con* 8 *escalas.* **Abajo**, *lo mismo para un umbral más alto ($\theta = 60$).*

Y es fácil reescribir esta expresión en términos de una función de coste infinitamente apuntada, $C_\delta(\mathbf{b})$, convolucionada con un filtro de suavizado:

$$C'(\mathbf{b}, \theta) \propto C_\delta(\mathbf{b}) * H(\mathbf{b}/\theta),$$

donde $*$ representa el operador de convolución; $H(\mathbf{b}) = \prod_{i=1}^{M} h(b_i)$, $C_\delta(\mathbf{b}) = \sum_{i=1}^{M} \delta(b_i)$, y el factor de proporcionalidad es $A(\theta)^{-M+1}$, con $A(\theta) = \int_{-\theta}^{\theta} h(x/\theta) = 4\theta/3$. El factor de escala, $A(\theta)^{-M+1}$, resulta de integrar $H(\mathbf{b}/\theta)$ a lo largo de todas las dimensiones excepto aquellas de la función

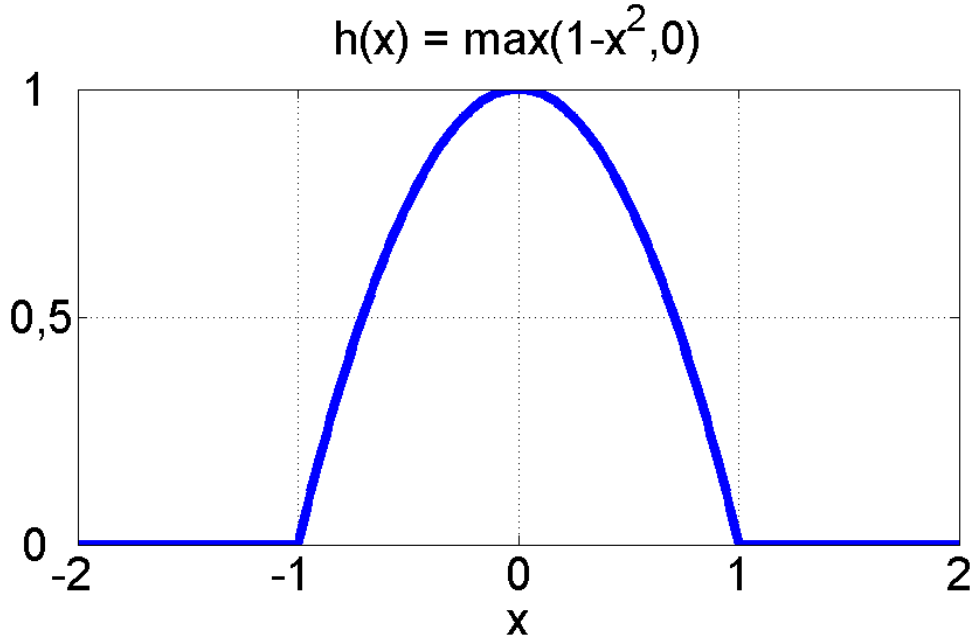

Figura 4.3: *Parábola invertida en el intervalo* $[-1, 1]$, *centrada en* 0 *y con máximo* 1. *Fuera de ese intervalo es constante a* 0.

delta correspondiente (en cuya dimensión se lleva a cabo una convolución unidimensional), y es irrelevante en términos de minimizar el vector $\mathbf{b}$.

La Figura 4.2 muestra que es más rápido encontrar un óptimo local cuando θ es grande, o, lo que es lo mismo, λ es pequeño, lo que corresponde a una función de coste suave. Además, teniendo un buen candidato para el óptimo global de un λ dado, podemos esperar obtener un buen resultado si buscamos desde él el óptimo más cercano correspondiente a un λ parecido pero ligeramente superior. De aquí podemos derivar el siguiente método. Fijar un λ pequeño, hacer descenso de gradiente hasta encontrar la convergencia, entonces fijar un λ ligeramente más grande, volver a hacer descenso de gradiente desde el anterior punto de convergencia hasta encontrar un nuevo óptimo, y continuar así hasta llegar al valor de λ deseado. Llamamos a este método ℓ_0-GM (de Minimización Gradual, *Gradual Minimization* en inglés). Una versión aproximada más rápida y más sencilla consiste en incrementar muy lentamente λ en cada iteración, de forma que se reduce drásticamente el número de iteraciones. De hecho, ambas versiones son equivalentes en el límite si el incremento de λ en cada iteración es infinitesimal. En términos del umbral θ, empezamos con el valor más alto posible (es decir, preservando la amplitud mayor de $\mathbf{a}^{LS}$), y disminuimos θ lentamente en cada iteración, hasta llegar al valor deseado. En la Figura 4.4 se ilustra el concepto que guía este método con un

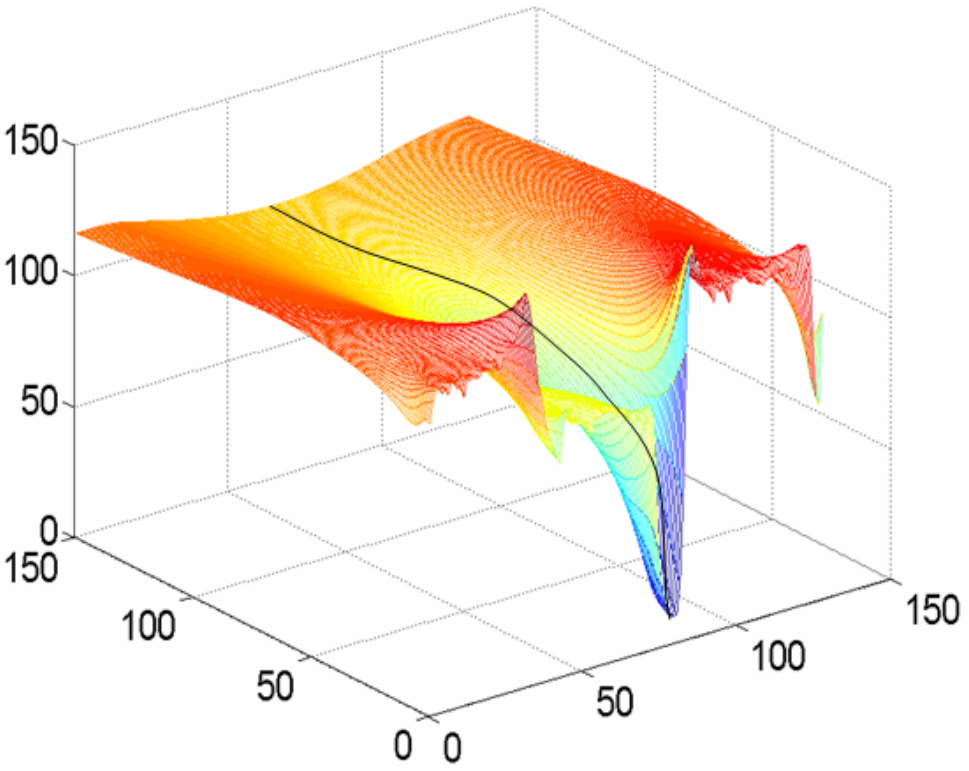

Figura 4.4: *Función de una dimensión con múltiples mínimos, progresivamente suavizada hasta tener sólo uno. La línea negra continua indica el camino que une los mínimos globales a través de la escala. Hemos usado, como filtro de suavizado, una versión normalizada (en área) de $h(x)$ (Ver Figura 4.3).*

ejemplo de una función con múltiples mínimos suavizada hasta conseguir una función con un sólo mínimo, y donde se muestra el camino que une los mínimos globales en cada escala. En este ejemplo existe continuidad entre los mínimos globales a través de las diferentes escalas, siendo esta una condición necesaria para que ℓ_0-GM pueda alcanzar el óptimo global. Esto en general no ocurre en casos prácticos. Hemos comprobado que el método encuentra el óptimo global de la función de coste para altos niveles de raleza (usando alrededor de decenas de coeficientes DT-CWT). Sin embargo, las condiciones que aseguren la optimalidad de la solución no entran en el ámbito de esta Tesis, donde nos interesa estudiar más bien el comportamiento de los métodos en condiciones prácticas.

La idea de suavizar una función de coste para evitar quedar atrapados en óptimos locales poco favorables está muy relacionada con otros esquemas de enfriamiento determinista, tales como [122, 123]. Algunos autores han propuesto esta idea como un heurístico para obtener algoritmos que

favorezcan la compactación de energía, usando bien umbralización suave [21] o dura [15, 93]. Sin embargo, los autores referidos no han planteado sus soluciones en términos de resolver un problema de optimización bien fundamentado.

En la Figura 4.5 se pueden ver, por un lado (líneas intermitentes), las trayectorias de convergencia del método IHT para diferentes umbrales fijos (lo que corresponde con buscar un óptimo local haciendo descenso de gradiente para un λ fijo). Por otro lado, se pueden ver dos trayectorias (círculos y curva continua) correspondientes a disminuir de forma exponencial el umbral $\theta^{(k)} = \theta^{(0)}\beta^k$ para dos valores de β diferentes. Mientras β se acerca a 1, la compactación de energía es mayor, pero también la convergencia es más lenta. Nosotros, como otros autores [17, 21], hemos experimentado que, en la práctica, la actualización dinámica decreciendo el umbral de forma exponencial proporciona un mejor compromiso entre coste computacional y calidad del resultado que otras funciones, tales como un descenso lineal. Mediante la reducción dinámica del umbral de ℓ_0-GM, no sólo estamos reduciendo dramáticamente el número de iteraciones requeridas para llegar a la convergencia, sino que también obtenemos una fidelidad significativamente más alta para el mismo nivel de raleza.

En la Figura 4.6 (arriba) se muestra una familia de curvas de fidelidad-raleza para diferentes valores de β. La teoría nos dice que la mejor curva posible tiene una asíntota hacia reconstrucción perfecta en N. Podemos apreciar como ℓ_0-GM aproxima esta asíntota conforme β se acerca a 1. Esta observación es aún más importante si consideramos que conseguir un óptimo global es mucho menos probable para niveles bajos de raleza que para los altos, porque el número de óptimos locales se incrementa rápidamente con λ.

4.3.1. Usando una sóla solución para cualquier nivel de raleza

Si optimizamos utilizando ℓ_0-GM para un conjunto de valores de λ, acabaremos teniendo múltiples soluciones, una por cada valor que tome el umbral en su recorrido descendente. ¿Qué criterio se debe seguir para elegir una solución en particular? Es posible encontrar un valor de umbral, θ_0, cuyo mínimo asociado a la función de coste, $C_0(\hat{\mathbf{b}}(\theta_0), \theta_0)$, pueda extenderse a $C_0(\hat{\mathbf{b}}(\theta_0), \theta)$ para aproximar el mínimo de la función de coste asociada a otro umbral θ (esto es, tal que $C_0(\hat{\mathbf{b}}(\theta_0), \theta) \approx C_0(\hat{\mathbf{b}}(\theta), \theta)$, para todo $\theta > \theta_0$)? La respuesta, sorprendentemente, es "sí". Este problema tiene un impacto práctico importante, porque en este caso podríamos usar $\hat{\mathbf{a}}'(\theta) = S_0(\hat{\mathbf{b}}(\theta_0), \theta)$ como sustituto, casi igual de bueno, de $\hat{\mathbf{a}} = S_0(\hat{\mathbf{b}}(\theta), \theta)$, pero

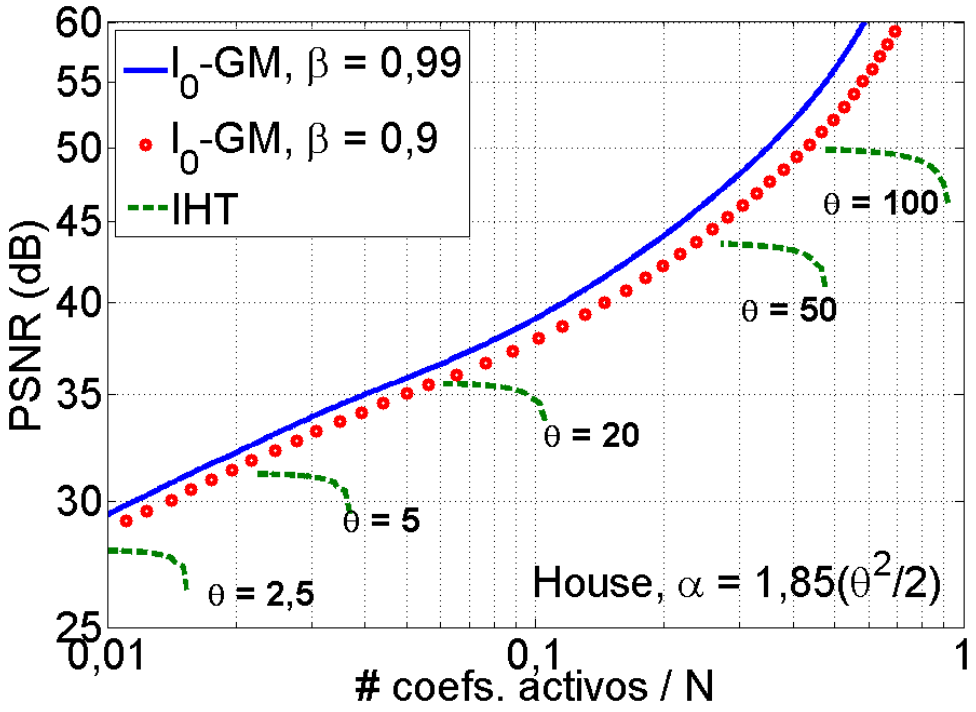

Figura 4.5: *Resultados de fidelidad-raleza usando ℓ_0-GM con $\beta = 0{,}9$ (círculos, $1{,}5 \cdot 10^2$ iteraciones) y $\beta = 0{,}99$ (línea continua, $1{,}5 \cdot 10^3$ iteraciones), comparado con IHT, usando varios umbrales fijos (líneas intermitentes, 10^5 iteraciones cada una). Se usa la imagen* House *y DT-CWT con 8 escalas.*

sin requerir usar, y almacenar[3], $\hat{\mathbf{b}}(\theta)$.

Hemos explicado antes como, para conseguir una buena solución para un cierto $\lambda_i > \lambda_j$, es bueno empezar desde la solución asociada a λ_j y, desde ahí, refinar hasta llegar a la solución asociada a λ_i. Esto parece implicar que las buenas soluciones para valores más altos de λ deberían ser razonablemente buenas también para valores más bajos. En la Figura 4.6 (abajo) se muestra la curva de fidelidad-raleza (rendimiento de la solución al problema de aproximación rala) obtenida aplicando diferentes umbrales sobre la solución obtenida usando un valor muy alto de λ posible. Como podemos ver, para $\beta < 0{,}99$ los resultados no solamente no han empeorado, sino que incluso mejoran el rendimiento de la curva obtenida durante la aplicación directa de ℓ_0-GM. Esto significa que con solo una solución de optimización para un determinado nivel de raleza (el más bajo posible) es suficiente para dar una buena solución para todos los valores de λ considerados. Esto es una

[3]Sin embargo, esto no evita tener que calcular $\hat{\mathbf{b}}(\theta)$, si $\theta > \theta_0$, porque el cálculo de $\hat{\mathbf{b}}(\theta_0)$ con ℓ_0-GM requiere calcular previamente $\hat{\mathbf{b}}(\theta)$, para todo $\theta > \theta_0$ (en la práctica, un muestreo suficientemente denso del intervalo $[\theta_{max}, \theta_0]$)

ventaja muy importante en la práctica, porque significa que no necesitamos almacenar todos los $\hat{\mathbf{b}}(\theta)$ para elegir el umbral θ, correspondiente a un cierto λ, en tiempo real. Esto permite, por ejemplo, adaptarnos a un ancho de banda del canal variable en comunicaciones, y ofrece, en general, un buen compromiso entre fidelidad y raleza.

4.4. Implementación

Hemos experimentado con varios marcos de Parseval diferentes, al igual que hicimos con el método ℓ_p-AP. Hemos elegido DT-CWT para mostrar los experimentos en este capítulo, aunque las conclusiones cualitativas de los experimentos son similares al usar cualquiera de ellos. Junto a Curvelets, esta representación ofrece los mejores resultados de compactación entre las comparadas. Además, la implementación MATLAB® disponible [120] es mucho más rápida que la de Curvelets [121].

Al igual que en el capítulo anterior, los coeficientes complejos de DT-CWT han sido separados en sus partes real e imaginaria para realizar un tratamiento homogéneo de todos los coeficientes. Además, también se ha añadido una escala extra compuesta de un sólo coeficiente, que refleja la media global de la imagen, para optimizar el rendimiento en cotas extremadamente altas de raleza.

En nuestra implementación del método ℓ_0-GM hemos comprobado que los mejores resultados se obtienen cuando el rango de descenso de θ es el mayor posible. Por eso, el primer valor que utilizamos es el segundo mayor valor en amplitud entre los coeficientes de la respuesta lineal a la imagen (para que se escoja al menos un coeficiente en la primera iteración). El proceso decrece el umbral hasta llegar al valor deseado. Este valor depende de la aplicación práctica de cada instancia del método. Siguiendo lo dicho en la sección anterior, nosotros escogemos para con un umbral bajo, para conseguir un buen rendimiento en todos los niveles de raleza.

4.5. Resultados y discusión del método ℓ_0-GM

La Figura 4.7 muestra varias curvas fidelidad-raleza para los siguientes métodos: ℓ_0-GM con $\beta = 0{,}99$, ℓ_0-AP, $l1$-AP post optimizado por mínimos cuadrados (ℓ_1-AP+OP), IHT y StOMP. Ver Capítulo 3 para más detalles sobre estos métodos. El incremento en el rendimiento de ℓ_0-GM con respecto los otros métodos es muy destacable. Estos datos muestran claramente que, minimizando directamente la norma ℓ_0, se obtienen, en las condiciones del

experimento, mejores mínimos locales al problema de aproximación rala que los conseguidos resolviendo el problema de relajación convexa, incluso optimizando por mínimos cuadrados los coeficientes del soporte elegido. Una importante diferencia de ℓ_0-GM con los métodos basados en proyecciones alternas es que aquí podemos barrer todo el rango de raleza en una sola ejecución del método, en vez de ejecutar muchas iteraciones para cada nivel. Las Tablas 4.1 y 4.2 muestran los datos numéricos de la Figura 4.7.

Existen, en la literatura, otras estrategias de descenso exponencial del umbral que pueden dar, dependiendo del caso concreto, resultados ligeramente superiores a los de ℓ_0-GM. Por ejemplo, en [17], el número de coeficientes preservados en cada iteración se incrementa linealmente.

La Figura 4.8 muestra una comparación visual de las aproximaciones ralas de los métodos usando $0{,}04 \cdot N$ coeficientes DT-CWT para la imagen *Einstein*. En ella vemos que ℓ_0-GM conserva significativamente mejor la información perceptualmente relevante de la original.

4.6. Descenso de gradiente para minimizar la norma ℓ_1: IST & ℓ_1-GM

4.6.1. Formulación alternativa de la función de coste de la relajación convexa

En la Ecuación (2.5) describimos el problema de relajación convexa, que volvemos a repetir aquí por conveniencia:

$$\hat{\mathbf{a}}^1(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}, \tag{4.7}$$

La función de coste asociada, a diferencia de cuando se utiliza la norma ℓ_0, es convexa y, por lo tanto, continua. Sin embargo, nos interesa realizar una transformación similar a la de aquel caso. La demostración de que la solución $\hat{\mathbf{a}}$ del problema:

$$(\hat{\mathbf{a}}, \hat{\mathbf{b}}) = \arg\min_{\mathbf{a},\mathbf{b}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2 \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}, \tag{4.8}$$

es equivalente a $\hat{\mathbf{a}}^1(\lambda)$ es análoga al caso de la norma ℓ_0. Podemos expresar $\hat{\mathbf{b}}$ como:

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\min_{\mathbf{a}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2\} \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}. \tag{4.9}$$

Primero, encontramos la expresión que minimiza la función de coste interna dado $\mathbf{b}$. Esta puede descomponerse como un sumatorio de un término por

		♯ coefs. activos/N		
Imagen	Método	0,00868	0,02536	0,04761
Barbara	StOMP	24,48	25,55	26,45
	IHT	23,16	27,25	29,89
	ℓ_0-AP	24,89	28,67	31,89
	ℓ_1-AP+OP	*25,24*	*29,38*	*32,60*
	ℓ_0-GM	**26,18**	**30,47**	**33,93**
House	StOMP	26,40	28,27	29,82
	IHT	24,08	30,35	32,91
	ℓ_0-AP	27,33	31,50	34,16
	ℓ_1-AP+OP	*27,75*	*32,01*	*34,73*
	ℓ_0-GM	**28,85**	**33,18**	**35,65**
Boat	StOMP	20,75	23,89	24,89
	IHT	16,82	24,74	27,09
	ℓ_0-AP	21,57	25,76	27,86
	ℓ_1-AP+OP	*21,96*	*26,12*	*28,37*
	ℓ_0-GM	**23,77**	**27,32**	**30,13**
Lena	StOMP	23,21	24,67	26,02
	IHT	17,27	26,22	28,93
	ℓ_0-AP	23,62	27,20	30,09
	ℓ_1-AP+OP	*24,17*	*27,96*	*30,95*
	ℓ_0-GM	**25,19**	**29,29**	**32,34**
Peppers	StOMP	21,43	24,19	25,28
	IHT	16,76	25,15	27,93
	ℓ_0-AP	22,61	26,39	29,17
	ℓ_1-AP+OP	*22,81*	*26,85*	*29,74*
	ℓ_0-GM	**24,03**	**28,43**	**31,57**

Cuadro 4.1: *Fidelidad (PSNR, en dB) para varios niveles de raleza, usando las imágenes de nuestro conjunto de prueba, cinco métodos distintos, y utilizando DT-CWT con* 8 *escalas. Los números en negrita indican el método que da la mejor aproximación para cada imagen y nivel de raleza, y las cursivas indican el segundo mejor. Las columnas corresponden a* 569, 1662 *y* 3120 *coeficientes activos. Hemos extraído directamente los valores de PSNR de los experimentos, excepto los de IHT, que han sido interpolados linealmente.*

cada componente de los vectores involucrados, de forma que el vector $\tilde{\mathbf{a}}^s(\mathbf{b})$ que minimiza el criterio más interno de la Ecuación (4.9) es:

$$\tilde{\mathbf{a}}^s(\mathbf{b}) = \min_{\mathbf{a}}\{\sum_{i=1}^{M} c(a_i, b_i)\},$$

		♯ coefs. activos/N		
Imagen	Método	0,18648	0,50209	0,98687
Barbara	StOMP	34,58	42,39	46,63
	IHT	35,68	44,12	48,56
	ℓ_0-AP	40,39	47,63	55,64
	ℓ_1-AP+OP	*41,57*	*51,22*	$>$ **100**
	ℓ_0-GM	**43,05**	**54,93**	$>$ **100**
House	StOMP	33,86	41,83	47,89
	IHT	38,14	44,61	50,75
	ℓ_0-AP	38,40	49,25	58,98
	ℓ_1-AP+OP	*39,20*	*51,25*	*62,31*
	ℓ_0-GM	**43,47**	**56,12**	$>$ **100**
Boat	StOMP	29,06	36,42	42,05
	IHT	32,07	39,30	45,65
	ℓ_0-AP	33,52	43,29	58,91
	ℓ_1-AP+OP	*34,25*	*45,02*	$>$ **100**
	ℓ_0-GM	**38,09**	**50,10**	$>$ **100**
Lena	StOMP	31,70	39,48	45,06
	IHT	34,33	42,30	48,21
	ℓ_0-AP	36,85	45,66	55,51
	ℓ_1-AP+OP	*37,79*	*47,95*	*63,72*
	ℓ_0-GM	**40,97**	**52,97**	$>$ **100**
Peppers	StOMP	31,60	38,50	43,17
	IHT	33,98	41,19	48,66
	ℓ_0-AP	37,20	44,84	57,09
	ℓ_1-AP+OP	*38,41*	*47,13*	$>$ **100**
	ℓ_0-GM	**39,93**	**51,63**	$>$ **100**

Cuadro 4.2: *Continuación de la Tabla 4.1. Las columnas corresponden, respectivamente, a* 12221, 32905 *y* 64682 *coeficientes activos.*

donde $c(a,b) = |a| + \lambda(b-a)^2$. La derivada en a de esta función es $\frac{\partial c(a,b)}{\partial a} = d + 2\lambda(a-b)$, donde:

$$d = \begin{cases} 1, & a > 0 \\ -1, & a < 0 \\ 0, & a = 0. \end{cases} \tag{4.10}$$

Para el caso $a > 0$, tenemos que:

$$\frac{\partial c(a,b)}{\partial a} = 1 + 2\lambda(a-b).$$

Igualando a cero obtenemos:

$$a = b - \frac{1}{2\lambda},$$

de donde se deduce que $b > \frac{1}{2\lambda}$, ya que $\lambda > 0$ por definición. Para el caso $a < 0$, análogamente, obtenemos:

$$a = b + \frac{1}{2\lambda},$$

y entonces $b < -\frac{1}{2\lambda}$. Uniendo estos dos casos tenemos que:

$$a = sign(b) \cdot (|b| - \frac{1}{2\lambda}),$$

cuando $|b| > \frac{1}{2\lambda}$.

Por otra parte, cuando $|b| \leq \frac{1}{2\lambda}$, el valor de a que minimiza la función de coste asociada cambia de signo con respecto a b. Como para cada cuadrante de la recta real sólo consideramos valores en el mismo cuadrante, eso implica que el mínimo está en cero.

Así, aplicando estos resultados en nuestro problema, obtenemos que el vector $\tilde{\mathbf{a}}^s(\mathbf{b})$ es el resultado de una umbralización suave de $\mathbf{b}$ con umbral $\theta = \frac{1}{2\lambda}$. Notamos esta operación como $\tilde{\mathbf{a}}^s(\mathbf{b}) = S_1(\mathbf{b}, \theta)$. Sustituyendo en la Ecuación (4.8):

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\|S_1(\mathbf{b}, \theta)\|_1 + \lambda\|\mathbf{b} - S_1(\mathbf{b}, \theta)\|_2^2 \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}.$$

Y finalmente, dado que la función de coste de la expresión previa es separable como una suma de términos independientes para cada índice, podemos escribir:

$$\begin{aligned} \hat{\mathbf{b}} &= \arg\min_{\mathbf{b}}\{C_1(\mathbf{b}, \theta) \text{ s.a. } \mathbf{\Phi b} = \mathbf{x}\}, \\ \hat{\mathbf{a}} &= S_1(\hat{\mathbf{b}}, \theta), \end{aligned}$$

donde:

$$C_1(\mathbf{b}, \theta) = \sum_{i=1}^{M} \text{mín}\left(|b_i| - \frac{\theta}{2}, \frac{b_i^2}{2\theta}\right). \tag{4.11}$$

4.6.2. Minimización de la función de coste con umbral fijo: IST

La derivación del método basado en descenso de gradiente sobre la función de coste $C_1(\mathbf{b}, \theta)$ es análogo a la expuesto para la función $C_0(\mathbf{b}, \theta)$. De la Ecuación 4.11 obtenemos:

$$\nabla C_1(\mathbf{b}, \theta) = \frac{1}{\theta}(\mathbf{b} - S_1(\mathbf{b}, \theta)),$$

y, tras proyectar sobre el espacio afín de reconstrucción perfecta, $\nabla^{S(\mathbf{\Phi},\mathbf{x})} C_1(\mathbf{b}, \theta) = (\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi})\nabla C_1(\mathbf{b}, \theta)$, tenemos las siguientes iteraciones:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \frac{\alpha}{\theta}(\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi}) \left(\mathbf{b}^{(k)} - S_1(\mathbf{b}^{(k)}, \theta)\right).$$

Una condición necesaria (y suficiente, en este caso) para llegar al mínimo global de la función $C_1(\mathbf{b}, \theta)$ es que $\nabla^{S(\mathbf{\Phi},\mathbf{x})} C_1(\mathbf{b}^*, \theta) = \mathbf{0}$, que es la condición de convergencia de las iteraciones. La elección de $\alpha = \alpha_0 = \frac{1}{2\lambda} = \theta$ nos lleva al conocido método IST:

$$\mathbf{b}^{(k+1)} = S_1(\mathbf{b}^{(k)}, \theta) + \mathbf{\Phi}^T \left(\mathbf{x} - \mathbf{\Phi} S_1(\mathbf{b}^{(k)}, \theta)\right).$$

Se ha demostrado [30, 39] que este procedimiento proporciona un mínimo global en el criterio de relajación convexa de la Ecuación (4.7).

Sin embargo, al igual que en el caso $p = 0$, en general la elección $\alpha = \theta$, aunque minimiza en α en un sólo paso la función de coste sin restricciones, no es óptima en términos de velocidad de convergencia. Hemos experimentado, también aquí, que se puede obtener una convergencia más rápida si se usa $\alpha \sim 1{,}85\alpha_0$, aunque ahora la diferencia con $\alpha = \alpha_0$ es muy pequeña. La Figura 4.9 muestra las correspondientes curvas de convergencia en las mismas condiciones que se han mostrado para IHT en la Figura 4.2. La convergencia es mucho más rápida y uniforme que en el caso IHT, debido a la convexidad de la función de coste.

4.6.3. Una minimización convexa más eficiente: ℓ_1-GM

Podemos derivar un método equivalente a IST y ℓ_1-AP, pero a menudo más eficiente. Para cualquier valor de θ, la función de coste $C_1(\mathbf{b}, \theta)$ es convexa y por lo tanto puede hallarse su mínimo global usando IST o ℓ_1-AP. Conforme crece el valor de θ, o, equivalentemente, decrece λ, la parte cuadrática de la función domina, lo que provoca una convergencia más rápida (como se ve en la Figura (4.9)). Además, sabemos que en este

caso sí existe continuidad entre los óptimos globales de la función para los distintos valores de λ, condición que viene asegurada porque sólo hay un mínimo y ese mínimo debe ser una función continua de λ. Esta propiedad asegura que, partiendo del óptimo global para un λ dado encontraremos muy rápidamente el óptimo para un λ ligeramente superior. De aquí derivamos un método similar a ℓ_0-GM pero para el caso de minimizar la norma ℓ_1. Es decir, fijamos un λ pequeño, hacemos descenso de gradiente con IST hasta converger, entonces fijamos un λ ligeramente más grande, volvemos a aplicar IST desde el anterior punto de convergencia, y repetimos hasta llegar al valor deseado de λ. Llamamos a este método ℓ_1-GM. Al igual que para ℓ_0-GM, una aproximación más rápida y más sencilla consiste en incrementar muy lentamente λ en cada iteración, de forma que se reduce enormemente el número de iteraciones.

En la Figura 4.10 se comparan las trayectorias de convergencia del método IST para cuatro diferentes umbrales con dos trayectorias de ℓ_1-GM, correspondientes a disminuir de forma exponencial de el umbral $\theta^{(k)} = \theta^{(0)}\beta^k$ para dos valores de β diferentes. Podemos ver como, en la práctica, el resultado de descenso exponencial con $\beta = 0{,}99$ necesita menos iteraciones ($1{,}5 \cdot 10^3$ por 10^3 de IST) para proporcionar un resultado casi óptimo para muchos valores de λ, tomando en total casi tantas iteraciones como IST toma para dar un sólo resultado. Además, usando $\beta = 0{,}9$ se consigue una buena aproximación al resultado óptimo con un número de iteraciones bastante reducido. También en este caso, como otros autores (por ejemplo, [42]), hemos experimentado que el descenso exponencial del umbral proporciona un mejor compromiso entre coste computacional y calidad del resultado que otras funciones decrecientes, tales como un descenso lineal.

En la Figura 4.11 se muestra una familia de curvas de fidelidad-raleza para diferentes valores de β. También se ha añadido los datos correspondientes a ℓ_1-AP (ver Capítulo 3) como referencia. Se puede apreciar como la fidelidad obtenida con ℓ_1-GM se aproxima al resultado de ℓ_1-AP conforme β se acerca a 1. Recordemos que se han ejecutado muchas iteraciones de ℓ_1-AP para estudiar su comportamiento en convergencia, y por lo tanto estos datos aproximan muy bien el óptimo global teórico del problema de relajación convexa asociado.

4.6.4. Ventajas prácticas de ℓ_1-GM

Hemos observado que ℓ_1-AP es más rápido si la solución que buscamos tiene un nivel de raleza medio-alto conocido (equivalentemente, un valor de λ medio-bajo), mientras que ℓ_1-GM es mejor cuando se tienen valores altos de λ. La Figura 4.12 muestra una comparación, usando la imagen

Barbara y DT-CWT con 8 escalas, de las iteraciones que se necesitan para alcanzar un resultado cercano al óptimo, para varios niveles de raleza, mediante los métodos ℓ_1-AP (descrito en el Capítulo 3) y ℓ_0-GM (con $\alpha = \theta$ y $\beta = 0{,}99$). Nótese que, para niveles bajos de raleza, ℓ_1-GM es más rápido. Este caso aparece a menudo en la práctica, cuando buscamos representaciones estrictamente ralas o cuando se resuelven algunos problemas de restauración, en concreto si se ha perdido alguna información localizada (ver Capítulo 6). Es decir, cuando el objetivo es representar perfectamente la imagen original o alguna parte observada de ella.

4.7. Conclusiones

En este capítulo se ha derivado un método de optimización, basado en umbralización iterativa con ajuste dinámico del umbral, para resolver el problema de aproximación rala. A diferencia de otros métodos heurísticos similares existentes, como por ejemplo [17, 21, 19], nuestra aproximación está justificada en la teoría y formulada como solución a un problema clásico de optimización.

Lo primero que hemos hecho ha sido reformular el problema de aproximación rala para ir a un problema de optimización equivalente, pero usando una función de coste continua y restringida, en vez de usar la función de coste original, discontinua y sin restringir, que no permite la aplicación de herramientas clásicas dde optimización. A continuación, hemos derivado un método realizando descenso en la dirección opuesta al gradiente, proyectado sobre el subespacio afín de vectores que cumplen la restricción, $S(\mathbf{\Phi}, \mathbf{x})$. El método resultante es una versión generalizada de IHT, si $p = 0$. Para terminar, hemos descrito el método ℓ_0-GM basado en actualización dinámica del umbral, mientras se desciende en la dirección opuesta al gradiente de la función de coste. Este método se ha justificado como un tipo de enfriamiento determinista que equivale a expresar la función de coste como el resultado de convolucionar una función de referencia infinitamente apuntada con un filtro de suavizado cada vez menos suave.

Los experimentos realizados demuestran que ℓ_0-GM ofrece mucho mejores resultados de compactación que otros métodos, como ℓ_0-AP, ℓ_1-AP+OP, IHT o StOMP. De hecho, su comportamiento cuando el número de coeficientes seleccionados se aproxima al número de píxeles de la imagen es asintóticamente óptimo, pues aproxima bien la asíntota teórica que la curva ideal tiene en este punto. Este método está a la altura del estado de la técnica en aproximación rala. Estos resultados demuestran que, bajo las condiciones prácticas presentadas en esta Tesis, tratar de resolver directamente el

problema de aproximación rala puede llevar a mínimos locales mejores que aquél que se obtiene resolviendo el problema de relajación convexa.

Análogamente, hemos derivado los métodos IST generalizado y ℓ_1-GM a través del descenso en la dirección opuesta al gradiente de una función restringida equivalente a la función de coste del problema de relajación convexa. Ambos métodos resuelven de forma óptima el problema (ℓ_1-GM cuando $\beta \to 1$). Hemos visto que el uso de ℓ_1-GM, comparado con IST y ℓ_1-AP, está recomendado para aquellas aplicaciones donde estamos restringidos a preservar alguna parte (o toda) de la observación.

En el futuro, usaremos las ideas presentadas aquí para otras normas. Aunque las matemáticas requeridas para usar cuasi-normas intermedias ($0 < p < 1$) pueden ser más complicadas, merecerá la pena, probablemente, para mejorar los resultados de compactación.

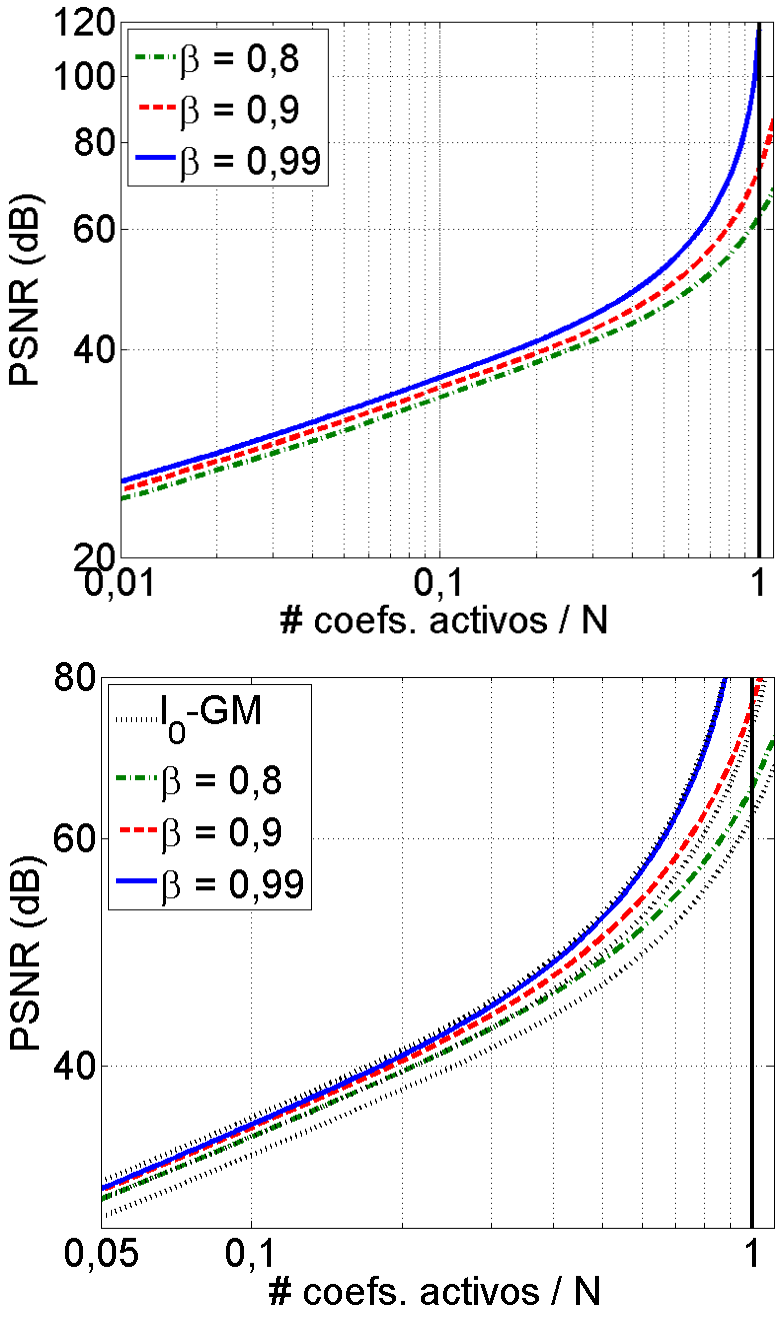

Figura 4.6: **Arriba**, *resultados de fidelidad de la aproximación rala promediados en el conjunto de prueba usando ℓ_0-GM con $\alpha = 1{,}85(\theta^2/2)$, tres diferentes valores de β y usando DT-CWT con* 8 *escalas.* **Abajo**, *calidad de la reconstrucción desde los coeficientes más altos en amplitud del vector resultante de ejecutar ℓ_0-GM para un valor muy alto de λ (muy baja raleza), y para los mismos valores de β. Las curvas punteadas corresponden con las del panel de arriba. El eje de ordenadas ha sido re-escalado para mejorar la visibilidad.*

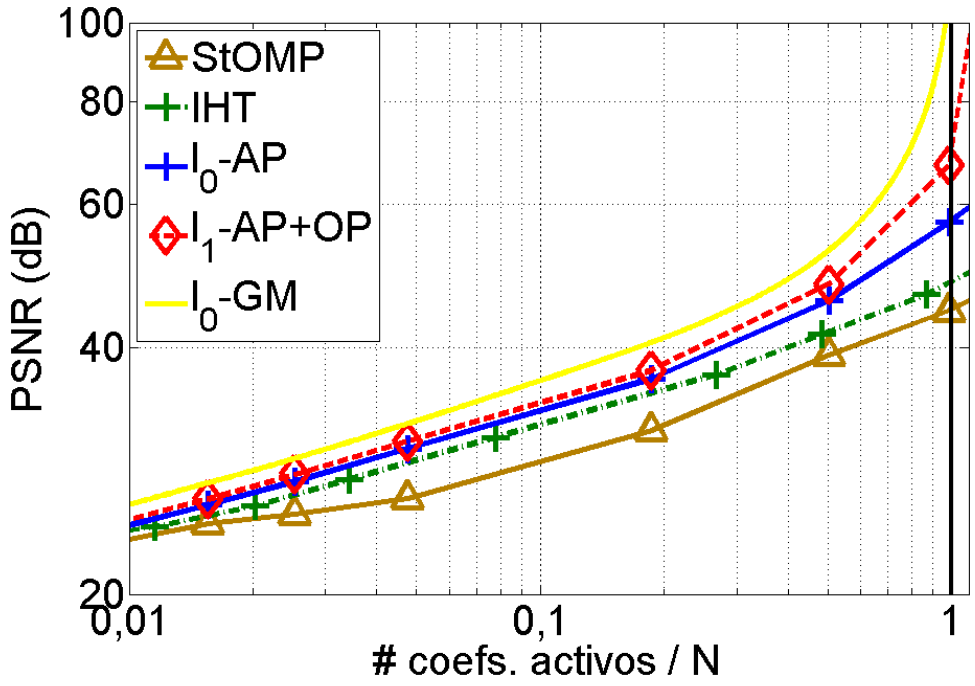

Figura 4.7: *Resultados de aproximación rala de nuestro método de optimización no convexa (ℓ_0-GM) promediados para las imágenes del conjunto de prueba, y comparados con otros métodos vistos previamente (StOMP, IHT, ℓ_0-AP y ℓ_1-AP+OP).*

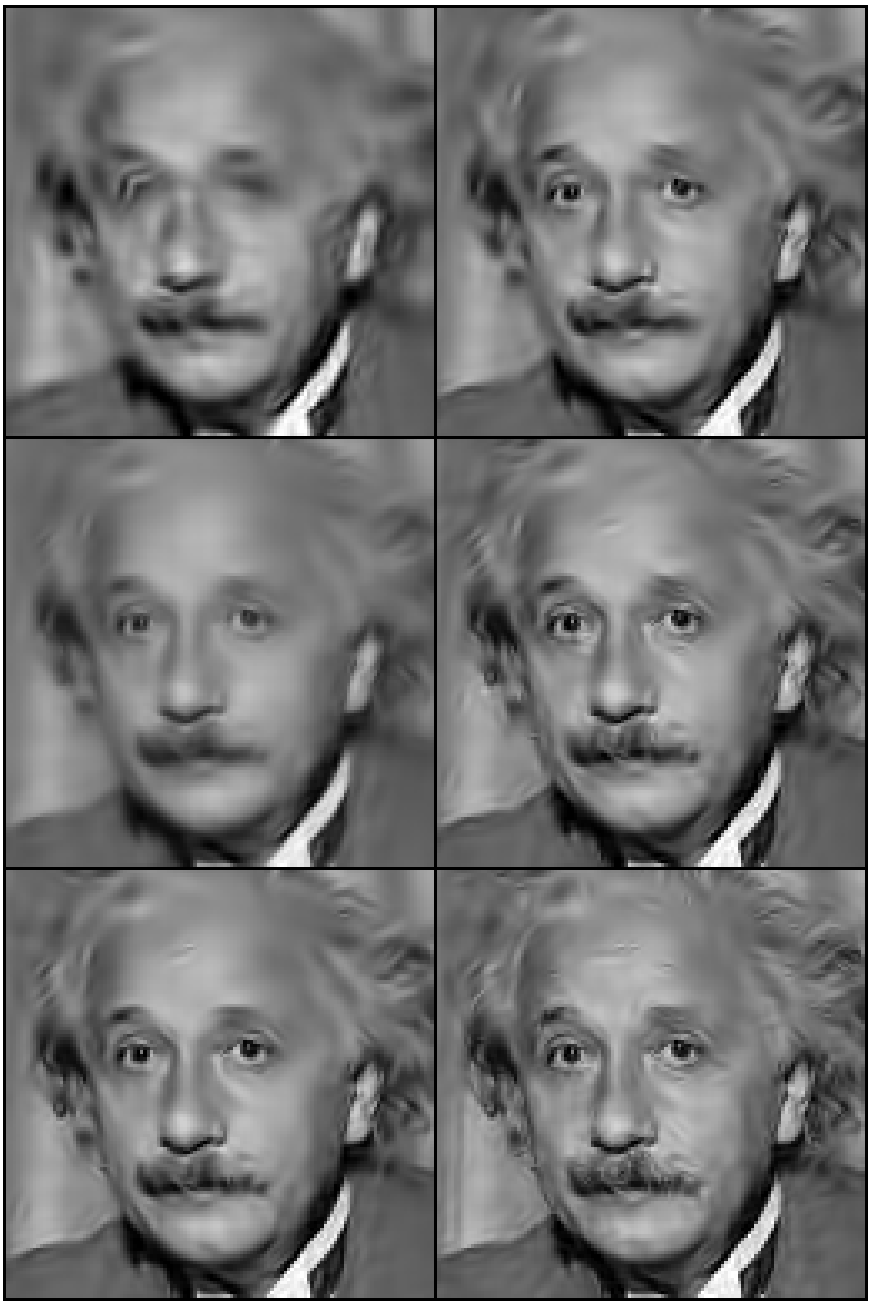

Figura 4.8: *Recorte a tamaño* 64×64 *de la reconstrucción de la imagen* Einstein *usando* $0{,}04 \cdot N$ *(*2605*) coeficientes activos, utilizando DT-CWT con* 8 *escalas, para varios métodos.* **Arriba - izquierda**, *resultado de StOMP, implementado como se vio en la Sección 3.4.1 (*28,98 *dB).* **Arriba - derecha**, *IHT (*31,20 *dB).* **Centro - izquierda**, ℓ_1*-AP (*29,70 *dB).* **Centro - derecha**, ℓ_0*-AP (*31,97 *dB).* **Abajo - izquierda**, ℓ_1*-AP+OP (*32,38 *dB).* **Abajo - derecha**, ℓ_0*-GM (*33,28 *dB).*

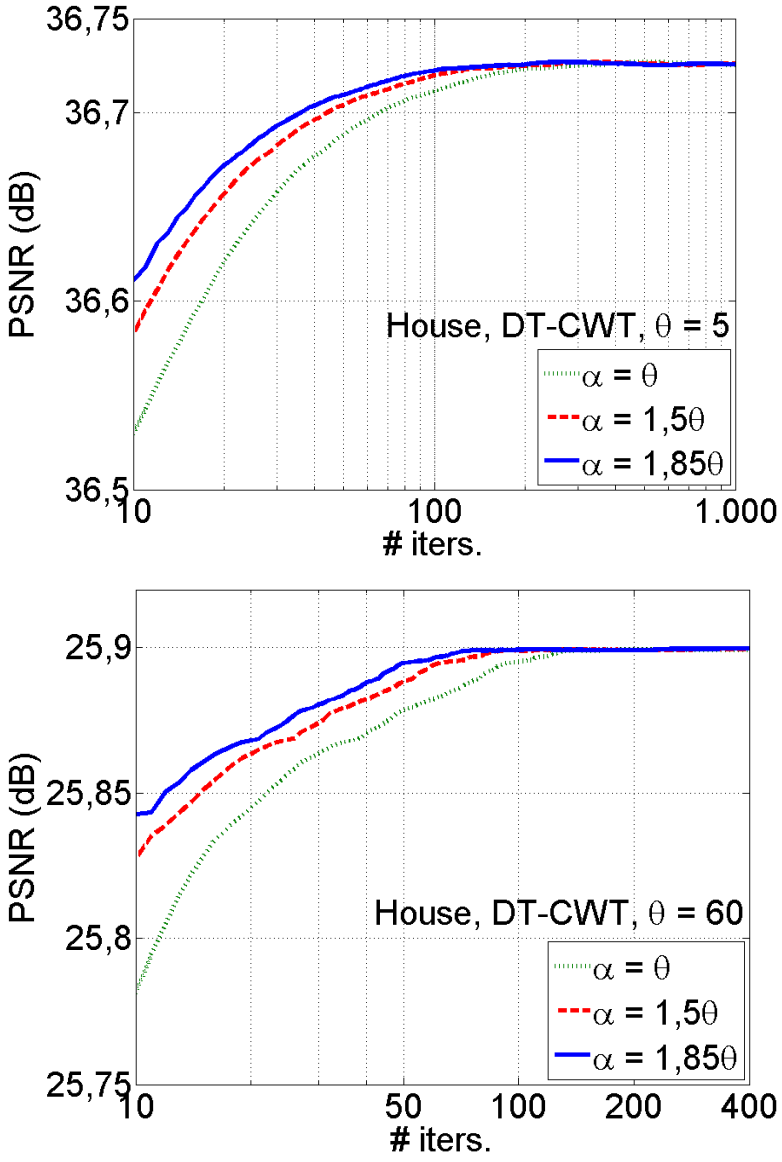

Figura 4.9: **Arriba**, *curvas de convergencia de IST con un umbral bajo* ($\theta = 5$) *y tres diferentes valores de* α. *Hemos usado la imagen* House *y DT-CWT con* 8 *escalas.* **Abajo**, *lo mismo para un umbral más alto* ($\theta = 60$).

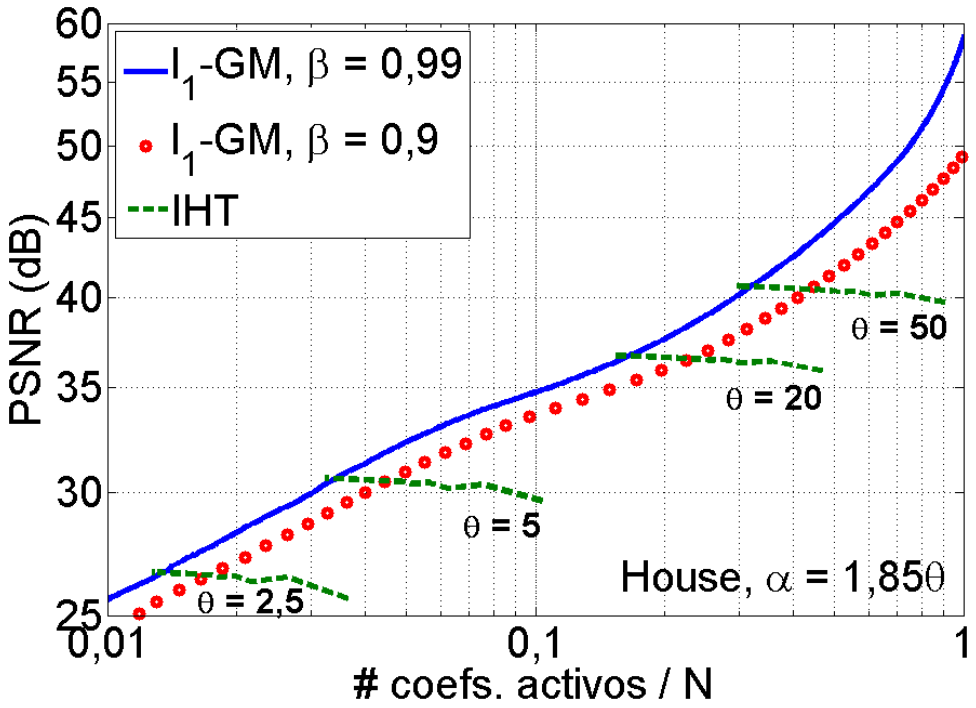

Figura 4.10: *Resultados de fidelidad-raleza usando ℓ_1-GM con $\beta = 0{,}9$ (círculos, $1{,}5 \cdot 10^2$ iteraciones) y $\beta = 0{,}99$ (línea continua, $1{,}5 \cdot 10^3$ iteraciones), comparado con IHT, usando varios umbrales fijos (líneas intermitentes, 10^3 iteraciones cada una). Se usa la imagen* House *y DT-CWT con 8 escalas.*

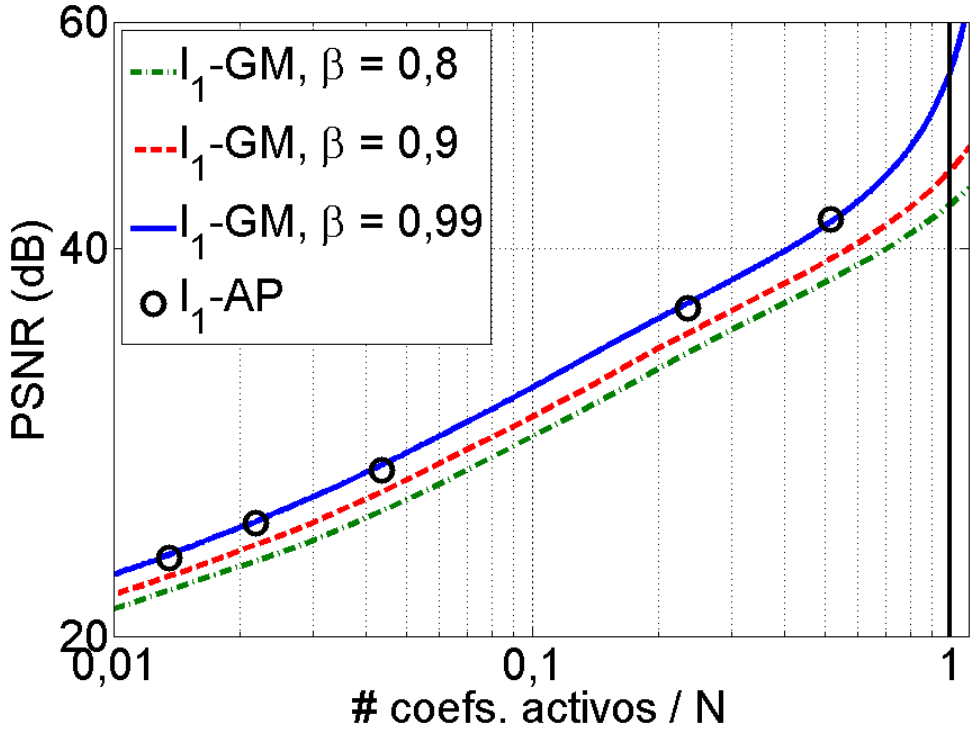

Figura 4.11: *Resultados de aproximación rala promediados en el conjunto de prueba usando ℓ_1-GM con $\alpha = 1{,}85\theta$, diferentes valores de β y usando DT-CWT con 8 escalas. También se muestran el resultado correspondiente a ℓ_1-AP.*

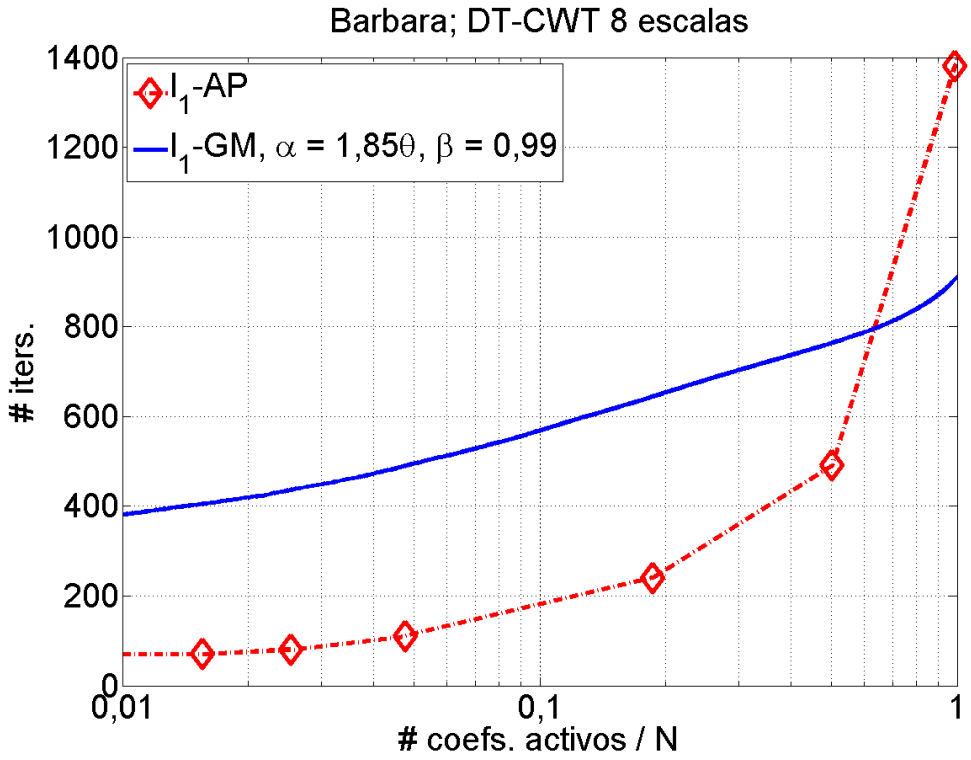

Figura 4.12: *Iteraciones llevadas a cabo por los métodos ℓ_1-AP y ℓ_0-GM ($\alpha = 1{,}85\theta$, $\beta = 0{,}99$) para alcanzar un resultado cercano al óptimo para distintos niveles de raleza. Usamos la imagen* Barbara *y DT-CWT con* 8 *escalas.*

Capítulo 5

Aplicación a restauración de imágenes

Consideramos ahora que la observación que tenemos es incompleta. Por ejemplo, que ha sufrido la pérdida de algunos píxeles, componentes de color, bits, resolución, etc. Nuestro objetivo es estimar esa información perdida. Utilizamos una aproximación basada en maximizar la fidelidad a la observación regularizada por un modelo *a priori* basado en propiedades estadísticas de las imágenes naturales.

Nuestro modelo de fidelidad a la observación se basa en el concepto de consistencia. Decimos que una imagen es consistente con una observación degradada cuando, aplicando esta degradación a la imagen, se obtiene la observación dada. Por consiguiente, para poder aplicar este concepto en la práctica, la degradación debe de ser perfectamente reproducible a partir de la imagen observada. En algunos casos, no es posible identificar la degradación precisa que ha sufrido una observación dada (por ejemplo, con ruido blanco gaussiano). Pero en otros sí lo es (por ejemplo, pérdida de bits, píxeles, componentes cromáticas, resolución, etc.). Llamamos a estas últimas degradaciones deterministas *a posteriori*.

Nuestro modelo *a priori* se basa en fomentar la raleza de la estimación. Esto se justifica mediante la observación de que la mayoría de las degradaciones disminuyen la raleza de la representación (por ejemplo, usando ondículas) con respecto a la imagen original [124, 125, 26]. En la Figura 12.1 vemos un ejemplo. La columna de la izquierda corresponde a un recorte de la imagen *Peppers* (arriba) y a una sub-banda de alta frecuencia de la respuesta lineal usando DT-CWT a esta imagen (abajo). La columna de la derecha corresponde a la pérdida aleatoria del 40 % de píxeles de la imagen (arriba) y a la sub-banda correspondiente (abajo). Se aprecia que la energía está menos concentrada en la sub-banda degradada.

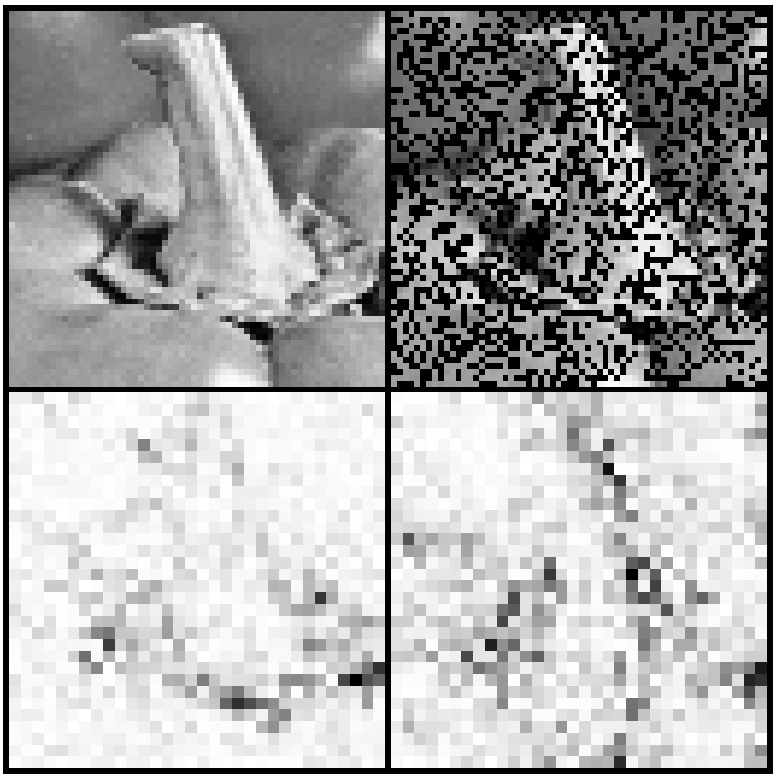

Figura 5.1: **Arriba a la izquierda**, *recorte de* Peppers *empezando en la fila* 111, *columna* 91. **Abajo a la izquierda**, *mismo recorte de la sub-banda de alta frecuencia del análisis lineal de* Peppers *con DT-CWT* 8 *escalas, correspondiente a* -45^o *de orientación. Previamente se ha duplicado el tamaño de esta sub-banda mediante replicado de píxeles para coincidir con el tamaño de la imagen.* **Arriba a la derecha**, *imagen degradada poniendo a cero, aleatoriamente, el* 40 % *de los píxeles.* **Abajo a la derecha**, *sub-banda correspondiente a la imagen degradada.*

Las dos observaciones hechas sobre las imágenes naturales en el Capítulo 2 (concentración de energía en respuesta lineal y aumento de raleza con métodos no lineales) nos llevan a dos variantes distintas para describir el conocimiento *a priori* que tenemos sobre ellas. Por un lado, podemos favorecer que la imagen estimada se exprese como combinación lineal de pocos vectores de la representación. Esto implica maximizar la raleza de los coeficientes de síntesis. Muchos autores han utilizado este concepto con anterioridad para diferentes problemas de restauración [46, 31, 66]. Denominamos a este concepto raleza en sentido de síntesis (SS, de *Synthesis-sense Sparseness* en inglés).

Aunque el uso de la aproximación SS es perfectamente legítimo y razonablemente exitoso en la práctica, se podría objetar la falta de una base empírica directa. La aproximación bayesiana tradicional para la restauración

de imágenes se basa en construir modelos *a priori* de la imagen que reflejen el comportamiento típico de las señales en muchas observaciones previas. Sin embargo, los coeficientes de síntesis de una representación óptimamente rala no pueden observarse directamente e, incluso, en ocasiones, ni siquiera son calculables, con exactitud, en la práctica (porque, en general, la solución óptima global no está disponible).

Siguiendo esta discusión, parece conceptualmente más correcto usar un modelo *a priori* estadístico basado en observaciones directas de la representación, que describa la forma típica de los coeficientes de la transformación lineal de las imágenes naturales. Llamamos a este concepto raleza en sentido de análisis (AS, de *Analysis-sense Sparseness*, en inglés). La aproximación AS es una extensión natural de muchos trabajos previos que, bajo diferentes contextos, han usado modelos de densidad ralos para la imagen transformada linealmente a un nuevo dominio (por ejemplo, [80, 20]). Es más, algunos autores han implementado métodos prácticos basados en AS para procesamiento de imágenes, con resultados muy positivos (por ejemplo, [21, 22]). Nosotros, como se verá en el capítulo siguiente, hemos experimentado, en general, un mejor rendimiento de AS frente a SS para restauración de imágenes, lo que está de acuerdo con [126].

En este capítulo mostramos cómo se pueden aplicar los métodos presentados en los capítulos anteriores a la restauración de degradaciones deterministas *a posteriori*. Hemos observado, para cada método, qué tipo de raleza es mejor favorecer en la práctica. Como consecuencia, utilizamos el método ℓ_p-AP para restauración con SS y el método ℓ_p-GM para el uso de AS.

Empezaremos explicando y formulando el conjunto de consistencia con la observación para una degradación cualquiera (Sección 12.1). Luego formularemos el problema de restauración basado en raleza de síntesis (Sección 12.2). A continuación mostraremos cómo adaptar el método ℓ_p-AP para resolverlo (Sección 12.3). Después formularemos el problema de restauración basado en raleza de análisis (Sección 12.4), y cómo adaptar el método ℓ_p-GM para resolverlo (Sección 12.5).

5.1. Consistencia con la observación

Tenemos una imagen degradada $\mathbf{y} \in \mathbb{R}^N$. Consideramos que la degradación consiste en la pérdida de información que puede identificarse (como bits, píxeles, componentes cromáticas, etc.). Asumimos que podemos conocer con exactitud, dado $\mathbf{y}$, qué elementos se conservan de la imagen original (degradaciones deterministas *a posteriori*). De esta forma, podemos

replicar la degradación asociada, notada como:

$$\mathbf{y} = f_{\mathbf{y}}(\mathbf{x}). \tag{5.1}$$

Definimos el conjunto de consistencia para la observación $\mathbf{y}$, $R(\mathbf{y})$, como aquellas imágenes que, tras ser degradadas a través de la función $f_{\mathbf{y}}(\mathbf{x})$, resultan en la misma observación. Matemáticamente:

$$R(\mathbf{y}) = \{\mathbf{x} \in R^N : \ f_{\mathbf{y}}(\mathbf{x}) = \mathbf{y}\}.$$

5.2. Formulación usando raleza en sentido de síntesis

Si tratamos con representaciones redundantes, tenemos que:

$$\mathbf{y} = f_{\mathbf{y}}(\mathbf{\Phi a}),$$

donde $\mathbf{a}$ es un vector de síntesis cuya reconstrucción resulta en la imagen original. La estimación mediante *Máximo A Posteriori* de $\mathbf{a}$ es:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\log p(\mathbf{a}) + \lambda\|\mathbf{y} - f_{\mathbf{y}}(\mathbf{\Phi a})\|_2^2\}.$$

donde $p(\mathbf{a})$ es un modelo *a priori* para los coeficientes de la representación. Como es común en la literatura cuando, como en nuestro caso, la representación es de tipo ondícula (por ejemplo, [3, 80, 84, 31]), asumimos coeficientes independientes y distribuciones con colas que decaen bruscamente, tales como la densidad gaussiana generalizada (*Generalized Gaussian density*, en inglés):

$$p(\mathbf{a}) \propto \exp\{-k\|\mathbf{a}\|_p^p\}.$$

Cuando $0 \leq p \leq 1$, esta distribución es *rala*, en el sentido de tener una función densidad de probabilidad que concentra la mayoría de los coeficientes alrededor de cero, de forma que sólo una pequeña proporción de ellos tienen amplitudes relativamente amplias. El logaritmo de este modelo es proporcional a la p-ésima potencia de la norma ℓ_p del vector, más alguna constante irrelevante ($\log p(\mathbf{a}) \propto \|\mathbf{a}\|_p^p + A$). Entonces, nuestro problema de optimización se plantea como sigue:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_p^p + \lambda\|\mathbf{y} - f_{\mathbf{y}}(\mathbf{\Phi a})\|_2^2\},$$

Nótese que esto es análogo al problema de minimización de una norma ℓ_p de la Ecuación (3.1), pero la fidelidad se mide en términos del residuo

entre la observación y la degradación de la estimación. Así, la función de coste que se minimiza está formada por la suma de dos términos, uno que corresponde a la raleza de la aproximación, y el otro a la distancia cuadrática al conjunto de consistencia $R(\mathbf{y})$. El parámetro λ controla la importancia relativa de cada término en la solución final. En la práctica, requerimos que la estimación, al igual que la imagen original, pertenezca a $R(\mathbf{y})$. Equivalentemente, requerimos que $\hat{\mathbf{a}}$ pertenezca al conjunto $S(\mathbf{y})$ de vectores de síntesis que representen imágenes del conjunto $R(\mathbf{y})$:

$$S(\mathbf{y}) = \{\mathbf{a} \in \mathbb{R}^M : \ \mathbf{\Phi a} \in R(\mathbf{y})\}.$$

Entonces, establecemos λ a infinito, lo que nos lleva al siguiente problema:

$$\begin{aligned} \hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} \|\mathbf{a}\|_p^p \text{ s.t. } \mathbf{a} \in S(\mathbf{y}), \\ \hat{\mathbf{x}} &= \mathbf{\Phi}\hat{\mathbf{a}}. \end{aligned} \tag{5.2}$$

5.3. Estimación usando ℓ_p-AP y raleza en sentido de síntesis

La solución a la Ecuación (12.2) tiene una cierta norma ℓ_p, $\|\hat{\mathbf{a}}\|_p^p = R^*$. Este valor puede encontrarse resolviendo el problema:

$$R^* = \text{mín}\{R \in \mathbb{R}^* : \ B_p(R) \bigcap S(\mathbf{y}) \neq \emptyset\}.$$

La intersección entre los conjuntos correspondientes, $B_p(R^*)$ y $S(\mathbf{y})$, tendrá, en general, más de un elemento. Entre ellos, elegimos el más cercano a la observación:

$$\hat{\mathbf{a}} = P^{\perp}_{S(\mathbf{y}) \bigcap B_p(\hat{R}^*)}(\mathbf{\Phi}^T\mathbf{y}).$$

Es fácil ver que podemos usar ℓ_p-AP (ver el Capítulo 3) para resolver este problema, con sólo sustituir el conjunto $S(\mathbf{\Phi}, \mathbf{x})$ con el conjunto $S(\mathbf{y})$. Entonces, obtenemos las siguientes iteraciones:

$$\begin{aligned} \hat{\mathbf{a}}^{(0)} &= P^{\perp}_{B_p(\hat{R}^*)}(\mathbf{a}^{LS}), \\ \hat{\mathbf{a}}^{(k+1)} &= P^{\perp}_{B_p(\hat{R}^*)}(P^{\perp}_{S(\mathbf{y})}(\hat{\mathbf{a}}^{(k)})). \end{aligned} \tag{5.3}$$

Estas iteraciones terminan $\|\hat{\mathbf{a}}^{(k+1)} - \hat{\mathbf{a}}^{(k)}\|_2 < \delta$, para $\delta > 0$. La prueba de que el punto fijo de estas iteraciones es un mínimo local de la distancia a $S(\mathbf{y})$ es completamente análoga a la mostrada en la Sección 3.1 para el caso de aproximación rala.

Derivamos ahora la expresión de la proyección de un vector $\mathbf{b}^o$ sobre $S(\mathbf{y})$:

$$\hat{\mathbf{b}}^p_{S(\mathbf{y})} = P^{\perp}_{S(\mathbf{y})}(\mathbf{b}^o) = \arg\min_{\mathbf{b}}\{\|\mathbf{b}-\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{b}\in S(\mathbf{y})\}. \tag{5.4}$$

$S(\mathbf{y})$ es ortogonal al conjunto A de respuestas lineales a imágenes, definido como :

$$A = \{\mathbf{b} :\ \exists \mathbf{x}\in\mathbb{R}^N,\ \mathbf{\Phi}^T\mathbf{x} = \mathbf{b}\}.$$

Entonces, análogamente al caso del espacio afín de reconstrucción perfecta de una imagen (ver Ecuación (3.4)), tenemos que:

$$\hat{\mathbf{b}}^p_{S(\mathbf{y})} = \mathbf{b}^o + \mathbf{\Phi}^T\mathbf{\Phi}[\hat{\mathbf{b}}^p_{S(\mathbf{y})} - \mathbf{b}^o].$$

Ahora definimos $S_A(\mathbf{y})$ como el conjunto de respuestas lineales cuya reconstrucción es consistente con la observación:

$$S_A(\mathbf{y}) = \{\mathbf{b}\in R^M :\ \exists \mathbf{x}\in R(\mathbf{y}),\ \mathbf{\Phi}^T\mathbf{x} = \mathbf{b}\}.$$

Tenemos que $\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = P^{\perp}_{S_A(\mathbf{y})}(\hat{\mathbf{b}}^p_{S(\mathbf{y})}) = \mathbf{\Phi}^T\mathbf{\Phi}\hat{\mathbf{b}}^p_{S(\mathbf{y})}$, y entonces:

$$\hat{\mathbf{b}}^p_{S(\mathbf{y})} = \mathbf{b}^o + \hat{\mathbf{b}}^p_{S_A(\mathbf{y})} - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o. \tag{5.5}$$

Para resolver $\hat{\mathbf{b}}^p_{S_A(\mathbf{y})}$, debemos derivar una expresión de la proyección ortogonal sobre $S_A(\mathbf{y})$ en términos de nuestra observación $\mathbf{b}^o$. Tenemos que:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \arg\min_{\mathbf{b}}\{\|\mathbf{b}^o-\mathbf{b}\|_2^2 \text{ s.t. } \mathbf{b}\in S_A(\mathbf{y})\},$$

y podemos expresar:

$$\mathbf{b}^o - \mathbf{b} = (\mathbf{b}^o - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o) + (\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o - \mathbf{b}).$$

Estas dos diferencias entre paréntesis son vectores ortogonales, pues el primero pertenece al espacio nulo de $\mathbf{\Phi}$, mientras que el segundo no tiene componente nula en $\mathbf{\Phi}$ (es decir, $\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b} = \mathbf{b}$, porque $\mathbf{b}\in S_A(\mathbf{y})$). Entonces, podemos escribir:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \arg\min_{\mathbf{b}}\{\|\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o-\mathbf{b}^o\|_2^2 + \|\mathbf{b}-\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{b}\in S_A(\mathbf{y})\}.$$

Como el primer término de la suma es independiente de $\mathbf{b}$, puede ignorarse en la minimización, resultando en:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \arg\min_{\mathbf{b}}\{\|\mathbf{b}-\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{b}\in S_A(\mathbf{y})\}.$$

Sabemos que, para cada vector $\mathbf{b} \in S_A(\mathbf{y})$, existe otro vector $\mathbf{x} \in R(\mathbf{y})$ tal que $\mathbf{b} = \mathbf{\Phi}^T\mathbf{x}$. Así, sustituyendo en la expresión previa, llegamos a:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \mathbf{\Phi}^T[\arg\min_{\mathbf{x}}\{\|\mathbf{\Phi}^T\mathbf{x} - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{x} \in R(\mathbf{y})\}].$$

Y dado que $\mathbf{\Phi}^T$ es un marco de Parseval:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \mathbf{\Phi}^T[\arg\min_{\mathbf{x}}\{\|\mathbf{x} - \mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{x} \in R(\mathbf{y})\}].$$

La minimización en $\mathbf{x}$ corresponde a la proyección ortogonal de $\mathbf{\Phi}\mathbf{b}^o$ sobre el conjunto de imágenes consistentes con la observación, $P^{\perp}_{R(\mathbf{y})}(\mathbf{\Phi}\mathbf{b}^o)$, por lo que obtenemos:

$$P^{\perp}_{S_A(\mathbf{y})}(\mathbf{b}^o) = \mathbf{\Phi}^T P^{\perp}_{R(\mathbf{y})}(\mathbf{\Phi}\mathbf{b}^o). \tag{5.6}$$

Y, sustituyendo en la Ecuación (12.5), tenemos finalmente que:

$$P^{\perp}_{S(\mathbf{y})}(\mathbf{b}^o) = \mathbf{b}^o + \mathbf{\Phi}^T(P^{\perp}_{R(\mathbf{y})}(\mathbf{\Phi}\mathbf{b}^o) - \mathbf{\Phi}\mathbf{b}^o).$$

Encontrar la proyección ortogonal sobre el conjunto de consistencia es trivial para una numerosa clase de degradaciones estrictamente reproducibles *a posteriori*, simplemente forzando a la observación a preservar los componentes deseados. Por supuesto, la forma concreta de esta proyección depende de cada degradación. Cuando tratemos cada aplicación por separado formularemos más concretamente cada una de ellas (Capítulo 6).

5.4. Formulación usando raleza en sentido de análisis

Cuando hablamos de respuestas lineales ralas, la raleza estricta, en general, no se puede alcanzar, porque no podemos evitar que varios coeficientes de análisis respondan de manera simultánea a la misma característica de la imagen. En vez de eso, podemos considerar que las respuestas típicas a las imágenes naturales concentran, en una pequeña proporción de coeficientes, una gran proporción de la energía total de la señal, y, así, podemos modelar la representación lineal como un vector estrictamente ralo $\mathbf{a}$, que representa a las respuestas más altas en amplitud, más un término de corrección gaussiano, $\mathbf{r}$. Entonces, si definimos $S_A(\mathbf{y})$ como el conjunto de respuestas lineales cuya reconstrucción es consistente con la observación, esto es:

$$S_A(\mathbf{y}) = \{\mathbf{b} \in R^M : \exists \mathbf{x} \in R(\mathbf{y}),\ \mathbf{\Phi}^T\mathbf{x} = \mathbf{b}\},$$

entonces podemos escribir nuestro problema de optimización como:

$$\begin{aligned}(\hat{\mathbf{a}}, \hat{\mathbf{r}}) &= \arg\min_{\mathbf{a},\mathbf{r}}\{\|\mathbf{a}\|_p^p + \lambda\|\mathbf{r}\|_2^2 \text{ s.a. } (\mathbf{a}+\mathbf{r}) \in S_A(\mathbf{y})\}, \\ \hat{\mathbf{x}} &= \mathbf{\Phi}(\hat{\mathbf{a}} + \hat{\mathbf{r}}).\end{aligned} \tag{5.7}$$

5.5. Estimación usando ℓ_p-GM y raleza en sentido de análisis

Siguiendo un camino completamente paralelo al utilizado para el problema de aproximación rala, a la hora de resolver el problema de la Ecuación (12.7) derivamos una expresión que depende sólo de un vector $\mathbf{b} = \mathbf{a} + \mathbf{r}$, y donde el conjunto de la restricción, $S(\mathbf{\Phi}, \mathbf{x})$, se sustituye por la nueva restricción, $S_A(\mathbf{y})$. Hay que remarcar que el nuevo conjunto ya no es afín, en general, y que, entonces, tenemos que considerar su curvatura proyectando el gradiente de la función de coste sobre su hiperplano tangente en cada punto frontera $\mathbf{b}$. Esta proyección puede calcularse como el límite:

$$\nabla^{S_A(\mathbf{y})} C_p(\mathbf{b}, \theta) = \lim_{\alpha \to 0} \frac{P^{\perp}_{S_A(\mathbf{y})}(\alpha \nabla C_p(\mathbf{b}, \theta))}{\alpha},$$

donde $P^{\perp}_{S_A(\mathbf{y})}$ es la proyección ortogonal sobre $S_A(\mathbf{y})$ (Ecuación (12.6)), y donde $C_p(\mathbf{b}, \theta)$ indica, con $p = 0$ o $p = 1$, las funciones de coste definidas en las Ecuaciones (4.5) y (4.11), respectivamente. El método de descenso de gradiente queda entonces formulado como:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \alpha \nabla^{S_A(\mathbf{y})} C_p(\mathbf{b}^{(k)}, \theta). \tag{5.8}$$

Sin embargo, en la práctica es más conveniente el uso de los siguientes cálculos en el bucle de estimación, que son más sencillos:

$$\mathbf{b}^{(k+1)} = P^{\perp}_{S_A(\mathbf{y})} \left(\mathbf{b}^{(k)} - \alpha \nabla C_p(\mathbf{b}^{(k)}, \theta) \right),$$

lo que asegura que el vector actualizado pertenece al conjunto de consistencia $S_A(\mathbf{y})$ para cualquier valor de α. Esta regla de actualización es equivalente a la que aparece en la Ecuación (12.8) si la proyección es lineal.

A causa de la estructura similar del problema de minimización descrito en la Ecuación (4.2) y del descrito en la Ecuación (12.7), se puede aplicar la misma estrategia para buscar una minimización global de la función de coste cuando $p = 0$ o $p = 1$. Esto implica, en la práctica, que se consigue un rendimiento de restauración significativamente mejor si usamos un umbral

que decae de forma exponencial hasta alcanzar el valor deseado, y entonces usamos ese umbral fijo hasta converger. Hemos visto empíricamente en varias aplicaciones que el umbral final óptimo en nuestra optimización está normalmente cercano a cero, como también describen [21, 22]. Así, en ausencia de una fuente de información adicional, un valor arbitrariamente pequeño del umbral[1] es válido como criterio de parada para las iteraciones.

[1]Umbrales demasiado bajos requieren más computación, por lo que tenemos de nuevo un compromiso entre calidad y tiempo de computación.

Capítulo 6

Varias aplicaciones

En este capítulo se presentan varias aplicaciones de los métodos presentados en esta Tesis a problemas concretos de restauración de imágenes bajo degradaciones deterministas *a posteriori* (ver Capítulo 12). Motivaremos cada uno de los problemas y formularemos el conjunto de consistencia asociado a cada una de ellas y la proyección ortogonal sobre él. Por último, presentaremos experimentos que demuestran que los métodos presentados son muy competitivos para las aplicaciones analizadas.

En la Sección 6.1 se aplican nuestros métodos de aproximación rala a la eliminación de artefactos de cuantificación espacial. En la Sección 6.2 se aplican a estimación de zonas perdidas de la imagen. En la Sección 6.3 estudiamos el problema de interpolación de mosaicos de Bayer de matrices de filtros de color (*Color Filter Arrays*, en inglés). Por último, en la Sección 6.4 estudiamos el problema del incremento de detalle en imágenes.

6.1. Eliminación de artefactos de cuantificación

6.1.1. Introducción

La cuantificación espacial es parte indispensable de la captura de imágenes con dispositivos digitales. Normalmente los artefactos que se derivan de ella, como falsos contornos y supresión de la textura de bajo contraste, están cerca o incluso por debajo del umbral de visibilidad. Sin embargo, en numerosas situaciones pueden resultar evidentes. Por ejemplo cuando se amplia el rango de luminancia local de una imagen para inspeccionar detalles de bajo contraste, o cuando se invierte la convolución de imágenes borrosas cuantificadas, sobre todo cuando hay poco ruido derivado de otras fuentes de error. También puede resultar útil como paso previo a la extracción de características locales sensibles, como el gradiente

de la luminancia. Otras posibles aplicaciones pasan por interpolar curvas de nivel en mapas topográficos o barométricos, o por usar un número reducido de bits por píxel para transmisión cuando no hay suficientes recursos para llevar a cabo una compresión más avanzada de la imagen.

Sorprendentemente, hasta hace poco tiempo la eliminación de artefactos de cuantificación en el dominio de la imagen (de ahora en adelante, des-cuantificación) ha recibido poca atención en la literatura científica. En contraste, la cuantificación en el dominio transformado ha sido tratada ampliamente, especialmente en el contexto de post-procesamiento de imágenes comprimidas (por ejemplo, ver [127, 128, 129, 130, 131]).

Sin embargo, en los últimos años se ha notado un creciente interés en la aproximación al problema en el dominio de la imagen. El primer trabajo del que tenemos noticia fue [132], que utilizaba la des-cuantificación como paso previo a la detección de bordes. Recientemente, se han publicado otros métodos basados en iteraciones de filtrado y corrección de la diferencia con la original [133, 134]. Pero este tipo de estrategias, aunque resultan en algoritmos eficientes, son demasiado sencillas para ofrecer resultados satisfactorios. En paralelo a estos trabajos, nosotros presentamos un método basado en favorecer la raleza en la representación lineal con ondículas redundantes [26]. La selección de coeficientes se realiza, en este método, directamente desde el vector de análisis, por lo que encuadramos esta técnica dentro de los heurísticos voraces. Este tres últimos métodos serán descritos en esta sección.

En esta sección comparamos el rendimiento de ℓ_p-AP adaptado para des-cuantificación en sus dos versiones desarrolladas ($p = 1$ y $p = 0$) y con respecto los métodos citados [26, 133, 134]. Veremos, a través de ejemplos de aplicación, que ℓ_0-AP ofrece un rendimiento significativamente superior al resto de métodos, y también a ℓ_1-AP.

6.1.2. Conjunto de consistencia

En este caso, el conjunto de consistencia está formado por aquellas imágenes que, al usar los mismos niveles de cuantificación, dan todas la misma observación. Así, siendo $\mathbf{y}$ una imagen cuantificada observada, el conjunto de consistencia asociado a $\mathbf{y}$, notado como $R_Q(\mathbf{y})$, se define como:

$$R_Q(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \; y_i - \frac{\delta_i}{2} < x_i \leq y_i + \frac{\delta_i}{2}, \; \forall i \in \{1, ..., N\}\},$$

donde δ_i indica el tamaño del intervalo de cuantificación asociado al píxel[1] y_i.

[1]Hemos asumido aquí cuantificación uniforme, por sencillez, pero todos los métodos descritos son fácilmente aplicables a otros tipos de cuantificación.

Dada una imagen $\mathbf{x} \in \mathbb{R}^N$, la proyección ortogonal sobre $R_Q(\mathbf{y})$ se calcula fácilmente como $\mathbf{z} = P^{\perp}_{R_Q(\mathbf{y})}(\mathbf{x})$, donde:

$$z_i = \begin{cases} x_i, & y_i - \frac{\delta_i}{2} < x_i \leq y_i + \frac{\delta_i}{2} \\ y_i - \frac{\delta_i}{2} + \epsilon, & x_i \leq y_i - \frac{\delta_i}{2} \\ y_i + \frac{\delta_i}{2}, & y_i + \frac{\delta_i}{2} < x_i, \end{cases}$$

donde $\epsilon \in \mathbb{R}^*$ (idealmente infinitesimal) es un artificio añadido para conseguir intersección vacía entre intervalos de cuantificación cerrados adyacentes.

6.1.3. Implementación

En los experimentos mostrados en el siguiente apartado hemos comparado nuestros métodos con tres métodos recientes aplicables a eliminación de artefactos de cuantificación. Describimos a continuación brevemente estos tres métodos y nuestra implementación de los mismos. Los valores que hemos dado a los diferentes parámetros que aparecen han sido optimizados a mano para que cada método pare las iteraciones en un nivel similar de aproximación a la convergencia final.

Umbralización directa y optimización. En [26] describimos un método para eliminar artefactos de cuantificación espacial, que fuerza un alto grado de raleza en la representación de la estimada con un diccionario redundante basado en ondículas. Para este propósito, diseñamos un operador lineal que devuelve la imagen de mínima norma ℓ_2 que preserva un conjunto de coeficientes significativos, y estimamos la imagen original minimizando la cardinalidad de ese subconjunto, siempre asegurando que el resultado sea compatible con la observación cuantificada. Implementamos esta solución mediante proyecciones alternas sobre conjuntos convexos. Para seleccionar el conjunto de coeficientes significativos, umbralizamos directamente las amplitudes de la representación lineal de la imagen, usando un umbral proporcional a la energía estimada de cada sub-banda original.

Esta aplicación está basada en la Umbralización Directa y Optimización (DT+OP) vista en el Capítulo 3, y por extensión este será el nombre que le demos aquí. Sus detalles pueden consultarse en [26]. En los experimentos que mostramos en el capítulo siguiente hemos utilizado DT-CWT con 7 escalas y asumimos que hay intersección entre los conjuntos cuando, en menos de 30 iteraciones, el MSE del vector proyectado sobre un conjunto con respecto al proyectado sobre el otro es menor o igual a 0,5.

Difusión restringida. En [134] se presenta el método de Difusión Restringida (CD, *Constrained Diffusion* en inglés), basada en combinar filtrados lineales con corrección no-lineal de la diferencia con la observación.

Nuestra implementación del método sigue los pasos explicados en [134] y resulta en un método iterativo que inicializa la estimación con la observada y consta de los dos siguientes pasos: 1) Convolución de la imagen estimada con la siguiente matriz:

$$\begin{pmatrix} 0 & \frac{1}{5} & 0 \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ 0 & \frac{1}{5} & 0 \end{pmatrix}, \tag{6.1}$$

y 2) Proyección sobre el conjunto de consistencia con la observación. Las iteraciones paran cuando el MSE de la estimación en una iteración con respecto a la anterior es menor que 0,5.

Restauración Iterativa Restringida y Regularizada. En [133] se describe un método para eliminar artefactos de cuantificación, llamado Restauración Iterativa Restringida y Regularizada (RIRR). En el citado trabajo se describe su aplicación para cuantificaciones vectoriales. Sin embargo, cuando tratamos con cuantificación uniforme, como es nuestro caso, el método se reduce a una estrategia sencilla basada en minimizar la salida de un filtro paso alto, forzando en cada paso a la estimación a pertenecer al conjunto de consistencia.

Nuestra implementación resulta en un método iterativo, que se inicializa con la observación como primera estimación y consta de los dos siguientes pasos: 1) Restar a la estimación su propia convolución con un filtro espacial laplaciano. 2) Proyectar sobre el conjunto de consistencia con la observación. Las iteraciones paran cuando el MSE de la estimación resultante en una iteración con respecto a la anterior es menor que 0,5.

ℓ_p-AP. Utilizamos una búsqueda binaria para encontrar el radio de la menor bola ℓ_p que tenga intersección no-vacía con el conjunto de vectores consistentes con la observación (ver Capítulo 12), cuyo intervalo se inicializa entre 0 y M para ℓ_0-AP y entre 0 y $\|\mathbf{\Phi}^T\mathbf{y}\|_1$ para ℓ_1-AP. Consideramos encontrado el radio buscado cuando el intervalo de búsqueda tiene un longitud menor o igual a 5000, y asumimos que existe intersección cuando en menos de 30 iteraciones el MSE entre el vector proyectado en un conjunto y en el otro es menor o igual a[2] 0,3.

De entre las representaciones comparadas, hemos visto que el mejor rendimiento lo ofrece DT-CWT con 8 escalas.

[2]La mayor tolerancia dada a los métodos basados en filtrado lineal les favorece, porque realizar más iteraciones les llevaría a suavizar demasiado la estimación y perder, en consecuencia, PSNR.

6.1.4. Resultados y discusión

6.1.4.1. Comparación entre ℓ_0-AP y ℓ_1-AP

La Figura 6.1 muestra un ejemplo de comparación entre el rendimiento de ℓ_0-AP y ℓ_1-AP a la hora de eliminar artefactos de cuantificación espacial, usando la cuantificación con 3 bits de la imagen *Einstein*. Se muestran los resultados usando tanto DT-CWT con 8 escalas como Curvelets con 6. Se observa que el rendimiento de ℓ_0-AP es mucho mejor, tanto en incremento de PSNR como visualmente, ya que consigue eliminar las discontinuidades debidas a la cuantificación pero manteniendo una alta definición en los bordes originales de la imagen. También observamos que los resultados usando Curvelets son ligeramente mejores en este caso, para ambos métodos, que los obtenidos usando DT-CWT.

En la Figura 6.2 se muestra un ejemplo similar para una imagen con menos textura (*Peppers*). Vemos que el comportamiento relativo entre los métodos es parecido en un análisis cualitativo, aunque la diferencia en calidad visual entre ℓ_1-AP y ℓ_0-AP no es tan grande en este caso como en el ejemplo de *Einstein*. Esto se debe a que la ausencia de textura hace que las estimaciones sobre-suavizadas no tengan un fuerte impacto visual. Sin embargo, la diferencia en PSNR sigue siendo favorable a ℓ_0-AP. Se puede ver también que ahora el rendimiento es algo mejor usando DT-CWT que Curvelets.

En cuanto al comportamiento de ℓ_p-GM, hemos experimentado que es muy poco satisfactorio tanto para $p = 0$ como para $p = 1$. Así, usando raleza de análisis no logra eliminar los artefactos de cuantificación; y usando raleza de síntesis obtiene estimaciones demasiado suaves. Nótese que, en este último caso (SS), la capacidad de compactar la energía de $\ell_0 - GM$ es mucho mayor que la de ℓ_0-AP, pero esto no se refleja de forma paralela en el rendimiento del método.

6.1.4.2. Comparación entre ℓ_0-AP y los métodos existentes

En este apartado comparamos el rendimiento de ℓ_0-AP con los métodos descritos previamente: RRIR [133], CD [134], y DT+OP [26]. En la Tabla 6.1 se muestra el promedio (se ha usado el MSE para promediar) del rendimiento de cada método en las imágenes de nuestro conjunto de prueba (Apéndice A) para el rango de posibles niveles de cuantificación en imágenes de 8 bits. Podemos ver que los métodos basados en imponer raleza, DT+OP y ℓ_0-AP, mejoran sensiblemente el rendimiento de los métodos basados en operaciones lineales más sencillas, salvo cuando la imagen se ha cuantificado con un sólo bit. En concreto, ℓ_0-AP es mejor para todos aquellos niveles que, en la

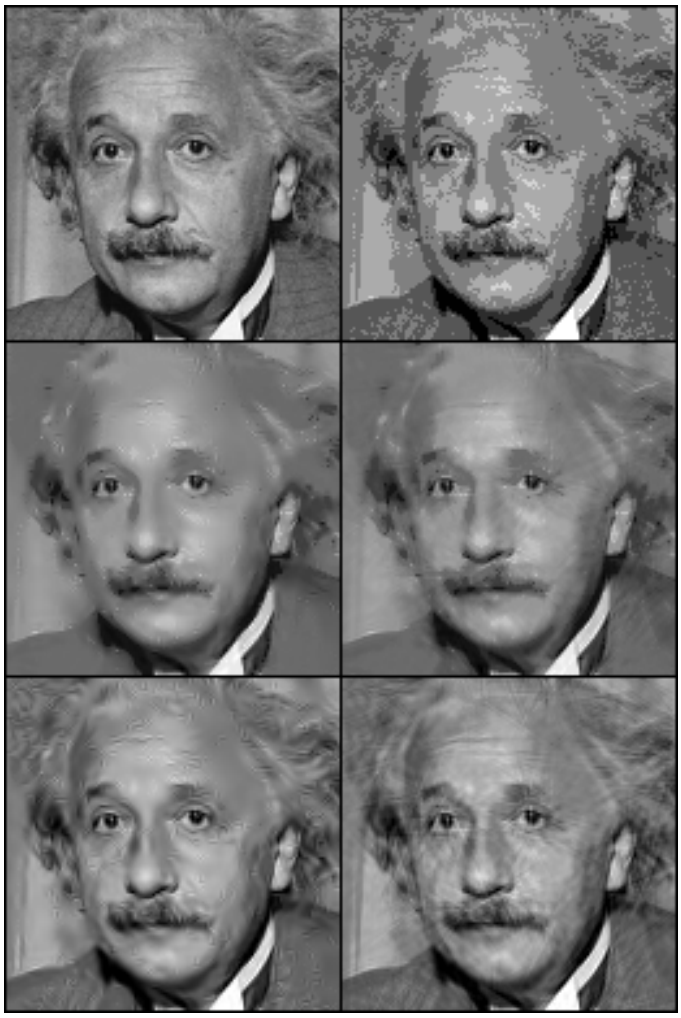

Figura 6.1: *Ejemplo de aplicación de ℓ_1-AP y ℓ_0-AP a la eliminación de artefactos de cuantificación espacial.* **Arriba - izquierda**, *imagen* Einstein *original, recortada a* 128×128 *píxeles.* **Arriba - derecha**, *cuantificación con* 3 *bits observada (PSNR:* 27,98 *dB).* **Centro - izquierda**, *resultado de ℓ_1-AP usando DT-CWT con* 8 *escalas (*30,17 *dB).* **Centro - derecha**, *resultado de ℓ_1-AP usando Curvelets* 6 *escalas (*30,61 *dB).* **Abajo - izquierda**, *resultado de ℓ_0-AP usando DT-CWT* 8 *escalas (*31,21 *dB).* **Abajo - derecha**, *resultado de ℓ_0-AP usando Curvelets* 6 *escalas (*31,38 *dB).*

práctica, resultan en artefactos visibles (rango bajo y medio).

Además, la apariencia visual de los resultados de los métodos basados en raleza es significativamente mejor que la de sus competidores, incluso para bajos números de bits de cuantificación. En las Figuras 6.3 y 6.4 puede verse una comparación visual de la aplicación de los métodos a las imágenes *Einstein* y *Peppers* cuantificadas con 3 bits. De todos ellos, el mejor, tanto

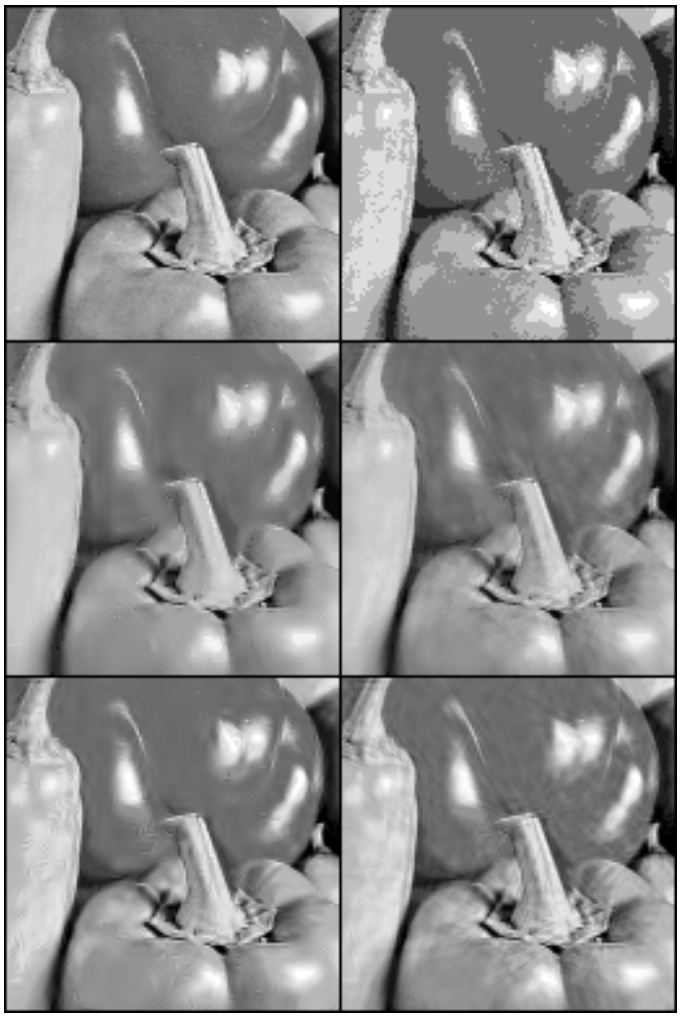

Figura 6.2: *Ejemplo de aplicación de ℓ_1-AP y ℓ_0-AP a la eliminación de artefactos de cuantificación espacial.* **Arriba - izquierda**, *imagen* Peppers *original, recortada a* 128×128 *píxeles.* **Arriba - derecha**, *cuantificación con* 3 *bits observada (PSNR:* 28,81 *dB).* **Centro - izquierda**, *resultado de ℓ_1-AP usando DT-CWT con* 8 *escalas (*29,08 *dB).* **Centro - derecha**, *resultado de ℓ_1-AP usando Curvelets* 6 *escalas (*29,50 *dB).* **Abajo - izquierda**, *resultado de ℓ_0-AP usando DT-CWT* 8 *escalas (*31,06 *dB).* **Abajo - derecha**, *resultado de ℓ_0-AP usando Curvelets* 6 *escalas (*30,85 *dB).*

visualmente como en PSNR, es ℓ_0-AP. Tanto RRIR como CD eliminan demasiadas componentes de alta frecuencia sin llegar a deshacerse del todo de los artefactos de cuantificación. El uso del umbral adaptativo permite a nuestro anterior método, DT+OP, eliminar los átomos seleccionados aisladamente que aparecen en el resultado de ℓ_0-AP. Sin embargo, la aparición de efecto *ringing* y el rendimiento pobre en zonas con predominio

	PSNR (dB)						
♯ Bits	1	2	3	4	5	6	7
Observada	16,40	22,73	29,22	34,71	40,59	46,45	51,11
RRIR	*17,62*	24,25	29,75	34,37	39,20	44,56	51,87
CD	**17,88**	*24,33*	29,60	34,38	39,30	44,65	*51,89*
DT+OP	16,46	24,05	*30,83*	*35,67*	**40,61**	**45,28**	49,27
ℓ_0-AP	17,29	**24,74**	**31,49**	**35,91**	*40,21*	*45,11*	**52,03**

Cuadro 6.1: *Promedio de PSNR obtenido usando las imágenes de nuestro conjunto de prueba, cuantificadas con todo el rango de posibles bits y restauradas usando los métodos RRIR, CD, DT+OP y ℓ_0-AP. La primera fila corresponde por la PSNR promediada de la imagen observada. En negrita se muestran los mejores resultados para cada número de bits, y en cursiva los segundos.*

de alta textura hacen que el efecto visual (además del rendimiento en PSNR) sea mejor en ℓ_0-AP. Para finalizar, mostramos en la última columna el resultado de ℓ_0-AP usando una representación conjunta DT-CWT - Curvelets (ver Apéndice E). Vemos que el incremento en la riqueza del diccionario, no sólo en número sino en el tipo de funciones elementales utilizadas, mejora significativamente el resultado, consiguiendo aumentar la PSNR y reduciendo drásticamente las funciones elementales aisladas que aparecen en la imagen estimada cuando sólo se usa una representación. Hemos escogido estos dos diccionarios para comparar con los resultados mostrados en 6.1 utilizando cada uno de ellos por separado. Sin embargo, usando otros diccionarios podemos mejorar aún más estos resultados. Por ejemplo, usando Curvelets y una versión de la Pirámide Orientable [135] sin residuo paso alto, la PSNR asciende a 31,99 dB para *Einstein* y 31,46 dB para *Peppers*.

En la tabla 6.1 podemos ver que, para altos números de bits de cuantificación, se produce una bajada en PSNR tras aplicar ℓ_0-AP. Sin embargo, en estos casos también se consigue quitar los artefactos de bajo contraste, mejorando la apariencia visual cuando realzamos el contraste de la imagen. La Figura 6.5 muestra un ejemplo. El panel de la izquierda es un detalle de 32×32 de una zona suave en una imagen fotográfica con 8 bits de resolución, con un factor de amplificación de contraste de aproximadamente 40 veces. El panel de la derecha es el mismo recorte en la imagen procesada con ℓ_0-AP. Se puede apreciar la apariencia más natural de esta imagen.

En cuanto al tiempo de computación de los métodos, la sencillez de RRIR y CD, que son métodos iterativos que sólo requieren una convolución y una proyección sobre el conjunto de consistencia por iteración, hace que

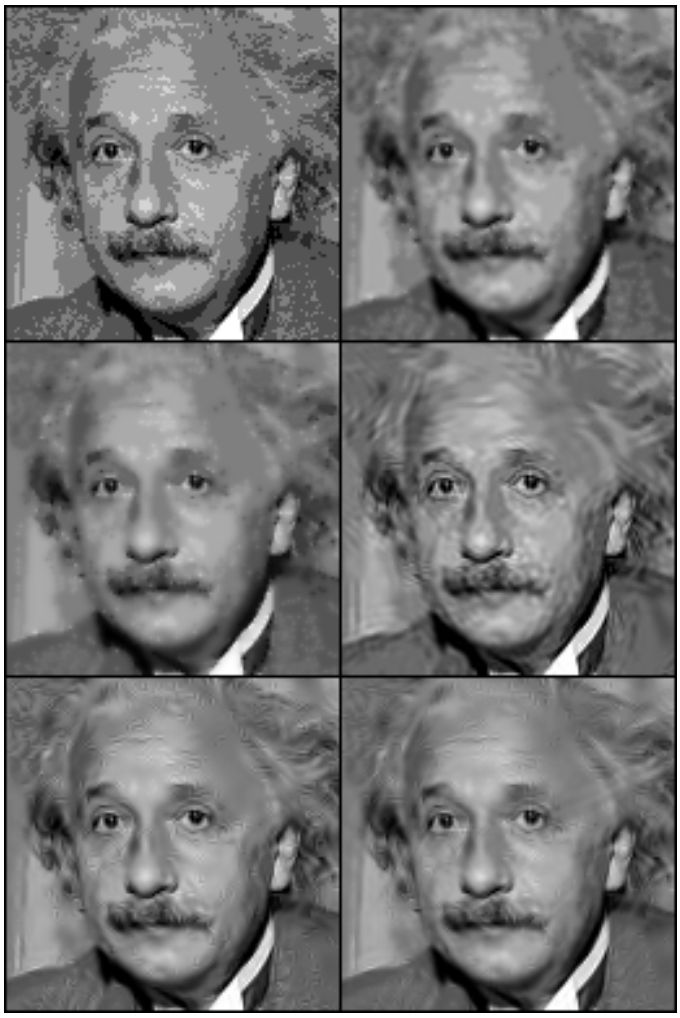

Figura 6.3: **Arriba - izquierda**, Einstein *cuantificada con* 3 *bits y recortada a* 128×128 *píxeles empezando en (*71, 41*), (*27,98 *dB). **Arriba - derecha**, *resultado de RRIR (*30,39 *dB). **Centro - izquierda**, *resultado de CD (*30,44 *dB). **Centro - derecha**, *resultado de DT+OP usando DT-CWT* 8 *escalas (*30,72 *dB). **Abajo - izquierda**, *resultado de* ℓ_0*-AP usando DT-CWT con* 8 *escalas (*31,21 *dB). **Abajo - derecha**, *resultado de* ℓ_0*-AP utilizando conjuntamente DT-CWT con* 8 *escalas y Curvelets* 6 *escalas, con igual factor de escala,* $\sqrt{\frac{1}{2}}$*, para ambos (*31,93 *dB).*

sean mucho más rápidos que los métodos basados en la imposición no lineal de raleza, ya que estos no sólo están dominados por el tiempo de análisis y síntesis de la representación asociada, sino que tienen que buscar la menor intersección entre los conjuntos involucrados en un bucle externo. Así, RRIR y CD realizan alrededor de 10 iteraciones para cada estimación, tardando apenas medio segundo en promedio, para imágenes de tamaño 256×256.

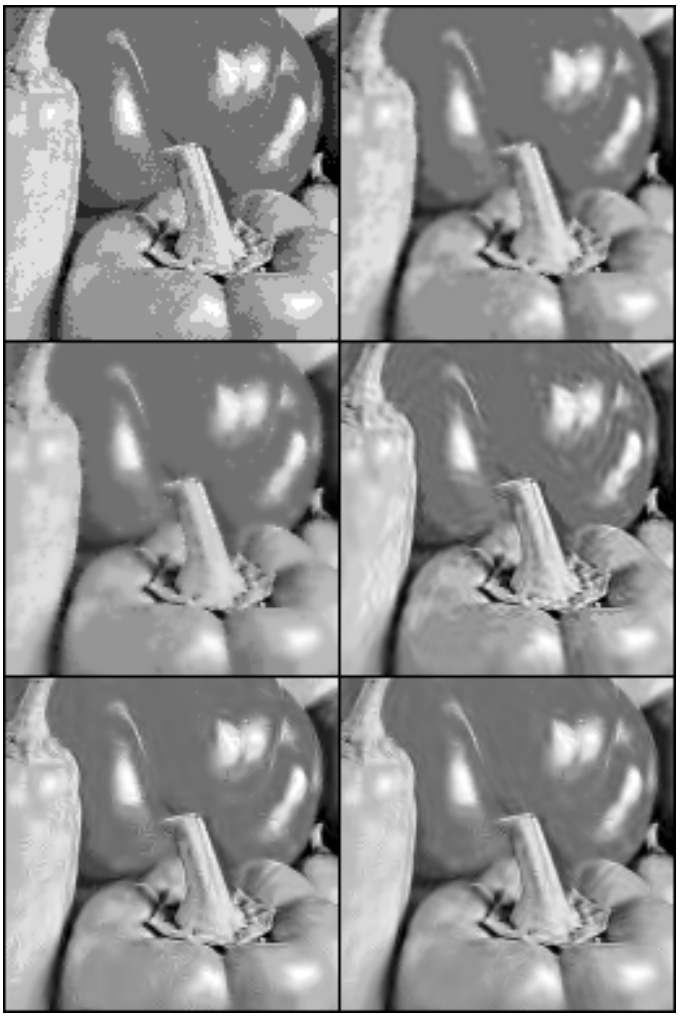

Figura 6.4: **Arriba - izquierda**, Peppers *cuantificada con* 3 *bits y recortada a* 128×128 *píxeles empezando en (*71, 41*), (*28,81 *dB). **Arriba - derecha**, resultado de RRIR (*29,65 *dB). **Centro - izquierda**, resultado de CD (*29,85 *dB). **Centro - derecha**, resultado de DT+OP usando DT-CWT* 8 *escalas (*30,38 *dB). **Abajo - izquierda**, resultado de* ℓ_0*-AP usando DT-CWT con* 8 *escalas (*31,07 *dB). **Abajo - derecha**, resultado de* ℓ_0*-AP utilizando conjuntamente DT-CWT con* 8 *escalas y Curvelets* 6 *escalas, con igual factor de escala,* $\sqrt{\frac{1}{2}}$*, para ambos (*31,46 *dB).*

Por otro lado, DT+OP tarda entre 15 segundos y 5 minutos usando DT-CWT con 8 escalas, variando según el número de bits de cuantificación de la imagen observada. Finalmente, ℓ_0-AP tarda 80 segundos en promedio usando DT-CWT con 8 escalas. Los tiempos de ejecución mostrados son de nuestras implementaciones MATLAB®, sobre un Intel®, $Core^{TM}2$ Duo a 1,66 GHz con 2 GB de RAM.

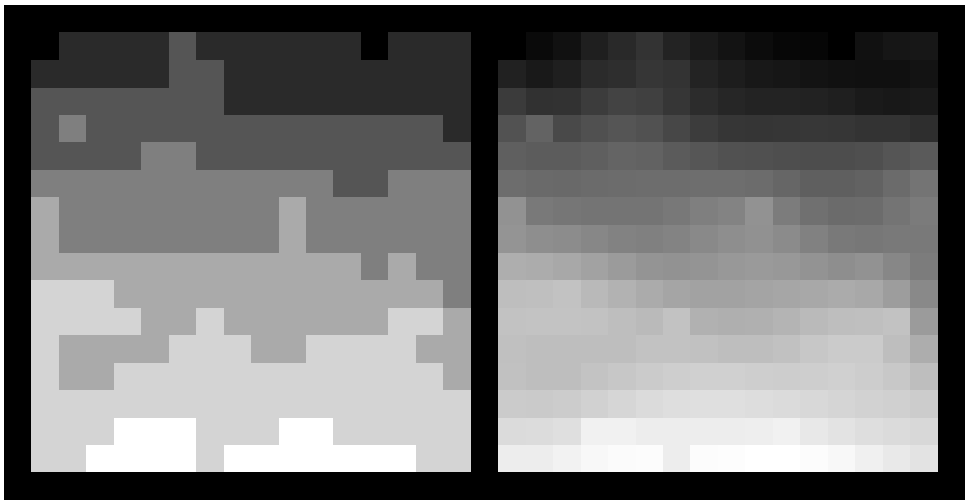

Figura 6.5: **Izquierda**, *detalle del cielo de una imagen fotográfica de* 8 *bits con contraste amplificado aproximadamente 40 veces.* **Derecha**, *mismo detalle tras procesar con* ℓ_0*-AP.*

6.1.5. Conclusiones

Hemos analizado el rendimiento del método ℓ_0-AP aplicado a la eliminación de artefactos de cuantificación espacial comparándolo con ℓ_1-AP y con otros métodos recientes publicados (RRIR, DT+OP CD). Podemos afirmar que la búsqueda de la imagen más rala dentro del conjunto de consistencia realizada mediante ℓ_0-AP ofrece resultados muy satisfactorios. En general, hemos visto que los métodos basados en aumentar la raleza son superiores a los métodos basados en operaciones lineales de filtrado de altas frecuencias. Sin embargo, ℓ_0-GM, que posee una mayor capacidad para compactar la energía en pocos coeficientes, consigue estimaciones que tienden a estar demasiado concentradas en bajas frecuencias. La discusión sobre por qué pasa esto se deja para un trabajo futuro.

6.2. Interpolación de regiones perdidas

6.2.1. Introducción

La pérdida de píxeles de la imagen es un problema asiduo tanto en la captura como en la transmisión de imágenes digitales. También es corriente querer hacer desaparecer detalles no deseados de una imagen (texto sobreimpreso, publicidad, un cable molesto sobre un bonito paisaje, etc.) o querer restaurar imágenes deterioradas por el tiempo.

En los últimos 30 años, o incluso más, se han propuesto muchas

técnicas diferentes para resolver el problema de recuperación de estos píxeles perdidos (usualmente referidas bajo el nombre de *in-painting*, en inglés). Por otro lado, los métodos de síntesis de textura también pueden usarse para rellenar regiones perdidas. Hay muchos trabajos que utilizan esta última estrategia, de entre los que podemos referirnos a [136, 137, 138, 36] entre otros. Desafortunadamente, la necesidad de indicarles manualmente desde qué zonas de la imagen deben extraer la información necesaria para la interpolación los convierte en métodos muy poco adecuados para la práctica. Las estrategias heurísticas más exitosas se basan en la combinación de propagación de bordes de la imagen (usando ecuaciones diferenciales parciales, PDE, de *partial differential equations*, en inglés) con síntesis local de texturas (por ejemplo, [139, 140, 141, 142]).

Existe un método rápido y sencillo de implementar que ofrece resultados comparables con estos métodos [143]. Está basado en la combinación iterativa de una operación lineal de filtrado y la restricción no lineal de preservar los píxeles observados.

También se han aplicado, recientemente, estrategias basadas en fomentar la raleza. Un buen ejemplo de este tipo de métodos es [22], que plantea una formulación basada en *Expectation-Maximization* para aproximar la solución más rala consistente con la observación a través de la minimización óptima de la norma ℓ_1.

En esta sección comparamos el rendimiento de nuestros métodos (ℓ_p-AP y ℓ_p-GM) adaptados para interpolación de píxeles o zonas perdidas de la imagen. Veremos, a través de ejemplos de aplicación, que ℓ_0-GM ofrece un excelente rendimiento, en términos de MSE. También compararemos con respecto a dos de las técnicas referidas ([22] y [143]).

6.2.2. Conjunto de consistencia

Cuando la degradación consiste en la pérdida de píxeles de la imagen, el conjunto de consistencia está compuesto por aquellas imágenes que, al sufrir la pérdida de los mismos píxeles, resulten en la misma observación. Así, dado un subconjunto, I, de índices fijos entre 1 y N, y dada una observación $\mathbf{y}$ que conserva los píxeles y_i de la imagen original para todo $i \in I$, definimos el conjunto de consistencia asociado a $\mathbf{y}$, $R_I(\mathbf{y})$, como:

$$R_I(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \ x_i = y_i, \ \forall i \in I\}.$$

Dada una imagen $\mathbf{x} \in \mathbb{R}^N$ y una matriz diagonal $\mathbf{D}$, de tamaño $N \times N$, donde cada elemento d_{ii} es 1 si $i \in I$ y 0 en caso contrario, la proyección ortogonal de un vector $\mathbf{x} \in \mathbb{R}^N$ sobre $R_I(\mathbf{y})$ es $P_{R_I(\mathbf{y})}(\mathbf{x}) = \mathbf{D}\mathbf{y} + (\mathbf{I} - \mathbf{D})\mathbf{x}$, donde $\mathbf{I}$ es la matriz identidad de tamaño $N \times N$.

6.2.3. ℓ_0-AP: nueva estrategia de búsqueda del radio

Hemos comprobado que ℓ_p-AP no ofrece buenos resultados de interpolación si buscamos la imagen más rala dentro del conjunto de consistencia. En la práctica, la calidad de la interpolación depende de encontrar un valor concreto de R que nos permita minimizar el error de nuestra estimación. Denominamos R_{opt} a este valor. Entonces, lo que buscamos es, dado un nivel R_{opt} de raleza, la proyección sobre el conjunto de consistencia de la imagen, con coeficientes dentro de la bola ℓ_p de radio R_{opt}, y cuyos píxeles en I estén más cerca, en sentido de MSE, a los píxeles observados en $\mathbf{y}$. Para encontrar este radio óptimo propusimos una solución basada en maximizar el valor cuadrático medio (MSV, *Mean Square Value* en inglés) de los píxeles interpolados [18]. Intuitivamente, vemos que para valores pequeños de R sólo se representarán las características más sobresalientes de la imagen observada, por lo que la estimación será muy suave y obtendremos necesariamente bajos valores de MSV en las zonas interpoladas. Cuando elegimos un radio R muy alto, los bordes rotos causados por las zonas perdidas se representarán mejor usando muchas funciones elementales que si las aproximásemos a más baja escala. Esto produce una interpolación pobre y, de nuevo, una amplitud baja en los píxeles estimados. Por último, si usamos valores intermedios para R, podemos esperar tener suficientes funciones como para representar todas las características relevantes de la imagen, pero no las suficientes como para describir los falsos bordes. Gracias a eso, los huecos de información perdida serán rellenados con las funciones del diccionario apropiadas, lo que causará un mayor MSV en las zonas interpoladas y por lo tanto una mejor interpolación.

La línea continua de la Figura 6.6, que corresponde con el eje izquierdo de la misma, muestra el MSE normalizado en los píxeles estimados para cada valor de R, donde R está normalizado por R_{opt}, que corresponde con el valor de R para el que se alcanza el mínimo de esta curva. La línea discontinua muestra el MSV normalizado, y se corresponde con el eje derecho. Denominamos R_{max} al valor de R donde esta curva alcanza su máximo. Las líneas punteadas indican la desviación típica de esta curva para cada valor de R. La línea formada por segmentos y puntos es el valor real de MSV de los píxeles perdidos en la imagen original, que es una cota superior para la curva de MSV. Todos estos valores están promediados en nuestro conjunto de prueba, con una máscara aleatoria donde aproximadamente el 40 % de los píxeles están perdidos. Para cada prueba se ejecutaron 250 iteraciones del método ℓ_0-AP. El método descrito propone estimar R_{opt} a partir del valor observado R_{max}. Así, para este porcentaje de píxeles perdidos, tenemos que $\hat{R}_{opt} = \frac{1}{0{,}7} R_{max}$.

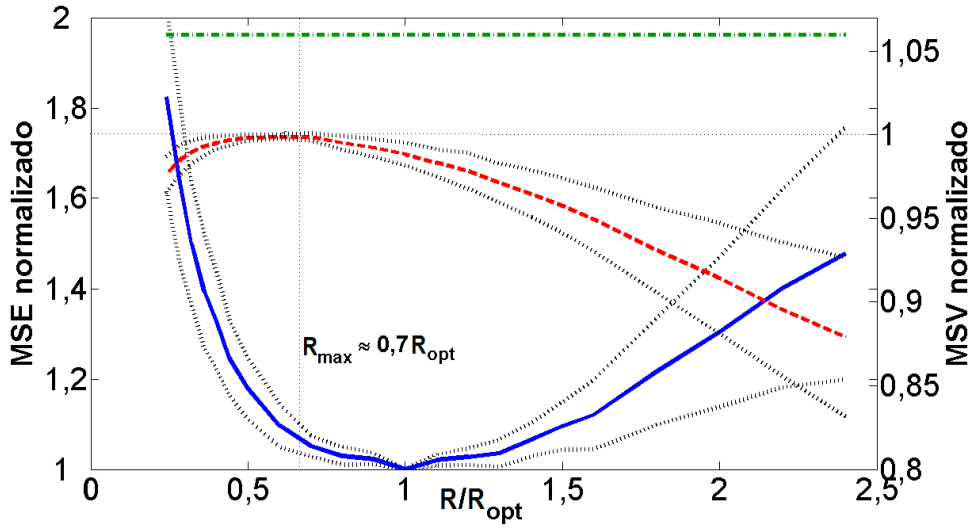

Figura 6.6: **Línea continua - eje izquierdo**, *error cuadrático medio normalizado de los píxeles estimados con respecto a valores normalizados al valor mínimo de esta curva en ordenadas, y al valor para el que ocurre este mínimo, en abscisas.* **Línea intermitente - eje derecho**, *valor cuadrático medio normalizado de los píxeles estimados.* **Línea punteada**, *desviación típica para cada valor del eje de abscisas.* **Línea mixta segmentos y puntos**, *valor cuadrático medio de los píxeles originales en las posiciones perdidas. Todas las curvas están promediados en nuestro conjunto de prueba, usando una máscara que elimina alrededor del* 40 % *de los píxeles.*

6.2.4. Implementación

Fast-inpainting. El método *Fast-inpainting* [143] consiste simplemente en aplicar iterativamente un filtrado lineal con la máscara de convolución bidimensional:

$$\begin{pmatrix} 0{,}073235 & 0{,}176765 & 0{,}073235 \\ 0{,}176765 & 0 & 0{,}176765 \\ 0{,}073235 & 0{,}176765 & 0{,}073235 \end{pmatrix} \tag{6.2}$$

y una proyección sobre el conjunto de consistencia para conservar los píxeles observados. En nuestra implementación, las iteraciones paran cuando el MSE de la estimación en una iteración con respecto a la anterior es menor que $1e-3$.

EM-inpainting. El método *EM-inpainting* [22] sigue la estrategia de la relajación convexa. Hemos utilizado la implementación disponible en el paquete MCALab 8.02 para MATLAB® que puede descargarse de [144]. Hemos utilizado los valores de los parámetros descritos en [22]. La representación utilizada es Curvelets con 6 escalas combinada con DCT local con tamaño de bloque 32×32. En [144] también puede encontrarse

las funciones de análisis y síntesis para esta última representación.

ℓ_p-AP y ℓ_p-GM. Para implementar ℓ_p-AP aplicado al problema que nos ocupa, primero estimamos el radio, para lo que se necesita varias interpolaciones, y luego ejecutamos el método otra vez para ese radio. Hemos realizado 100 iteraciones para cada interpolación. Para ℓ_p-GM hemos experimentado que el rendimiento no mejora significativamente cuando β es superior a 0,8. Además, en este caso no se consiguen diferencias significativas usando valores de α mayores que 1. Paramos las iteraciones cuando el umbral es inferior a 0,1.

Como ℓ_1-GM y *EM-inpainting* siguen una estrategia muy similar (favorecer la raleza a través de minimizar la norma ℓ_1, sólo presentamos los resultados de aquella implementación que nos da mejor rendimiento en cada caso.

6.2.5. Resultados y discusión

6.2.5.1. Pérdida aleatoria de píxeles

Consideraremos aquí que los píxeles se han perdido de forma independiente siguiendo una cierta distribución. La restauración de este tipo de degradación se conoce con el nombre de rellenado (*filling-in*, en inglés).

Comparamos aquí sistemáticamente el rendimiento de ℓ_0-GM (usando SA) con respecto a ℓ_1-GM (usando también SA) y *Fast-inpainting*. En las Tablas 6.2 y 6.3 se muestra el promedio del rendimiento de cada método en las imágenes de nuestro conjunto de prueba para un amplio rango del porcentaje de píxeles perdidos. Hemos incluido los resultados de ℓ_0-GM con dos representaciones diferentes: Curvelets con 6 escalas usado por sí sólo y combinado con DCT local, con tamaño de bloque 32×32. Hemos usado factores $\sqrt{0{,}5}$ para ambos diccionarios. Para ℓ_1-GM, sólo se muestra la representación combinada (que da mejores resultados). Vemos que, minimizando la norma ℓ_0, se obtienen los mejores resultados, excepto para porcentajes muy altos de píxeles perdidos. Volvemos a ver también que el uso de diccionario incoherentes combinados mejora la calidad de la estimación, gracias a la capacidad extra para representar características con pocos coeficientes.

En la Figura 6.7 podemos ver las imágenes correspondientes a los métodos comparados en el caso de pérdida aleatoria de $\approx 80\,\%$ de los píxeles de la imagen *Barbara*. Se observa que el método basado en minimizar la norma ℓ_1 (ℓ_1-GM) no consigue interpolar de una forma suficientemente suave los píxeles perdidos, de forma que se mantienen los artefactos en la imagen. Por su parte, la interpolación lineal no consigue recuperar la

textura. Sorprendentemente, ℓ_0-GM consigue recuperar bien, dado el alto porcentaje de píxeles perdidos en la degradación, tanto las zonas más suaves como las más ricas en textura. Se puede ver también que el uso de dos representaciones combinadas elimina los artefactos inherentes a la representación (átomos seleccionados de manera aislada).

Método	PSNR (dB)				
$\approx$ % píxeles perdidos	10	20	30	40	50
Observada	24,05	20,90	19,13	17,90	16,82
Fast-inpainting	37,72	34,26	32,23	30,67	29,08
ℓ_1-GM	42,21	38,14	35,48	33,29	30,84
ℓ_0-GM (Curv.)	*43,00*	*39,15*	*36,67*	*34,72*	*32,56*
ℓ_0-GM (Curv.+LDCT)	**43**,**48**	**39**,**65**	**37**,**09**	**35**,**14**	**32**,**96**

Cuadro 6.2: *PSNR (a través de MSE promedio) obtenido al restaurar las imágenes de nuestro conjunto de prueba tras la pérdida aleatoria de diferentes porcentajes de píxeles. La PSNR de la imagen observada se ha calculado usando la media en las posiciones perdidas. Para ℓ_0-GM y ℓ_1-GM se han utilizado Curvelets con 6 escalas y DCT local con tamaño de bloque 32×32 y factores de escala iguales para ambos ($\sqrt{0{,}5}$). ℓ_0-GM se presenta también usando sólo Curvelets. En negrita se muestran los mejores resultados para cada porcentaje, y en cursiva los segundos.*

Método	PSNR (dB)			
$\approx$ % píxeles perdidos	60	70	80	90
Observada	16,06	15,42	14,78	14,30
Fast-inpainting	27,78	26,48	*24,85*	**22**,**88**
ℓ_1-GM	28,75	26,38	22,84	18,54
ℓ_0-GM (Curv.)	*30,70*	*28,45*	24,79	*19,86*
ℓ_0-GM (Curv.+LDCT)	**31**,**03**	**28**,**77**	**25**,**14**	19,68

Cuadro 6.3: *Continuación de la Tabla 6.2 para mayores porcentajes de píxeles perdidos.*

En cuanto al tiempo de computación, ℓ_1-GM y ℓ_0-GM tardan en total un tiempo, obviamente, similar ($\approx$ 3 minutos). Sin embargo, *Fast-inpainting* es muy rápido (aproximadamente 0,5 segundos por imagen), lo que, junto a su sencillez, le convierte en una alternativa cuando no hay suficientes recursos para aplicar ℓ_0-GM.

Figura 6.7: *Ejemplo visual de interpolación de píxeles perdidos aleatoriamente.* **Arriba - izquierda**, *imagen* Barbara *de nuestro conjunto de prueba recortada a* 128×128. **Arriba - derecha**, *pérdida aleatoria de* $\approx 80\,\%$ *de los píxeles rellenados con la media de la imagen (PSNR:* 14,75 *dB).* **Centro - izquierda**, *interpolación conseguida con* ℓ_1*-GM (*23,26 *dB).* **Centro - derecha**, *resultado de* Fast-inpainting *(*24,84 *dB).* **Abajo - izquierda**, *resultado de* ℓ_0*-GM usando Curvelets* 6 *escalas (*25,19 *dB)* **Abajo - derecha**, *interpolación conseguida con* ℓ_0*-GM combinando Curvelets* 6 *escalas con DCT local con tamaño de bloque* 32×32 *y factores de escala iguales* $\sqrt{0{,}5}$ *(*25,65 *dB).*

6.2.5.2. Pérdida de zonas de píxeles

Recuperar zonas perdidas en la imagen es más difícil que interpolar píxeles aleatorios. En este apartado vamos a comparar el rendimiento de ℓ_0-GM respecto a los métodos descritos previamente. Tanto las imágenes de ejemplo como las máscaras de píxeles a estimar pueden encontrarse

en [144]. También hemos descargado de esa página los resultados mostrados del método *EM-inpainting*, forzando a continuación el valor de los píxeles observados para igualar las condiciones de comparación con los otros métodos[3].

La Figura 6.8 muestra un ejemplo particularmente interesante porque las zonas perdidas se localizan tanto en las zonas de textura como en las suaves. El panel de arriba a la izquierda es la observación, con los píxeles perdidos rellenados con la media de los píxeles observados. El panel de arriba a la derecha corresponde con *Fast-inpainting* (32,71 dB), y el de abajo a la izquierda con *EM-inpainting* (34,14 dB) usando Curvelets y DCT local usando bloques de 32×32. El último panel es el resultado de ℓ_0-GM usando Curvelets 6 escalas (34,92 dB). Es relevante observar que, a diferencia de lo que ocurría con la pérdida aleatoria de píxeles, ahora el resultado basado en minimizar la norma ℓ_1 es mejor que el basado en filtrados lineales iterativos. De nuevo, nuestro método ℓ_0-GM ofrece el mejor rendimiento. En la Figura 6.9 se puede ver ampliado el segundo cuadrante del resultado de *EM-inpainting*, a la izquierda, y ℓ_0-GM, a la derecha. Es particularmente interesante la recuperación parcial del ojo hecha por ℓ_0-GM, aunque también notamos una interpolación mucho mejor en la nariz y la boca.

La Figura 6.10 muestra la aplicación práctica a la restauración de fotografías dañadas. La imagen obtenida en [144] está algo deformada, por lo que se ha ampliado metiendo un factor de escala de 1,4 en la dirección vertical. También se ha eliminado la última fila de la imagen como requisito para el funcionamiento de las funciones de análisis y síntesis de la DCT local que utilizamos (número de filas y columnas múltiplo de la mitad del tamaño de bloque usado). El orden de los métodos[4] es el mismo que en la Figura 6.8, aunque en ℓ_0-GM se ha utilizado Curvelets 6 escalas combinado con DCT local. De nuevo, para *EM-inpainting* se han forzado los píxeles observados para comparar en igualdad de condiciones con el resto de métodos. Vemos nuevamente que ℓ_0-GM tiene un comportamiento cualitativo superior al resto de métodos. En contraste con los otros métodos, la línea gruesa horizontal inferior es apenas visible usando ℓ_0-GM. También las caras de las niñas, particularmente la más mayor y la más menor, están claramente mejor recuperadas con nuestro método.

[3]Notar que forzando los píxeles observados incrementa necesariamente la PSNR de la estimación, aunque puede hacer más visibles algunos artefactos.

[4]El resultado de *EM-inpainting* que se puede obtener en [144] tiene un tamaño distinto a la observación y a la máscara de píxeles perdidos, por lo que hemos replicado 3 veces la primera columna.

Figura 6.8: **Arriba - izquierda**, *imagen* Barbara *observada con los píxeles perdidos rellenados con la media de la imagen (PSNR:* 24,19 *dB).* **Arriba - derecha**, *resultado de* Fast-inpainting *(*32,71 *dB).* **Abajo - izquierda**, *resultado de* EM-inpainting *usando Curvelets con* 6 *escalas y DCT local con tamaño de bloque* 32×32 *(*34,14 *dB) y ambos factores de escala con valor* $\sqrt{0{,}5}$. **Abajo - derecha**, *nuestro resultado con* ℓ_0*-GM usando Curvelets* 6 *escalas (*34,92 *dB).*

6.2.6. Conclusiones

En esta sección hemos aplicado los métodos desarrollados en esta Tesis al problema de interpolación de píxeles perdidos en la imagen. Los resultados demuestran que la estrategia clásica de búsqueda del radio para ℓ_p-AP no ofrece resultados satisfactorios, así que hemos introducido una nueva técnica heurística para encontrar el radio, basada en maximizar el MSV de los píxeles estimados. En general, esta solución tiene una raleza menor que la obtenida con la estrategia clásica.

Figura 6.9: **Izquierda**, *detalle del resultado de* EM-inpainting *mostrado en la Figura 6.8 (*34,38 *dB).* **Derecha**, *lo mismo para el método* ℓ_0*-GM (*35,13 *dB).*

Figura 6.10: **Arriba - izquierda**, *fotografía real dañada.* **Arriba - derecha**, *resultado de* Fast-inpainting. **Abajo - izquierda**, *resultado de* EM-inpainting *usando Curvelets y DCT local.* **Abajo - derecha**, *nuestro resultado con* ℓ_0*-GM usando Curvelets* 6 *escalas y DCT local con tamaño de bloque* 32×32 *y ambos factores de escala a* $\sqrt{0{,}5}$.

Nuestros experimentos demuestran, sin embargo, que la aplicación de ℓ_0-GM usando la aproximación AS nos lleva a mucho mejores resultados. Esto sí es consistente con el modelo de favorecer las estimaciones ralas. Hemos comparado con un método muy eficiente (*Fast-inpainting*) [143] basado en la combinación de operaciones lineales y no lineales pero sobre el dominio de la imagen. Tambie´n hemos comparado con métodos basados en fomentar raleza a través de la minimización de la norma ℓ_1 (*EM-inpainting* [22] y ℓ_1-GM). En todos nuestros experimentos, tanto de pérdida de píxeles aleatorios como de pérdida de zonas localizadas relativamente grandes, ℓ_0-GM se ha mostrado muy superior a sus competidores.

En conclusión, hemos observado de nuevo que los métodos basados en encontrar soluciones sub-óptimas a la minimización de la norma ℓ_0 se comportan sustancialmente mejor que aquellos que minimizan de forma óptima la norma ℓ_1, para imágenes y representaciones usuales. Nuestros resultados también mejoran a otros algoritmos heurísticos. Debemos mencionar el buen comportamiento que muestra ℓ_0-GM tanto con altos porcentajes de píxeles perdidos aleatoriamente como en zonas de alta textura eliminadas y complicadas de interpolar. A este buen resultado ayuda el uso de Curvelets como representación, ya que la forma característica de sus funciones elementales representa eficientemente las características elongadas. Además, hemos visto que, normalmente, combinando Curvelets con DCT local se mejoran aún más los resultados, ya que se gana riqueza expresiva en las zonas de textura.

6.3. Interpolación espacial-cromática para mosaicos en cámaras digitales

6.3.1. Introducción

La mayoría de las cámaras digitales convencionales están basadas en la tecnología de Matrices de Filtros de Color (CFA, de *Color Filter Array*) introducida en [145]. Esto significa que contienen un sensor que capta sólo un color en cada píxel. Para reconstruir la imagen de color completa es necesario, en consecuencia, interpolar los componentes de color no captados en cada posición. Se conoce a este proceso de interpolación con el nombre de desmosaicado (*de-mosaicing*, en inglés).

Se han propuesto técnicas muy diversas para resolver este problema (ver [146] para una revisión). Para poder integrar el desmosaicado como parte del proceso de captura de una imagen digital, mantener una velocidad de cálculo alta es muy importante, sobre todo teniendo en cuenta el

incremento cada vez mayor de la resolución de los sensores CCD. La interpolación bilineal, por ejemplo, es muy rápida, pero el hecho de procesar independientemente cada canal cromático, ignorando la correlación entre ellos, resulta en pobres resultados [147]. Por otro lado, los métodos iterativos de alto rendimiento son demasiado lentos para capturar imágenes en tiempo real [148, 149, 150]. Sin embargo, sus buenos resultados permiten obtener imágenes de calidad si es posible post-procesar la imagen después del proceso de captura. Finalmente, se han desarrollado algunos métodos basados en filtrados lineales que tienen en cuenta la correlación entre canales, intentando llegar a un buen compromiso entre tiempo de computación y calidad de la imagen [151, 152, 147, 153].

En esta sección exploraremos el rendimiento de ℓ_0-GM aplicado a desmosaicado. Dada su naturaleza iterativa y basada en operaciones de análisis y síntesis con diccionarios redundantes, no podemos esperar competir en velocidad con los métodos existentes. Sin embargo, veremos que la alta calidad de sus resultados lo convierte en una buena alternativa cuando se puede post-procesar la imagen tras la captura. Primero compararemos los resultados de ℓ_0-GM con otros métodos propuestos en esta Tesis, y después lo compararemos exhaustivamente con tres métodos muy competitivos en el estado de la técnica actual.

6.3.2. Conjunto de consistencia

La degradación inherente al mosaico de Bayer consiste, al igual que ocurría en el problema de recuperación de zonas perdidas, en la pérdida de "píxeles"de la imagen tridimensional formada por los tres planos cromáticos de la imagen. Por lo tanto, el conjunto de imágenes consistentes con la observación será análogo al que hemos utilizado en la sección anterior. Es decir, está formado por todas aquellas imágenes tridimensionales que coincidan en las componentes de color observadas. Así, dado un conjunto de índices, I, entre 1 y $3N$, y dada una observación $\mathbf{y} \in \mathbb{R}^{3N}$, que conserva con respecto a la imagen original todos los píxeles y_i con $i \in I$, definimos el conjunto de consistencia asociado a $\mathbf{y}$, $R_d(\mathbf{y})$ como:

$$R_d(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \ x_i = y_i, \ \forall i \in I\}.$$

Dada una imagen RGB $\mathbf{x} \in \mathbb{R}^{3N}$ y una matriz diagonal $\mathbf{D}$ de tamaño $3N \times 3N$ donde cada elemento d_{ii} es 1 si $i \in I$ y 0 en caso contrario, la proyección ortogonal de un vector $\mathbf{v}$ sobre $R_d(\mathbf{y})$ es sencillamente $P_{R_d(\mathbf{y})}(\mathbf{v}) = \mathbf{D}\mathbf{y} + (\mathbf{I} - \mathbf{D})\mathbf{v}$, donde $\mathbf{I}$ es la matriz identidad de tamaño $3N \times 3N$.

6.3.3. Restricción añadida para favorecer la correlación cromática espacial

Como hemos dicho, existe una fuerte correlación entre las distribuciones de amplitudes de los píxeles de los tres planos cromáticos de una imagen RGB. De esta forma, si aplicamos nuestros métodos independientemente a cada canal no conseguimos buenos resultados, ya que no tener en cuenta esa correlación causa fuertes artefactos visuales en las estimaciones. Por esto, introducimos una modificación, en los métodos que favorecen la raleza, que mantiene la metodología y las propiedades de convergencia pero consigue conservar mejor la correlación entre canales.

Para empezar, vamos a cambiar el espacio de color desde el que transformamos las imágenes a nuestro dominio redundante de la representación. Utilizaremos YUV en vez de RGB, ya que en este nuevo espacio se reduce mucho la correlación entre canales, porque se compone de una componente de luminancia y dos de crominancia. Como la transformación de un espacio a otro y viceversa es lineal, no afecta a las propiedades geométricas o de convergencia del método, pues la transformación compuesta por el cambio en el dominio del color y la transformación redundante todavía es un marco ajustado, siempre que esta última ya lo fuese.

Las componentes cromáticas U y V representan diferencias entre colores. De alguna forma están indicando como de correlacionados están los canales. Nosotros queremos que esa correlación se mantenga alta, para evitar los clásicos artefactos de alta frecuencia en los colores. Por lo tanto, buscamos que los elementos de una transformación de este tipo sean bajos, especialmente en las frecuencias altas. De esta forma, buscamos incluir una restricción de suavidad en los canales cromáticos de YUV. Esto es consistente con muchas representaciones de imágenes en color, que usan menos información para los canales de crominancia que para la luminancia. Un ejemplo es el sistema PAL de televisión en color.

La modificación que proponemos consiste en introducir una etapa más en cada iteración del algoritmo. Esta nueva etapa pone a cero todos los coeficientes de las sub-bandas de más alta frecuencia de la representación wavelet de U y V. Esto puede interpretarse como un conocimiento añadido al modelo *a priori*, de forma que favorecemos simultáneamente no sólo a las imágenes que tengan representaciones ralas sino también a aquellas cuyos canales cromáticos esté altamente correlacionados en las altas frecuencias espaciales. Nótese que forzar a cero las altas frecuencias de la transformación de los canales cromáticos de YUV no está en contra de buscar alta raleza, ya que se están disminuyendo el número de coeficientes de alta

energía. Además, esta restricción conlleva una proyección ortogonal sobre un conjunto convexo, por lo que, básicamente, no afecta a las propiedades de convergencia del método.

Una idea parecida puede verse en [152], donde se aplica un filtro paso - bajo a las frecuencias de los canales R y B, que consigue eliminar los artefactos de color que aparecen tras la interpolación bilinear. Sin embargo, también suaviza en exceso la imagen, mientras que nuestro método preserva muy bien los detalles de alta frecuencia de la lumninancia, como veremos en el apartado 6.3.5.

6.3.4. Implementación

Métodos existentes. Hemos comparado con métodos que ofrecen prestaciones competitivas con el estado de la técnica actual. El primer está basado en mantener la correlación entre canales usando proyecciones alternas [148]. El segundo es un método iterativo que actúa sobre las diferencias de color utilizando un criterio de parada adaptativo en ese espacio, con el objetivo de mantener la correlación entre canales de color y eliminar los artefactos de cremallera [150]. Por último, también comparamos con un método basado en heurísticos bien adaptados a la naturaleza del problema, que es efectivo para mantener la correlación entre canales, aunque también poco eficiente en términos de computación [149]. Todos estos métodos se encuentran implementados en un paquete MATLAB® disponible en la página web del Profesor Xin Li [154].

ℓ_p-GM. En todos los experimentos hemos usado $\alpha = \alpha_0$ y $\beta = 0{,}8$, que ofrecen el mejor compromiso entre tiempo de computación y calidad. Hemos establecido el umbral de parada para las iteraciones en 0,01. Para esta aplicación utilizamos 5 escalas tanto en DT-CWT como en Curvelets.

6.3.5. Resultados y discusión

6.3.5.1. Comparación entre ℓ_0-GM y ℓ_1-GM

Los resultados que se obtienen para este problema usando ℓ_p-AP son peores, y mucho más lentos, que los obtenidos con la interpolación bilinear. Por ese motivo eliminamos este método de la comparación.

En nuestros experimentos también hemos encontrado que ℓ_0-GM ofrece mucho mejores resultados que ℓ_1-GM. La Figura 6.11 ofrece un ejemplo, donde los métodos se han aplicado al mosaico de Bayer de patrón 'GB' construido con la imagen 15 de la base de datos de *Eastman Kodak* [155]. En el panel superior izquierdo se muestra un recuadro de tamaño 64×64 de la imagen original con un patrón de muy alta frecuencia y, por ello,

particularmente complicada de interpolar. El resultado del panel superior derecho corresponde a ℓ_1-GM, utilizando DT-CWT. La PSNR por cada canal es: 37,24 dB para el canal rojo (R), 39,87 para el verde (G), y 37,17 para el azul (B). Vemos que se mantienen algunos artefactos de color. El panel inferior izquierdo es el resultado de ℓ_0-GM con Curvelets (39,29, 42,27 y 38,20 dB). Por último, el panel inferior derecho corresponde a ℓ_0-GM con DT-CWT (39,59, 41,99 y 39,07 dB). Vemos que los resultados de ℓ_0-GM son ambos significativamente mejores que el de ℓ_1-GM, en términos de eliminar artefactos de color. La PSNR es similar en ambas representaciones para ℓ_0-GM, pero tenemos que, por una parte, las diferencias en esos niveles de decibelios no son significativas; y por otra la correlación entre colores parece mantenerse mejor usando DT-CWT. Además, nuestra implementación usando Curvelets es más de 4 veces más lenta que usando DT-CWT, lo que significa que tarda sobre los 100 minutos por cada imagen completa de [155], mientras que DT-CWT tarda sobre 25 minutos.

6.3.5.2. Comparación entre ℓ_0-GM y otros métodos

En la Tabla 6.4 se muestra el MSE promedio para cada canal RGB obtenido con los diferentes métodos para las imágenes 18, 31, 32, 33, 12, 34, 39, 15, 40, 16, 17 y 19 de [155] (misma selección que [150]). También se promedia el error en métrica ∇E^*_{ab} de S-CIELab [156] y el tiempo de computación. Hemos mostrado el MSE en vez de la PSNR porque es un dato más significativo en estos ejemplos, debido a la escasa diferencia entre los métodos y a la alta relación señal ruido que consiguen. Además, en las Tablas 6.5 y 6.6 hemos desglosado estos resultados para cada imagen. Vemos que, aunque ℓ_0-GM es más lento que sus competidores, ofrece un rendimiento comparable al resto, siendo el mejor o segundo mejor en varios casos.

Sin embargo, queremos enfatizar que nuestro método rinde particularmente bien en zonas de alta frecuencia muy complicadas de interpolar sin artefactos. En la Figura 6.12 hemos puesto el recorte de los resultados de nuestros métodos para una zona especialmente difícil de la imagen 15 de [155] (tamaño 128×128 empezando en la posición (228, 323)). Se puede apreciar la significativa reducción de artefactos de color que consigue nuestro método. También hemos reducido los artefactos de cremallera con respecto a los métodos [149] y [148]. Estos artefactos se deben a la imposición de los valores observados, de acuerdo con la estructura de mosaico, en zonas que no están suficientemente bien interpoladas [150].

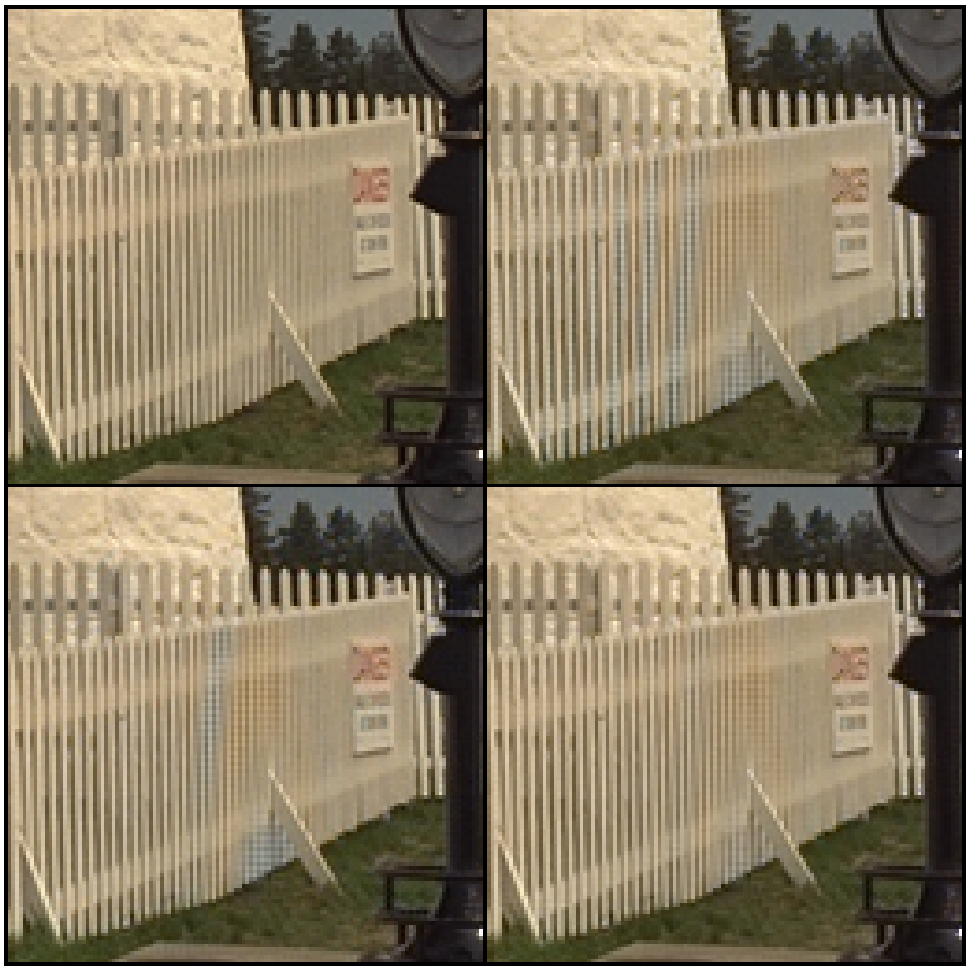

Figura 6.11: *Comparación visual entre los métodos ℓ_1-GM y ℓ_0-GM aplicados a desmosaicado.* **Arriba-Izquierda**, *Detalle de la imagen* 15 *de la base de datos de* Eastman Kodak, *recortada a tamaño* 64×64. **Arriba-Derecha**, *resultado de desmosaicado con patrón de Bayer 'GB' del método ℓ_1-GM usando DT-CWT. La PSNR para los canales R, G y B respectivamente es* 37,24 *dB,* 39,87 *dB y* 37,17 *dB.* **Abajo-Izquierda**, *resultado de ℓ_0-GM usando Curvelets (*39,29*,* 42,27 *y* 38,20 *dB).* **Abajo-Derecha**, *resultado de ℓ_0-GM usando DT-CWT (*39,59*,* 41,99 *y* 39,07 *dB).*

6.3.6. Conclusiones

Hemos planteado en esta sección la aplicación de ℓ_0-GM al problema de interpolación de los componentes cromáticos perdidos tras tras capturar una imagen en un mosaico de Bayer. Hemos aplicado nuestros métodos para encontrar aproximaciones ralas a la representación de la imagen en el dominio YUV. Para favorecer la regularidad cromática de la imagen, hemos introducido una proyección extra que minimiza la norma ℓ_2 de los

Método	MSE			∇E^*_{ab}	Tiempo
	R	G	B	*S-CIELab*	(sg.)
Lu & Tam [149]	8,67	5,59	11,91	**0,78**	932,22
Gunturk et al. [148]	7,92	*3,61*	10,60	0,84	*9,40*
Li [150]	**7,66**	**3,46**	**8,61**	*0,83*	**2,31**
ℓ_0-GM	*7,68*	4,31	*9,53*	0,93	1238,50

Cuadro 6.4: *Promedio de MSE y error promedio S-CIELab para las 12 imágenes descritas en el texto, incluyendo tiempos de computación. En negrita se muestra el mejor resultado para cada columna, y en cursiva el segundo mejor. Ver el texto para más detalles.*

componentes de alta frecuencia de los canales U y V forzando a cero las subbandas de más alta frecuencia en cada iteración. Esta modificación no altera la interpretación geométrica y las propiedades básicas de convergencia del algoritmo.

Hemos vuelto a observar que la solución basada en favorecer raleza mediante la minimización local de la norma ℓ_0 mejora a la minimización óptima de ℓ_1. También hemos visto que ℓ_0-GM es comparable con métodos competitivos con el estado de la técnica, aunque su velocidad sea algo menor. Además, hemos visto que nuestro método se comporta particularmente bien, superando claramente a sus competidores, en zonas particularmente difíciles de manejar, reduciendo significativamente los artefactos. Por lo tanto, podemos concluir que es una alternativa prometedora cuando se pueden procesar las imágenes como paso posterior a la captación de la misma.

El resultado al que llega ℓ_0-GM sigue conservando bastantes artefactos de cremallera. Estos artefactos se deben al hecho de forzar los píxeles observados en zonas que no están suficientemente bien interpoladas.

6.4. Incremento de detalle

6.4.1. Introducción

A menudo las imágenes sufren un proceso de pérdida de resolución. Esto puede ocurrir, por ejemplo, el la captación usando fotodetectores que integren la luz que incida sobre ellos; o también al ser transmitidas, debido a una capacidad limitada del canal.

El incremento de detalle o super-resolución de imágenes consiste en el proceso de obtener una imagen o secuencia de imágenes de alta resolución a partir de un conjunto de observaciones de baja resolución [157]. Hay muchos

Imagen	Método	MSE			∇E^*_{ab}	Tiempo
		R	G	B	*S-CIELab*	(sg.)
	[149]	3,26	*1,51*	**3,16**	**0,56**	981,77
18	[148]	**2,69**	2,01	4,88	0,62	9,71
	[150]	4,25	1,99	3,49	0,63	1,78
	ℓ_0-GM	*3,05*	**1,44**	*3,24*	*0,58*	1303,50
	[149]	9,73	6,43	11,23	**0,88**	980,13
31	[148]	*7,98*	*3,46*	13,29	0,97	10,09
	[150]	8,15	**3,26**	*9,86*	**0,88**	2,07
	ℓ_0-GM	**7,21**	3,82	**8,25**	*0,93*	1368,15
	[149]	**3,06**	**1,75**	**4,12**	**0,51**	969,55
32	[148]	*3,43*	*2,21*	6,59	0,63	8,63
	[150]	3,99	2,49	5,51	0,65	2,39
	ℓ_0-GM	4,29	2,23	*4,81*	*0,57*	1316,93
	[149]	20,50	12,17	19,81	**1,29**	970,12
33	[148]	18,28	*7,42*	23,06	1,40	8,84
	[150]	*16,74*	**6,28**	**14,48**	*1,35*	3,09
	ℓ_0-GM	**15,71**	8,44	*17,88*	1,48	1384,49
	[149]	*3,30*	2,05	*4,06*	**0,53**	971,74
12	[148]	**3,29**	*1,79*	5,43	*0,60*	10,21
	[150]	3,57	**1,77**	**3,95**	0,58	2,30
	ℓ_0-GM	3,40	2,02	4,37	*0,60*	1428,25
	[149]	*8,27*	5,14	*7,27*	**0,73**	973,10
34	[148]	**6,84**	*3,78*	7,61	0,76	9,05
	[150]	8,38	**3,29**	**6,90**	*0,75*	2,06
	ℓ_0-GM	9,65	4,39	9,05	0,96	1403,20

Cuadro 6.5: *Datos de MSE, error S-CIELab y tiempo de computación para 6 de las 12 imágenes descritas en el texto y los cuatro métodos comparados. En negrita se muestra el mejor resultado para cada columna y cada imagen, y en cursiva el segundo mejor, salvo en la columna de tiempos, donde coinciden siempre con los datos de la Tabla 6.4. Ver el texto para más detalles.*

trabajos que tratan el problema cuando hay múltiples observaciones (por ejemplo, vídeo). Este problema se llama super-resolución dinámica (*dynamic super-resolution*, en inglés, ver, por ejemplo [158, 159, 157]). Aquí nos hemos centrado en el caso de tener una sola observación, llamado super-resolución estática (*static or single-frame super-resolution*, en inglés). Este problema también se conoce como incremento de detalle, re-escalado de

Imagen	Método	MSE			∇E^*_{ab}	Tiempo
		R	G	B	*S-CIELab*	(sg.)
	[149]	4,91	3,18	5,50	*0,83*	976,80
39	[148]	4,53	**1,67**	6,23	0,92	8,73
	[150]	*4,34*	*1,72*	*4,92*	**0,82**	2,32
	ℓ_0-GM	**3,30**	1,78	**3,94**	**0,82**	1385,81
	[149]	6,80	4,53	8,02	**0,74**	975,13
15	[148]	*6,67*	*2,67*	*7,71*	**0,74**	9,66
	[150]	**5,88**	**2,51**	**6,86**	**0,74**	2,24
	ℓ_0-GM	7,08	4,12	7,93	*0,88*	1144,42
	[149]	6,83	5,12	34,98	*0,62*	890,25
40	[148]	10,57	**2,17**	*7,71*	**0,61**	9,77
	[150]	**5,35**	*2,80*	7,80	*0,62*	2,40
	ℓ_0-GM	*5,55*	3,04	**7,44**	0,69	1488,72
	[149]	8,97	6,49	10,04	*0,85*	832,32
16	[148]	*7,43*	**3,26**	**9,34**	**0,84**	8,29
	[150]	8,08	*3,28*	*9,45*	0,86	1,99
	ℓ_0-GM	**7,22**	4,52	10,13	1,10	1495,98
	[149]	*7,72*	4,71	**8,85**	**0,85**	833,02
19	[148]	**7,06**	*4,66*	10,95	*0,92*	9,77
	[150]	8,65	**4,52**	*9,52*	0,94	2,10
	ℓ_0-GM	9,82	6,01	11,97	1,07	1436,15
	[149]	20,64	13,96	25,91	**0,99**	832,75
19	[148]	16,33	*8,19*	*24,35*	*1,05*	10,02
	[150]	**14,58**	**7,57**	**20,59**	1,09	3,04
	ℓ_0-GM	*15,80*	9,94	25,38	1,46	1452,55

Cuadro 6.6: *Continuación de la Tabla 6.5 para el resto de imágenes utilizadas.*

imágenes (*image scaling*, en inglés), interpolación, acercamiento (*zooming-in*, en inglés) o aumento del tamaño de la imagen (*enlargement*, en inglés).

Existen métodos lineales muy sencillos (tales como interpolación bilineal, bicúbica, etc), que tratan a todos los píxeles por igual, interpolando el valor de cada uno a partir de una combinación lineal del valor de sus vecinos. La linealidad limita seriamente la calidad final. Por una parte se produce un excesivo emborronamiento, que difumina los bordes e impide la conservación de los detalles. Por otro lado, a menudo aparecen artificios de alisado (*aliasing*, en inglés).

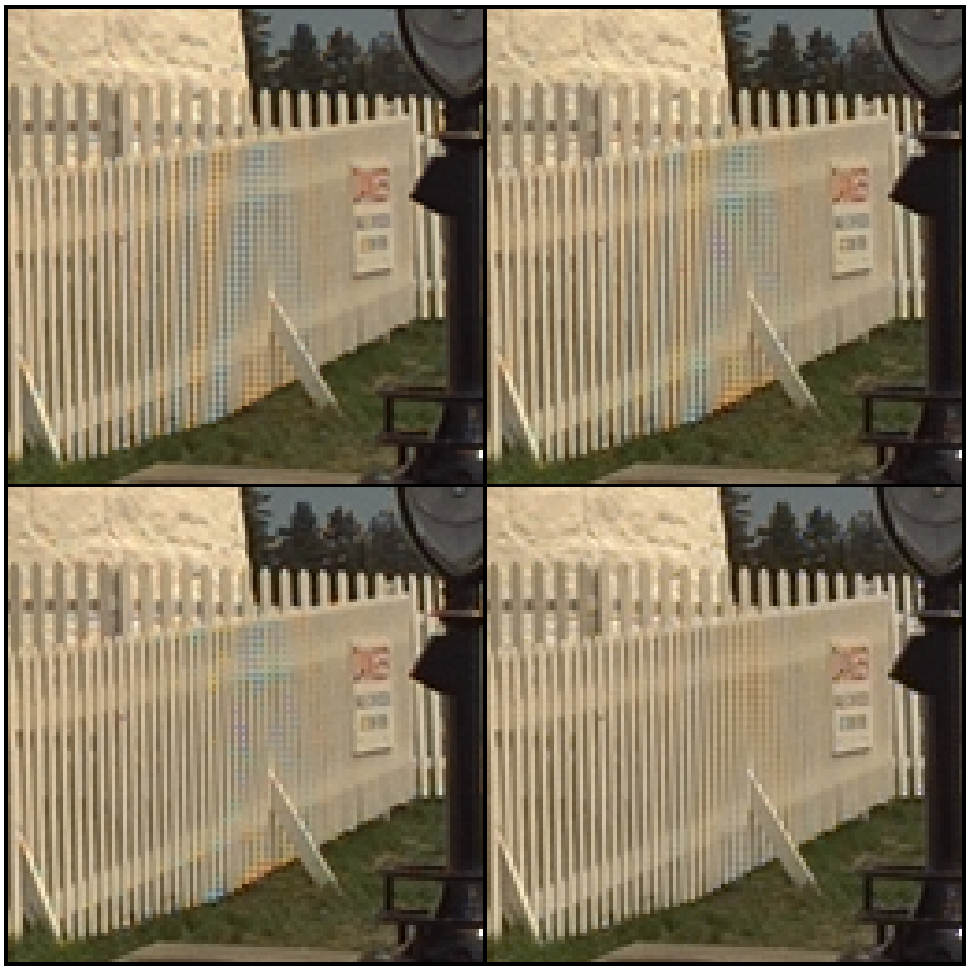

Figura 6.12: *Comparación visual entre los métodos de desmosaicado.* **Arriba - izquierda**, *resultado del método de [149] sobre imagen 15 de la base de datos de* Eastman Kodak *(*Lighthouse*). Se ha recortado la imagen a tamaño* 64×64 *para mejorar la visibilidad de los artefactos. La PSNR de las respectivas bandas R,G,B es* 39,81, 41,57 *y* 39,09 *dB.* **Arriba - derecha**, *resultado del método de [148] (*39,89, 43,87 *y* 39,26 *dB).* **Abajo - izquierda**, *resultado del método de [150] (*40,39, 44,14 *y* 39,73 *dB).* **Abajo - derecha**, *nuestro resultado con* ℓ_0*-GM (*39,59, 41,99, 39,07 *dB).*

Estos problemas de la interpolación lineal han motivado el estudio de algoritmos de super-resolución más potentes, basados en técnicas no lineales. Todos ellos tienen en común que aprovechan, de una forma u otra, la fuerte correlación entre píxeles vecinos para estimar los valores de los píxeles perdidos. Algunos de ellos utilizan técnicas más o menos heurísticas adaptadas al problema (ver, por ejemplo, [160, 161]). Otros están basados en aprendizaje de las relaciones entre las imágenes observadas y los originales de alta resolución, como [162]. En estos métodos se echa en

falta un modelo matemático que sustente su buen rendimiento. Existen otra serie de métodos, más cercanos al ámbito de esta Tesis, basados en modelos *a priori* de la estadística de las imágenes naturales. Por ejemplo, [163] establece un modelo basado en distribuciones de alta kurtosis. De forma similar, [164] desarrolla un método bastante exitoso que utiliza un método basado en favorecer la raleza a través de minimizar la norma ℓ_1 de la estimación. Ver [165] para una revisión reciente (2006) e interesante de los métodos de super-resolución estática no lineales.

En este capítulo presentamos la aplicación de ℓ_p-GM al problema de super-resolución estática. Veremos que las dos versiones estudiadas, $p = 0$ y $p = 1$, se comportan de manera similar tanto en términos de PSNR como visuales, aunque ℓ_0-GM obtiene estimaciones más ralas. Hemos comparado también, por tener una referencia, con la interpolación por vecino más cercano y con la interpolación bilineal como representantes de los métodos lineales. Nuestro objetivo en esta sección no es tanto plantear una alternativa a los métodos actuales de super-resolución estática sino realizar una primera aproximación de la potencialidad de los métodos y modelos usados en esta Tesis para este tipo de problemas.

6.4.2. Conjunto de consistencia

Llamamos L al número de píxeles que se promedian para obtener cada píxel observado. Notamos $\mathbf{y} \in R^{N/L}$ a la imagen observada. Definimos una familia de conjuntos de índices, J_i, con $i = \{1, ..., N/L\}$, y que corresponde a los bloques no solapados de tamaño $\sqrt{L} \times \sqrt{L}$ formados por aquellos $j \in \{1, ..., N\}$ cuyos píxeles correspondientes en la imagen original, $\mathbf{x}_0 \in \mathbb{R}^N$, han sido promediados para dar el valor observado y_i. También definimos $\mathbf{x}_0^{J_i}$ como el bloque formado por los píxeles de $\mathbf{x}_0 \in \mathbb{R}^N$ en las posiciones indicadas por J_i. El conjunto de consistencia, $R_a(\mathbf{y})$, está entonces formado por aquellas imágenes $\mathbf{x} \in \mathbb{R}^N$ cuyos bloques asociados, $\mathbf{x}^{J_i}$, preservan los promedios observados. Entonces tenemos que:

$$R_a(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \ < \mathbf{x}^{J_i} >= y_i, \ \forall i \in \{1, ..., N/L\}\}, ,$$

donde $< \mathbf{a} >$ denota el valor promedio de los coeficientes del vector $\mathbf{a}$. Geométricamente, es fácil ver que la proyección sobre el subespacio afín de vectores que preservan un promedio dado consiste en restar un vector constante en amplitud, que es la diferencia entre el promedio de los coeficientes del vector que se proyecta y el promedio objetivo. Así, dado $\mathbf{x} \in \mathbb{R}^N$, tenemos que $P_{R_a(\mathbf{y})}(\mathbf{x}) = \mathbf{z}$, donde, para todo $i \in \{1, ..., N/L\}$ y $j \in \{1, ..., N\}$::

$$z_j^{J_i} = x_j^{J_i} - (< \mathbf{x}^{J_i} > - y_i).$$

6.4.3. Implementación

ℓ_p-GM. Hemos experimentado que, en su aplicación a super-resolución estática, los mejores resultados ℓ_p-GM, tanto para $p = 0$ como para $p = 1$, se obtienen cuando se usa DT-CWT con sólo 3 escalas. Los parámetros utilizados son $\alpha = \alpha_0$ y $\beta = 0{,}8$. Las iteraciones se paran cuando el umbral está por debajo de 0,01.

6.4.4. Resultados y discusión

En la Tabla 6.7 comparamos los métodos ℓ_0-GM, ℓ_1-GM, replicado del vecino más cercano y filtrado bilineal a través de la PSNR de la estimación promediada en las imágenes de nuestro conjunto de prueba. Para empezar, es interesante recalcar el buen comportamiento del método más sencillo posible, que es el del vecino más cercano. Este se debe a que está imponiendo la media local, lo que en promedio es la mejor estrategia lineal posible para esta degradación. Por otro lado, vemos que el método ℓ_p-GM, en sus dos casos $p = 0$ y $p = 1$, se comporta bastante bien. Es curioso observar que, a diferencia del resto de degradaciones estudiadas, los dos casos de nuestro método tienen resultados muy similares en cuanto a PSNR. Esto no quiere decir que los resultados sean estrictamente similares. Esa PSNR similar se consigue en ℓ_0-GM con una selección significativamente más rala de coeficientes que en ℓ_1-GM ($\approx 2{,}15 \cdot 10^5$ y $\approx 2{,}45 \cdot 10^5$ en promedio respectivamente). A la vista de estos resultados, concluimos que el rendimiento relativo de ℓ_0-GM ha bajado con respecto a las anteriores aplicaciones, probablemente porque a partir de cierto punto aumentar la raleza en el conjunto de consistencia (al menos usando nuestros métodos) no ayuda a disminuir el error con respecto a la original (esta hipótesis es consistente con los malos resultados de ℓ_0-GM en la eliminación de artefactos de cuantificación espacial). De hecho, si utilizamos valores de β más cercanos a 1, la raleza aumenta aún más pero el error de la estimación también crece.

En la Figura 6.13 se muestra una comparación, usando *House*, de los resultados visuales de los métodos. Tanto el correspondiente a ℓ_1-GM como el de ℓ_0-GM tienen una apariencia visual más nítida y un mejor comportamiento en bordes que las dos estrategias lineales comparadas. Incluso se elimina significativemente el alisado (obsérvese, por ejemplo, los bordes del tejado). Por último, se ve que, en este caso, apenas hay diferencia, pese a la disparidad en la raleza entre ℓ_1-GM y ℓ_0-GM.

	PSNR (dB)				
Método	*Barbara*	*Boat*	*House*	*Lena*	*Peppers*
Vecino más cercano	27,09	26,54	30,52	27,68	25,74
Interpolación bilineal	24,16	24,25	25,84	25,41	23,76
ℓ_1-GM	**27,49**	**28,84**	*33,50*	*30,66*	*28,06*
ℓ_0-GM	*27,14*	*28,80*	**33,53**	**30,75**	**28,07**

Cuadro 6.7: *PSNR (promediado en MSE) obtenido en la super-resolución realizada por diferentes métodos para recuperar el tamaño original de las imágenes de nuestro conjunto de prueba promediadas en bloques* 2×2 *y submuestreadas. En negrita se muestran los mejores resultados para cada imagen, y en cursiva el segundo mejor.*

6.4.5. Conclusiones

En esta sección hemos explorado la aplicación de los métodos ℓ_0-GM y ℓ_1-GM al problema de super-resolución estática. Ambos mejoran claramente el rendimiento de los métodos no lineales comparados (vecino más cercano e interpolación bilineal). Sin embargo, vemos que la solución más rala, ofrecida por ℓ_0-GM, no es mejor que la de ℓ_1-GM, y además, para ambos métodos el incremento en el rendimiento se satura al aumentar el nivel de raleza de la estimación. Estos son resultados preliminares que necesitan mejorarse en el futur, a través de entender mejor los mecanismos de favorecimiento de la raleza para este tipo de aplicaiones.

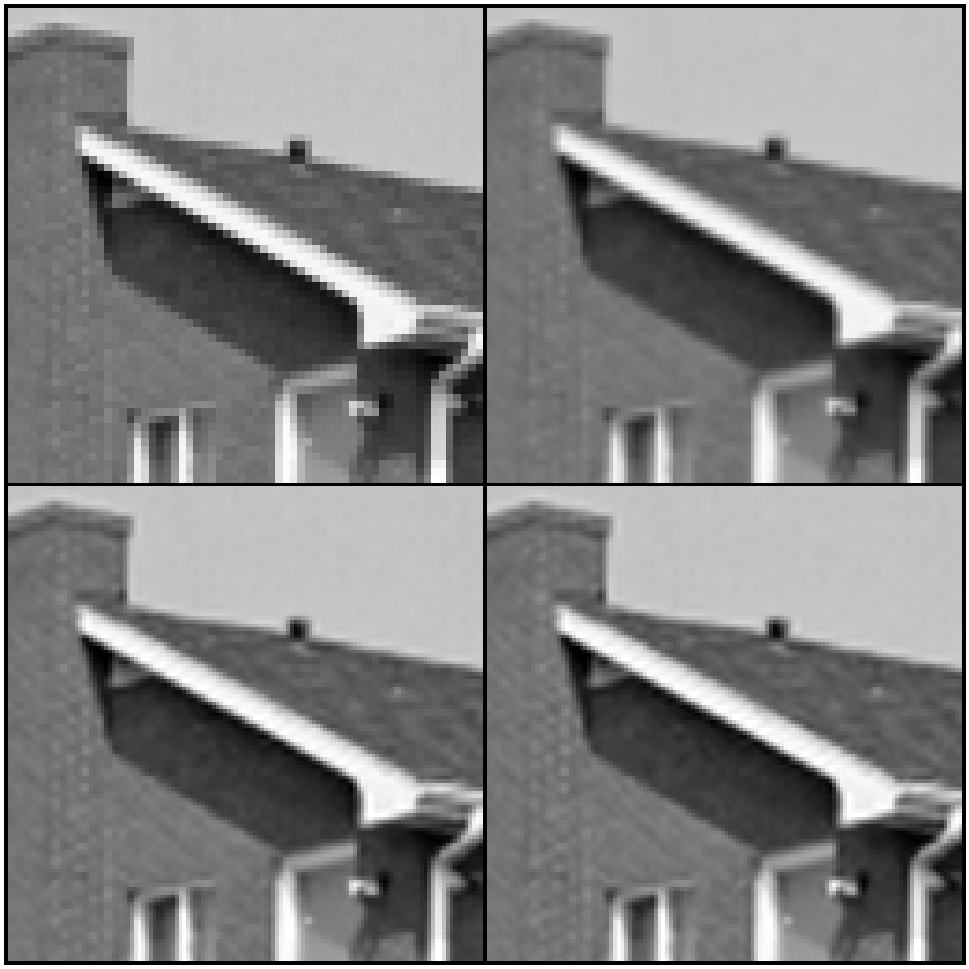

Figura 6.13: **Arriba - izquierda**, *resultado de aplicar interpolación por vecino más cercano (replicado) al promediado de bloques* 2×2 *y submuestreo de la imagen* House, *para recuperar la resolución original (PSNR:* 30,52 *dB). Hemos recortado una región de tamaño* 128×128 *para mejorar la visibilidad.* **Arriba - derecha**, *resultado de la interpolación bilineal (*29,65 *dB).* **Abajo - izquierda**, *resultado de* ℓ_1*-GM (*33,50 *dB).* **Abajo - derecha**, *resultado de* ℓ_0*-GM (*33,53 *dB).*

Capítulo 7

Conclusiones y trabajo futuro

7.1. Conclusiones

La principal conclusión que se puede extraer del trabajo presentado es que, a pesar de que la optimización global planteada en el problema de aproximación rala es NP-complejo, no sólo es posible sino que es relativamente sencillo encontrar formulaciones equivalentes que permitan, usando marcos ajustados, aplicar técnicas conocidas de optimización para encontrar, al menos, mínimos locales. Aunque los métodos propuestos han sido recientemente propuestos y utilizados como heurísticos, hasta el momento, que nosotros sepamos, no se había establecido un marco teórico apropiado que permitiese obtenerlos como solución a un problema clásico de optimización. Esto choca con la creencia, ampliamente extendida en la comunidad científica de este campo, de que sólo es posible fundamentar estos métodos en la teoría si se usan aproximaciones convexas a la función de coste.

El objetivo de esta Tesis no ha sido la obtención de condiciones teóricas bajo las cuales estos métodos encuentren el óptimo global. En su lugar, hemos preferido estudiar exhaustivamente su rendimiento en condiciones reales de procesamiento de imágenes, tanto en términos de capacidad de concentración de la energía como en términos de su aplicación a diversos problemas de restauración.

Hemos derivado dos métodos diferentes para resolver el problema de aproximación rala. El primero de ellos se formula desde el problema equivalente de minimizar el error cuadrático medio dentro con respecto a la imagen representada desde los vectores de síntesis de una bola ℓ_p de radio dado. Resulta en un método basado en proyecciones ortogonales alternas sobre dos conjuntos: 1) el conjunto de vectores que reconstruyen perfectamente la imagen; y 2) la bola ℓ_p de radio dado. Hemos denominado

ℓ_p-AP a este método. Nos hemos centrado en los casos $p = 0$, para el que obtenemos soluciones locales para el problema de aproximación rala, y $p = 1$, con el que encontramos la solución global del problema de relajación convexa. En los experimentos hemos demostrado que, en las condiciones presentadas (imágenes naturales y niveles útiles de raleza en diccionarios redundantes utilizados asiduamente en procesamiento de imágenes), el resultado de ℓ_0-AP mejora a ℓ_1-AP y a otras técnicas existentes, como la umbralización iterativa con umbral fijo y los métodos voraces. Sin embargo, hemos visto que la elección de las funciones elementales utilizadas para la aproximación es ligeramente mejor en ℓ_1-AP. Además, la necesidad de elegir un valor determinado de raleza supone un grave inconveniente en la práctica.

Para superar estos problemas, hemos propuesto otro método. Primeramente, hemos reformulado nuestro problema de optimización encontrando una función continua y restringida equivalente a la función de coste del problema de aproximación rala (discontinua y no restringida). Entonces, hemos derivado una versión generalizada de IHT usando descenso de gradiente en esta nueva función. Hemos demostrado que el punto de convergencia de este método es un mínimo local al problema de aproximación rala. Finalmente, hemos propuesto el método ℓ_0-AP, mediante la reescritura de la nueva función como una función de coste infinitamente abrupta convolucionada con un filtro de suavizado, lo que nos permite usar una aproximación parecida al enfriamiento determinista para justificar el uso de umbrales decrecientes. Hemos derivado también una versión similar de este método usando la norma ℓ_1, ℓ_1-GM, cuyo uso está recomendado cuando requerimos a la estimación a tener bajos niveles de raleza.

Hemos experimentado que ℓ_0-GM supera a todos los métodos comparados en términos de obtener buenas aproximaciones ralas, incluido ℓ_0-AP y la minimización del error de reconstrucción para el soporte dado por ℓ_1-AP. De esta forma, podemos concluir que, en términos de compactación de energía, los métodos basados en resolver directamente el problema de aproximación rala ($p = 0$) superan, en las condiciones de este estudio, a los métodos basados en resolver el problema de relajación convexa ($p = 1$).

Una observación importante que hemos realizado es que el número de coeficientes necesarios para obtener reconstrucción perfecta de la imagen usando ℓ_0-GM tiende a la asíntota teórica conforme el enfriamiento se realiza más lentamente. Esto indica un comportamiento asintóticamente óptimo en la curva fidelidad - raleza. Además, la solución conseguida con ℓ_0-GM, para niveles bajos de raleza, aproxima muy bien la solución óptima para otros niveles de aproximación, si realizamos umbralización *a posteriori* de este resultado. Esto ofrece importantes ventajas en la práctica, pues nos permite ahorrar la búsqueda del nivel óptimo de raleza como primera etapa,

simplificando la implementación final del método y aumentando la eficiencia de ajustar el nivel de raleza posteriormente.

En cuanto a la aplicación a problemas de restauración, hemos visto que es generalmente fácil adaptar nuestros métodos para restaurar imágenes afectadas por degradaciones estrictamente reproducibles. Hemos estudiado dos tipos de raleza diferentes que pueden ser utilizados como modelo *a priori* en las aplicaciones. Por un lado, la raleza de síntesis (SS) asume que las imágenes naturales pueden expresarse como combinación lineal de pocas funciones elementales del diccionario. Esta aproximación, aunque válida, tiene dudosa justificación empírica. En vez de esto, hemos propuesto el uso de modelos *a priori* sobre los coeficientes de análisis (SA), que se basa en la observación de que los vectores de análisis de una transformación redundante de tipo ondícula concentran su energía en pocos coeficientes. A diferencia de SS, esta segunda opción permite tener una base empírica justificada por observaciones directas. La adaptación de los métodos para fomentar SA no requiere cambiar la consistencia del marco conceptual.

Hemos propuesto el uso de ℓ_0-AP, con SS, para la eliminación de artefactos de cuantificación espacial y de ℓ_0-GM, con SA, para diversos problemas de interpolación, como la recuperación de píxeles perdidos, la construcción de imágenes en color a partir de mosaicos y el incremento de detalle. Hemos visto que la minimización de la norma ℓ_1 ofrecía resultados más pobres para casi todas las aplicaciones estudiadas. Además, se ha visto que nuestras propuestas ofrecen un rendimiento similar o superior al de otros métodos existentes en la literatura. Hasta donde llega nuestro conocimiento, nadie había aplicado antes técnicas que favoreciesen la raleza de la estimación para el problema de eliminación de artefactos de cuantificación espacial.

A pesar de los buenos resultados obtenidos, los experimentos de restauración realizados nos permiten concluir que el modelo utilizado puede ser mejorado. Por ejemplo, hemos visto que los resultados de ℓ_0-GM en la eliminación de artefactos de cuantificación espacial son peores que los de ℓ_0-AP, siendo los primeros claramente más ralos. Además, el método *per se* no consigue interpolar satisfactoriamente redes regulares de píxeles perdidos, pues cae en mínimos locales poco favorables que tienden a representar los artefactos más sobresalientes. Hemos tenido que utilizar modificaciones más o menos heurísticas para forzar al método a salir de estos mínimos locales. No obstante, este problema no sólo se debe achacar al método, sino también a una concepción de alguna forma demasiado simple del modelo de raleza.

7.2. Trabajo futuro

Creemos que esta Tesis abre varas vías interesantes de investigación para el futuro. Por un lado, la estadística marginal de los coeficientes de la respuesta de las ondículas redundantes a imágenes naturales puede aproximarse más fielmente utilizando normas intermedias ($0 < p < 1$) [80, 78]. En este sentido, creemos que es posible, aunque ni mucho menos trivial, derivar un método análogo a ℓ_0-GM y ℓ_1-GM que esté basado en ellas. También está justificado el uso de normas intermedias como modelo *a priori* para los coeficientes de análisis en la aplicación a restauración de los métodos (ver, por ejemplo, [31]).

Por otro lado, hay mucho campo de estudio en cuanto a obtener modelos *a priori* para los coeficientes de síntesis de una forma más justificada. Por un lado, se pueden usar las normas intermedias como compromiso entre el buen rendimiento de ℓ_0 y la habilidad de ℓ_1 para evitar mínimos locales. Por otro lado, estamos trabajando en un método totalmente justificado en base a Expectación-Maximización, basado en maximizar la verosimilitud de un modelo para los coeficientes de síntesis.

Con una orientación más práctica, pero también con importantes repercusiones teóricas, tenemos la intención de explorar los problemas de interpolación con redes regulares. Creemos que estos problemas se deben a la excesiva sencillez del modelo.

Por último, sería interesante la aplicación de los métodos propuestos a problemas clásicos de restauración de imágenes, como ruido aditivo y emborronamiento, a través de una formulación estadística del problema de aproximación rala, para buscar la solución del *Máximo A Posteriori* (MAP) del problema de restauración (como se ve, por ejemplo, en [78]).

English translation

Capítulo 8

Introduction

8.1. Introduction and objectives

Human brain has adapted throughout a long evolution, as well as during our personal development, to efficiently deal with visual stimuli [1]. So, there is a strong connection between the physical origin of those stimuli and the human visual system structure. In addition, vision is arguably the most powerful among human senses, in terms of the amount of information that it can acquire and process by time unit.

On the other hand, we have always had the need to transmit information to other people. To do it, we count on some tools limited to the precise place and moment where the message is emitted, as phonetic language. However, we also need to communicate to larger number of people, even though they are not present at the moment of emitting the message. This achievement was made first, and not by chance, through visual stimuli, as idiographic language or paintings. Some examples are Palaeolithic art and symbolic writings.

To obtain an image from the world around consists of projecting the three dimensional space we are living in onto a two dimensional surface, reproducing the shape of the objects and their details. Under the term *natural image* we denote those images captured, typically through photography, from the real world. This means that they are similar to the visual information usually captured by our eyes. Surprisingly enough, for the message to be interpreted properly, it is not necessary that those images are perfect projections of the real world. Indeed, our visual system is able to detect and recognise the represented objects even if they are distorted, up to a certain degree, naturally. This is a characteristic skill of very advanced visual systems which exploit in a massive way this kind of information.

Nowadays, we are witnessing an unprecedented technological revolution.

We are increasingly processing more and more information, which has to be transmitted to more and more people As a natural consequence of the dominant role of vision in our perception, one of the areas more affected by this revolution is that of digital images. In the last years, techniques for capturing, processing, transmitting and storing these images have developed beyond expected. In fact, digital images have already substituted analogical ones as the main representation vehicle. Thanks to the vast possibilities of digital technology, the image manipulation tools have exponentially increased. We keep on demanding more quality without compromising speed, and, therefore, a new effort is needed for improving the capture and the posterior processing. In addition, because of the greater importance of digital communications, it is also increasingly important to save bandwidth, and so it is sought to maximise the visual quality for a given information support (number of bits).

In the beginning, problems such as image coding for compression, enhancement, removing noise and annoying artifacts, lost information recovery, pattern recognition, etc., were approached heuristically, using more or less *ad-hoc* techniques. However, now there is no doubt about the importance of developing good image models allowing a more generic application to a wide variety of tasks.

Most of these applications are related to human vision. These kinds of tasks are carried out continuously in our brain. Therefore, to develop a good model it is convenient to pose the following question. How can our visual system discriminate the relevant information in a distorted image?

Obviously, an arbitrary image does not represent objects of our world, which have a typical structure allowing us to recognise them. This structure is reflected in natural images [1, 2], which consist, typically, of localised oriented features (edges, lines, corners, etc.) and relatively large smoothly varying areas, possibly with some texture in them. Left superior panel in the Figure 8.1 is an example of a typical natural image. On the other hand, random images, as that seen in the right superior panel in the Figure 8.1, do not have, in general, any structure at all. Due to the huge amount of visual stimuli processed, and learned to process through evolution, our brain is able to distinguish very clearly the original image underlying a degraded version of itself (see inferior panel in the Figure 8.1, formed as an additive mixture of the two panels above). Thus, if we want to automate this behaviour, it is very important to have good *a priori* knowledge of the typical structure of natural images, as many authors have pointed before (e.g., [3, 1, 4, 5]).

One of the criteria usually considered when evaluating the efficiency of a neural system is the maximisation of the ratio between the amount of information and the number of neurons required to represent it [5]. In a

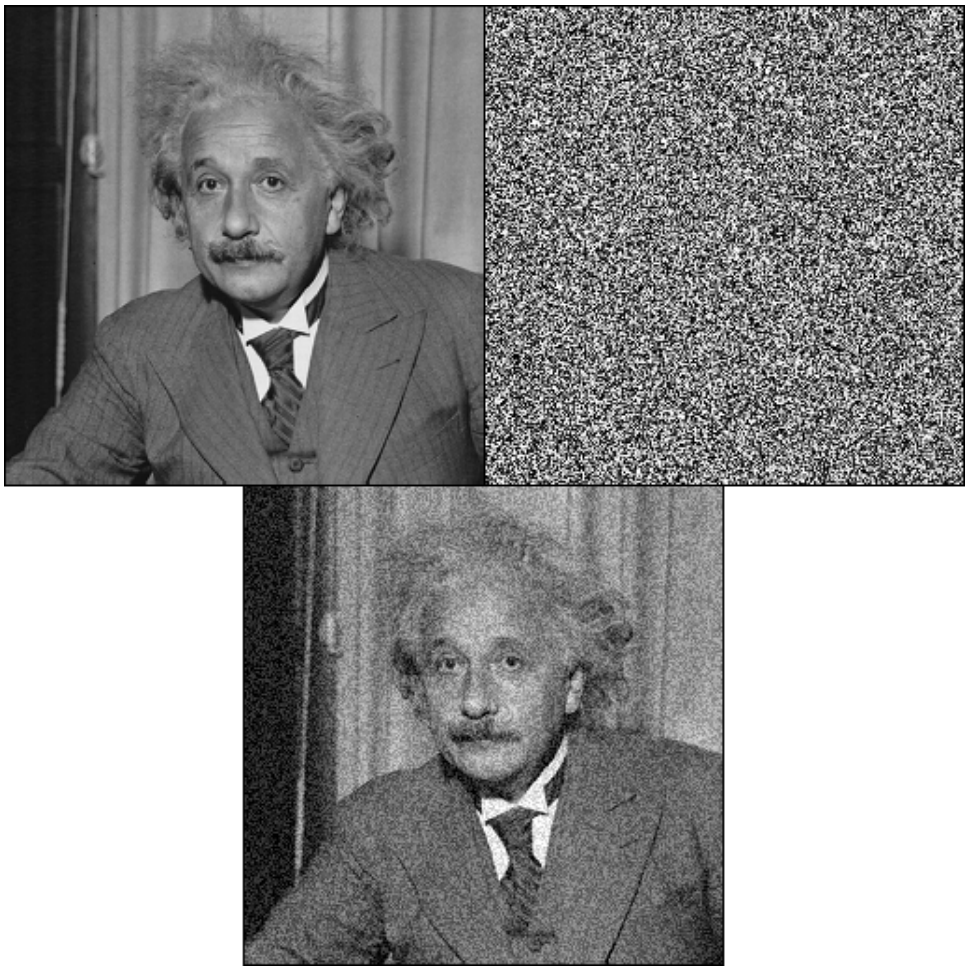

Figura 8.1: **Top left**, Einstein *standard image*. **Top right**, *random image (white Gaussian noise)*. **Bottom**, *sum of the two images above*.

similar way, we can pose the same goal when we deal with natural images. If we manage to represent them with as few numbers as possible, it will be easier not only to store them, but also to get more powerful statistical descriptions. As a consequence, we will also increase the performance in restoration tasks.

Unfortunately, despite their typical structure, the large size of the set of natural images and the strong statistical dependence between neighbouring pixels make the modeling too complex to be done in the pixel domain. The ability to accumulate information in few elements can be considerably powered by transforming the image from pixels to new domains. Let us come back to the example shown in the left panel of Figure 8.1. If we only take the 10 % of the pixels with largest deviation with respect to the global

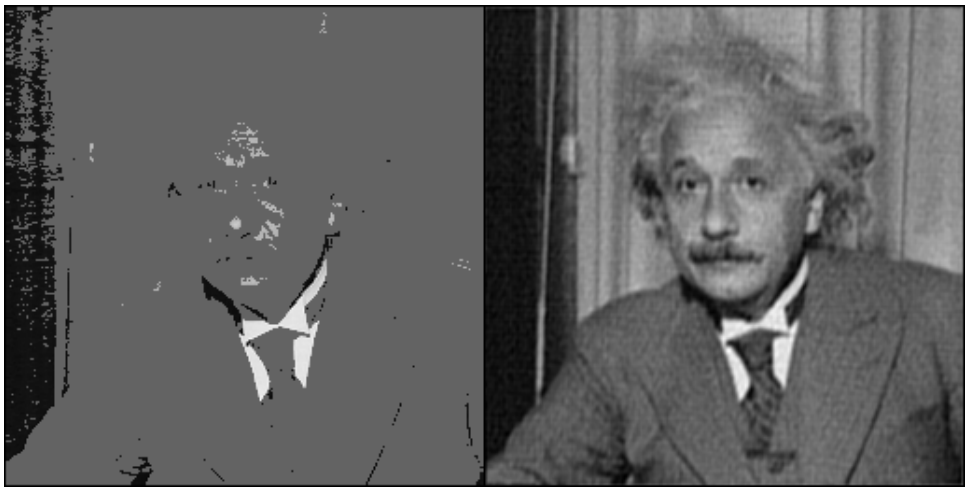

Figura 8.2: **Left**, *image obtained using the* 10 % *of pixels with largest deviation with respect to the global mean of the* Einstein *image.* **Right**, *image built using the* 10 % *of the coefficients with largest amplitude in the Fourier representation of the same image.*

mean, we obtain the image in the left panel of Figure 8.2, where, as we can see, most of the features of the original image have been removed. On the other hand, right panel of Figure 8.2 shows the image made by the 10 % of the frequencies in the Fourier domain whose coefficients are largest in amplitude. Although obtained using the same number of coefficients, the latter image is much closer to the original than the former one, both in objective and subjective terms.

This property also makes easier, as we have already mentioned, the statistical description of natural images. For example, they typically have large areas with smooth texture, and, thus, the energy of the Fourier representation is usually concentrated in the low frequencies. When degrading an image, if every frequency is uniformly corrupted (e.g., white Gaussian noise) then dominant frequencies are relatively less affected. Intuitively, we see that, by removing the lower amplitude frequencies, relatively very affected but with little significance to reconstruct the original image, we would greatly reduce the amount of noise in the observation, while maintaining a high fidelity to the original image. As a consequence, when most of the energy is concentrated in few coefficients, removing those with low amplitude will result in more important reduction of noise in the image than when the energy is more uniformly distributed.

The Oxford English Dictionary defines *sparseness* as the property of being thinly dispersed. The Cambridge Advanced Learner's Dictionary defines it as the property of being small in numbers or amount, often scattered over a large area. In this Thesis, we have made an extended use of

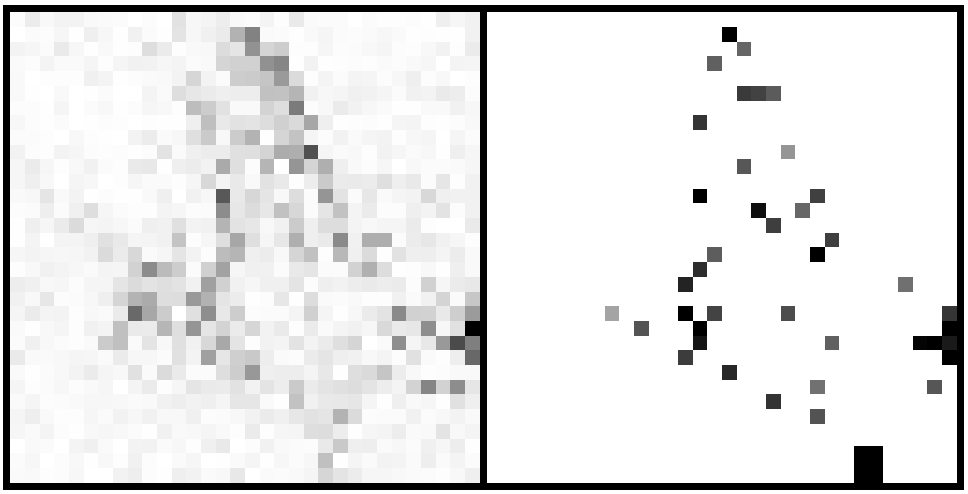

Figura 8.3: **Left**, *sub-band of the representation of a natural image under a wavelet-like (DT-CWT) filter bank. Dark pixels represent high amplitude coefficients and light ones those with low amplitude.* **Right**, *same sub-band of a sparse approximation to the natural image.*

this term, interpreting it as a continuous concept based on the concentration of most of the energy of a discrete signal in a relatively small proportion of coefficients. In order to measure the degree of energy concentration we use some norm of the representation vector. In the Figure 8.3 we can see two different two-dimensional distribution of coefficients. Whereas the right panel shows a *sparse* distribution, the left shares-out more the energy among the coefficients.

Apart from Fourier, there are other representations which further favour an efficient processing, making easier to describe the image statistics. For example, linear representations based on multi-scale pass-band filters, called generically *wavelets*, are specially well adapted to represent several properties of the natural images, as scale invariance and the existence of locally oriented structures. It has been experienced that the responses to this kind of filters of natural images typically produce sparse distributions [3, 6].

Redundant representations, those using more coefficients than pixels in the image, favor the extraction of relevant local features [7, 5], so they allow for a more powerful image analysis and processing compared to critically sampled ones. Also, being invariant to translation, rotation, phase, etc. [8, 9], they usually provide better results in image restoration (e.g., [7, 8]).

Nevertheless, direct transformation to a redundant domain does not necessarily increase the sparseness of the representation with respect to non-redundant ones (see, for example, [12], or Chapter 9 in this Thesis). We need, in addition, some non-linear methods to further increase the

sparseness. We define a dictionary as a set of vectors, also called atoms. The *sparse representation problem* is defined as finding the expression of an input signal as linear combination of as few elements as possible from a given representation. In this Thesis, we will tackle a common variant, called *sparse approximation problem*, that allows some tolerance in the fidelity to the input signal. Thus, we can obtain sparse representations with different approximation error, which is very useful for coding and restoration of images. The main issue that has traditionally prevented from solving these problems is their tremendous complexity [13], which has been considered, for a long time, as a too severe obstacle in practice. However, during the last years, and thanks to both a spectacular improvement in the computation speed and to increased consciousness of the importance of sparseness in signal processing, the methods for solving them have been object of big interest. Then, more or less efficient techniques to find sub-optimal solutions have been developed (e.g., [14, 13, 15, 16, 17, 18, 19]); mostly to use the sparseness as *a priori* knowledge for restoration purposes (e.g., [20, 16, 21, 22, 23]).

Current trends in these kinds of methods follow three main strategies. The oldest one tries to incrementally express the image by using greedy techniques, that is, sequentially using those vectors better approximating the part of the image still not represented (e.g., see [3, 24, 25]). The main problem of these techniques is that, being very inefficient, they often get trapped in non-favourable local minima in terms of energy compaction. There are other greedy techniques based on different selection strategies, as [26], where, in a previous work to this Thesis, we selected the significant coefficients by directly thresholding the observed representation vector.

The next strategy, by order of appearance, is based on reducing the complexity of the problem by changing the search for strict sparseness by the minimisation of the sum of the absolute values of the coefficients of the representation. This results in the minimisation of a convex function (see, e.g., [27, 13, 46]). In fact, this technique is called *convex relaxation*. Solving it provides, in general, a local solution to the sparse representation or approximation problems. The first approaches to this problem used numerical optimization algorithms, as conjugate gradient or interior-point methods (e.g., [27, 13, 28]). However, these are often inefficient, moreover when dealing with many dimensions, as typically happens in image processing.

Finally, during the last years, a number of efficient methods have been developed that are providing good practical results in both compaction and restoration problems. They are based on iteratively combining linear with shrinkage operations (see, e.g., [16, 30, 31, 18, 19]). In Chapter 9 we will

make a more extended review of the existing literature in this area.

In the literature, much effort has been devoted to find the constraints under which these two types of techniques manage to find the optimal solution to the sparse representation or approximation problems (see, among others, [32, 33, 34, 35]). But these conditions appear to be too restrictive for being accomplished in most practical image processing scenarios, where natural images, typical representations and useful sparseness levels are used. This has been already shown, for example, in [36, 18].

8.2. Contribution of this Thesis

In this Thesis we derive two iterative, though relatively efficient, methods to solve the sparse approximation problem. Besides trying to minimize the ℓ_0-quasi-norm[1] for a given approximation error, we have also developed the resulting version of both methods when using the sum of absolute values of the coefficients (ℓ_1-norm) as the criterion to be minimised, as proposed by the convex relaxation methods.

The first of the methods presented is based on reformulating the sparse approximation problem as finding, given p and R, the best approximation to the image inside the ℓ_p-ball of radius R. Our solution uses alternated orthogonal projections [37, 38] between the ℓ_p-ball and the set of vectors that represent perfectly the image. We have focused on the cases $p = 0$ and $p = 1$. Similar methods can be found in the literature, using both $p = 0$, where they have been derived heuristically [17], as well as $p = 1$ [39, 40]. We have called this method ℓ_p-AP.

The second one is based on re-expressing the cost function of the sparse approximation problem, which is discontinuous and unconstrained, to obtain a continuous and constrained equivalent version. By means of gradient descent in this new cost function, we obtain a generalised version of the Iterative Hard Thresholding (IHT) method [41]. This derivation allows us to prove that the fixed point of the iterations of this method is a local minimum to the sparse approximation problem. Next, we will show the proposed method, which consists of performing gradient descent over gradually less smoothed versions of the new cost function. This method, that we call ℓ_p-GM, has been used before [42, 17, 21]. Nevertheless, this is the first time that it is derived as the solution to an optimisation problem. We have also studied the counterpart of this method, ℓ_1-GM, which solves the convex relaxation problem.

[1] Though the ℓ_p-norm is not strictly a norm when $0 \leq p < 1$, in this Thesis we will usually use this term for every p value, for simplicity sake.

We have followed a very rigorous methodology for obtaining our methods as solutions to optimisation of well-defined standard criteria. However, instead of trying to guarantee that these methods will reach the global optimum under some given constraints, as other authors do (e.g., [32, 33, 34]), in this Thesis we have sought to obtain useful results in practical image processing conditions. Opposite to the extended assumption that the methods based on maximising the strict sparseness are intractable in practice, we have experienced that our sub-optimal methods based on minimisation of the ℓ_0-norm provide much better results than those based on minimising alternative convex criteria (as the ℓ_1-norm). Specially, ℓ_0-GM offers an excellent compaction performance, as other methods based on dynamically adjusting a threshold, and superior not only to our first method, but also to widely used greedy heuristics and convex relaxation techniques. In fact, we show that it has a nearly optimal asymptotic behaviour, when the number of active coefficients of the representation approaches the number of pixels of the image.

In addition, the interest of the proposed techniques is considerably increased when studying the application of the methods to different image restoration problems. We will show very high-quality results in a wide variety of applications, such as removing spatial quantisation artifacts (*de-quantising*), recovering missing pixels (*in-painting*), spatial-chromatic interpolation in digital camera mosaics (*de-mosaicing*), or static super-resolution. Up to our knowledge, this is the first time that this kind of methods are applied to de-quantising.

The content of this document is divided in the following chapters. In Chapter 9 the sparse approximation problem is stated, motivating it by the need for increasing the energy compaction achieved by linear transforms. The main traditional strategies for solving this problem are also analysed in detail. Next, Chapter 10 develops the first method proposed in this Thesis, ℓ_p-AP. We focus on the cases $p = 0$ and $p = 1$, and compare their performance one to each other and also with respect to other methods existing in the literature. In Chapter 11, we derive the IHT method and prove that its fixed point is a local minimum to the problem. Then, we obtain the second proposed method, ℓ_0-GM. We also show an analogous derivation for $p = 1$, resulting in the ℓ_1-GM method. In the results section we compare the behaviour of the proposed method versus the previous one and versus those existing in the literature. We also present the practical advantages of ℓ_1-GM compared to other methods solving optimally the convex relaxation problem. In Chapter **??** we see how to adapt our methods to restoration problems. Finally, in Chapter 13 we show the results of applying these methods to the restoration of several different degradations (see above).

Chapter 14 concludes this Thesis.

Capítulo 9

The Sparse Approximation Problem

The sparse approximation problem can be defined as minimising a measurement of the error when approximating an image as linear combination of a limited number of atoms taken from a redundant set (dictionary). In this work, we will measure this approximation error by means of the Mean Square Error (MSE). In this chapter we show that, in a redundant domain, there are infinite ways of representing an image. Traditionally, the minimum energy solution has been chosen, because it can be easily calculated (linearly). Nevertheless, non-linear methods have much higher potential to compact the energy in few coefficients. These kinds of methods have been extensively used and they are very useful for restoration purposes.

In Section 9.1 we motivate the use of non-linear methods to obtain sparse representations in redundant domains. In Section 9.2, we formulate the sparse approximation problem, and we describe the most important methods that has been used for solving it in Section 9.3. Finally, in Section 9.4 we analyse the conditions under which the sparse approximation problem can be solved optimally using convex optimisation, and same for greedy heuristics.

9.1. Analysis-based sparseness vs. Synthesis-based sparseness

Mathematically, to represent an image as a linear combination of vectors taken from some redundant set means to solve a system of linear equations with more equations than unknowns. There are, therefore, infinite solutions.

How to choose one of them? The minimum Euclidean norm solution has been traditionally chosen. Being a linear solution, it is easy to calculate it fast. In fact, usual "inverse"transforms used in image processing for representing images are minimizing the Euclidean norm (trough pseudo-inverse). However, as we have discussed in the Introduction, there are good reasons to look for solutions concentrating the energy in as few as possible coefficients. These solutions express the image, possibly with some error, as a linear combination of fewer vectors from the dictionary than the linear solution. Although some authors have observed that the linear response to wavelet filter banks of natural images already concentrates most of the energy in relatively few coefficients (e.g., [3, 43, 44, 6]), nevertheless the minimum Euclidean norm solution tends to spread as much as possible the energy among the coefficients, which makes it inadequate for sparse representation. Next we illustrate that the energy compaction in redundant dictionaries is much bigger for certain non-linear transformations of the image.

Figure 9.1 shows the fidelity (in dB) to the original image obtained by approximating it using different representations and a wide range of sparseness levels, that is, of number of vectors involved in the approximation. Data have been averaged (using the MSE) for the five images in our test set (see Appendix B). The fidelity to the original image is measured by the Peak Signal-to-Noise Ratio (PSNR), defined as $10 \cdot \log_{10}(\frac{\rho^2}{MSE})$ and which is measured in decibels (dB), and where ρ is the maximum value of the involved signals. In our case $\rho = 255$, because we are dealing with 8-bits monochromatic images. In this Thesis, we usually represent the number of active coefficients normalized by the total number of pixels in the image). Each curve in Figure 9.1 has been obtained by reconstructing the image using the largest coefficients (in amplitude) of each linear transform, for different sparseness levels. Three of the representations used are critically sampled (pixels, Fourier, Haar Wavelets) and the fourth one is redundant (Dual Tree Complex Wavelets or DT-CWT [45]).

We can see how the quality of the approximation for a given number of elementary functions is increased when we transform the pixels to the Fourier domain, and even more when we use critically sampled wavelets. Unfortunately, the performance of the linear redundant representation suffers a brisk fall, due to many coefficients responding to the same feature of the image, which results in a sparseness decrease. Then, as we have pointed out before, if we want to outperform the critically sampled wavelets, we need to use a non-linear vector selection mechanism. In Figure 9.1 we also show the result obtained using DT-CWT with the best method proposed in this Thesis (ℓ_0-GM, see Chapter 11). We can see that it provides a great

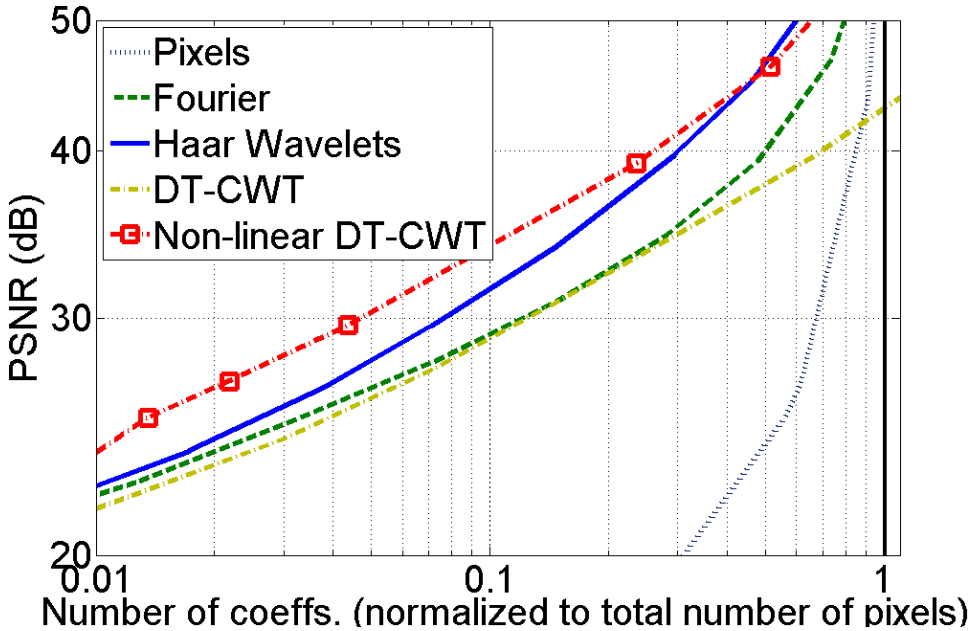

Figura 9.1: *Sparse approximation results comparison for several representations. Data have been obtained using the largest in amplitude coefficients of the linear responses and then averaging in our test set. Performance is measured in terms of PSNR given a number of coefficients. This number is normalised by the total number of pixels in the image.*

improvement over the approximations obtained using linear representations. The need for non-linear selection mechanisms to better concentrate the energy of the representation in few coefficients has been already treated by a number of authors, as, for example [28, 13, 46, 47, 48, 49].

Figure 9.2 illustrates the effect of using this type of selection mechanisms on the transform with DT-CWT of the *Peppers* image. Top left panel shows the coefficients of a sub-band of the linear representation of the image. In central panel we show the coefficients of the representation obtained non-linearly by maximising the sparseness (using ℓ_p-AP, see Chapter 10). In the former we can see a less sparse distribution of coefficients that in the latter, which strongly decreases simultaneous responses to the same feature. This better compaction makes possible that, using only a small proportion of the total number of coefficients (around 7 times less than pixels in the image, in this case) a high reconstruction quality is preserved (35,70 dB in this example, see right panel).

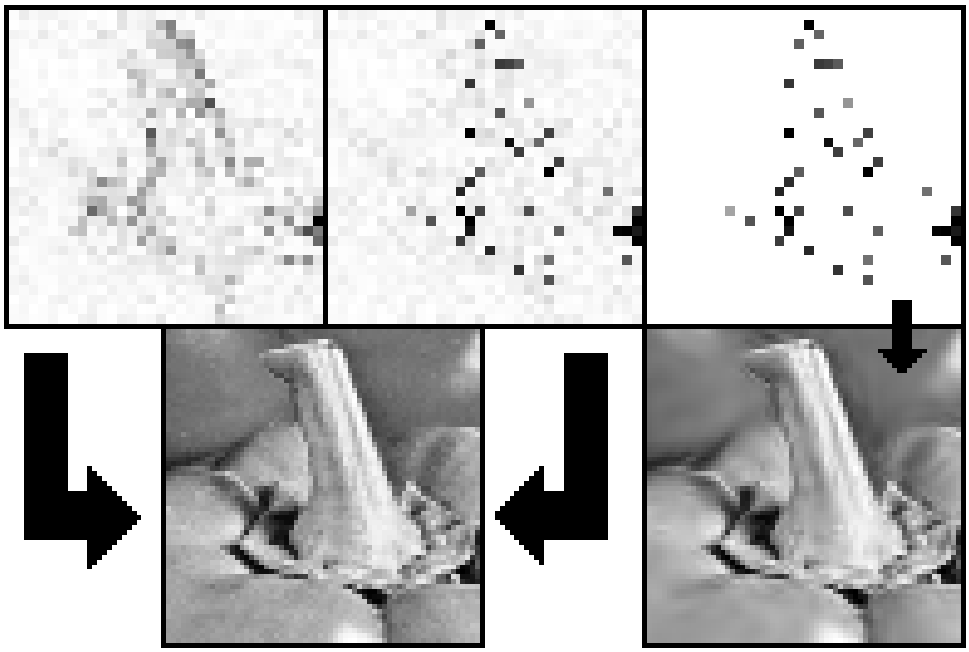

Figura 9.2: **Top-left**, *highest-frequency sub-band of analysis vector of* Peppers *image using DT-CWT with* 8 *scales. Light and dark points correspond, respectively, with low and high amplitudes of the coefficients. The size of the sub-band has been doubled in both directions through replication of rows and columns in order to match the size of the image, and then it has been cropped to* 64×64 *for visibility.* **Top-centre**, *same sub-band, but this time non-linearly obtained using the* ℓ_0*-AP method (see Chapter 10).* **Top-right**, *result of applying a threshold in amplitude to the result in central panel (preserving* 7 *times less coefficients than pixels in the image).* **Bottom-left**, *original image, which is perfectly reconstructed by the representations corresponding to the left and central panels of the top row.* **Bottom-right**, *approximation obtained with the sparse coefficients corresponding to the top-right panel (*35,67 *dB).*

9.2. Formulation of the sparse approximation problem

Next we formulate the sparse approximation problem. Let $\mathbf{\Phi}$ be a $N \times M$ matrix with $M > N$ and $range(\mathbf{\Phi}) = N$, representing the synthesis operator of a Parseval tight frame[1]. N is the number of pixels in the original domain and M the number of coefficients in the transformed domain. Then, for an observed image, $\mathbf{x} \in \mathbb{R}^N$, the linear system of equations:

$$\mathbf{\Phi a} = \mathbf{x}, \tag{9.1}$$

[1]A linear transform with a Parseval tight frame preserves the Euclidean norm of the original vector. We will use the term Parseval frame for simplicity.

has infinite solutions in $\mathbf{a} \in \mathbb{R}^M$. If we want to favour one of them, we should add some additional criteria. Thus, we can introduce a function $f(\mathbf{a})$ to discriminate this solution, and the problem is set out as:

$$\hat{\mathbf{a}}^f = \arg \min_{\mathbf{a} \in \mathbb{R}^M} f(\mathbf{a}) \text{ s.t. } \mathbf{\Phi a} = \mathbf{x}. \tag{9.2}$$

Among the possible options for $f(\mathbf{a})$, the p-th power of the ℓ_p-norm has often been used. For a given value of p, this is defined as $\|\mathbf{a}\|_p^p = \sum_{i=1}^{M} |a_i|^p$. In Figure 9.3 we show the shape of this function in its one-dimensional version and for several values of p. We have already mentioned that the most commonly used norm has been the Euclidean, $p = 2$, obtaining for it the minimum energy solution, $\mathbf{a}^{LS}$. This is specially easy to calculate for Parseval frames. In fact, $\mathbf{\Phi}^T = \mathbf{\Phi}^T[\mathbf{\Phi}\mathbf{\Phi}^T]^{-1}$ is the analysis operator of the Parseval frame, calculated as the pseudo-inverse of $\mathbf{\Phi}$, that is, $\mathbf{a}^{LS} = \mathbf{\Phi}^T\mathbf{x}$. However, as we have seen in the previous section, this is not an appropriate solution in terms of maximising the sparseness, which is measured using the ℓ_0-norm. This is expressed, by extension of the definition of norm, as the number of non-zero coefficients in the vector. Then, the sparse representation problem is expressed as:

$$\hat{\mathbf{a}}^0 = \arg \min_{\mathbf{a}} \|\mathbf{a}\|_0 \text{ s.t. } \mathbf{\Phi a} = \mathbf{x}. \tag{9.3}$$

However, redundant dictionaries typically used in image processing do not allow representations of natural images where most of the coefficients are exactly zero. It is more useful to search for representations concentrating most of the energy in as few as possible coefficients, so most of them have relatively small amplitudes. This kind of distributions, as we have exemplified in the previous section, have the property that a few high-amplitude coefficients can approximate the image with an error that can be acceptable for most applications. This is why many authors prefer to relax the constraint of Equation (9.3), and to formulate the sparse approximation problem as:

$$\hat{\mathbf{a}}^0(\lambda) = \arg \min_{\mathbf{a}} \{\|\mathbf{a}\|_0 + \lambda \|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}, \tag{9.4}$$

where $\lambda \in \mathbb{R}^*$ is a real positive number controlling the relative importance between the sparseness and fidelity terms in the cost function; so the higher its value, the smaller reconstruction error of the solution, whilst the sparseness is reduced. Note that Equation (9.4) is equivalent to either minimise $\|\mathbf{a}\|_0$ for a given quadratic error (which depends on λ) or to minimise the quadratic error for a given ℓ_0-norm of the approximation (which also depends on λ). Equation (9.3) is a particular case of

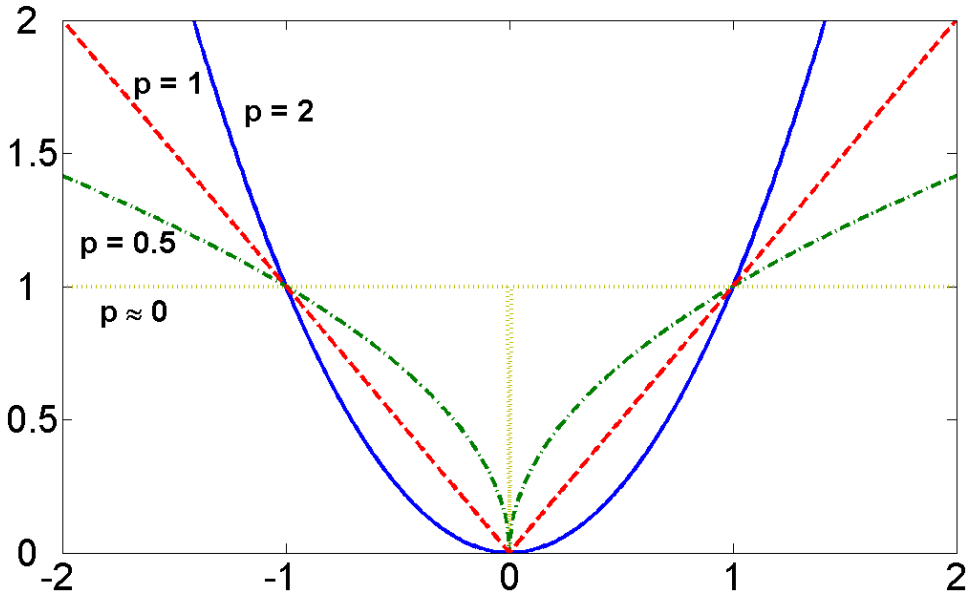

Figura 9.3: *One-dimensional p-th power of the ℓ_p-norm for different p values.*

Equation (9.4), when λ tends to infinite. As an example, the sub-band shown in the central panel of Figure 9.2 can be understood as belonging to the solution of (9.4) when λ is infinite, where the image is perfectly reconstructed but there is no sparseness. On the other hand, the sub-band in the right panel would correspond to the solution with a lower value of λ, using less coefficients but reducing the quality of the reconstruction of the image.

9.3. The sparse approximation problem in the literature

Finding the global optimum of the sparse approximation problem is a combinatorial problem, and, thus, NP-complex. It requires to look for every possible combination of columns of $\mathbf{\Phi}$, and solve for the least squares solution, choosing the one providing the lowest MSE. Some of the first approximations were restricted to some dictionaries (for example, wavelet packets or localised trigonometric functions) where it is possible to extract the orthonormal basis representing more efficiently the image [43]. But, during the last two decades, more effective and general methods to find local solutions to both problems (9.3) and (9.4) have become popular. We can classify these techniques in three main groups: greedy heuristics, methods

based on convex relaxation and methods based on iterative shrinkage. Next we will review the most important literature references in these three families.

9.3.1. Greedy heuristics

The first strategies to obtain local solutions to Equation (9.4) using general dictionaries were derived heuristically. Among them, the most frequently used comes from the observation that it is convenient, in order to approximate a given vector from a given a collection of vectors, start by selecting from the latter that vector having maximum correlation with the former (see, for example, [52]).

This family of algorithms is well known and widely used. In fact, these methods have been re-invented in several fields. In statistical modeling they are called *forward stepwise regression*, and they have been used since the 60s (see, for example, [53, 54] and references therein). When used in signal processing, they have been termed *Matching Pursuit* (MP) [14] and *Orthogonal Matching Pursuit* (OMP) [24], among others. In approximation theory they are referred as Greedy Algorithms [55, 56, 57, 58]. A wide review of these methods, applied to non-linear approximation, can be found in [59].

In our context, MP is the simplest greedy method. It is implemented through a set of indices, I, indicating the functions of the dictionary that have been already selected to form the approximation, and a residual, $\mathbf{r}$, which is the part of the image not yet represented. The set of indices is initialised empty, $I^{(0)} = \emptyset$, and the residual to the entire image, $\mathbf{r}^{(0)} = \mathbf{x}$. In each iteration $k+1$, the selected basis is updated by adding that vector having maximum correlation with the residual:

$$I^{(k+1)} = I^{(k)} \bigcup \{i : \langle \phi_i, \mathbf{r}^{(k)} \rangle \geq \langle \phi_j, \mathbf{r}^{(k)} \rangle \forall j \neq i, j \notin I^{(k)}\},$$

where $\langle \cdot, \cdot \rangle$ indicates inner product of two vectors. The estimation is updated then as:

$$\hat{\mathbf{x}}^{(k)} = \sum_{j=1}^{k} \langle \phi_{i(j)}, \mathbf{r}^{(j)} \rangle \phi_{i(j)},$$

where $i(j)$ represents the chosen index at i-th iteration. The next step is to update the residual as:

$$\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} - \langle \phi_i, \mathbf{r}^{(k)} \rangle \phi_i.$$

The iterations end either when the desired error level is reached, or when the desired number of terms has been selected. The main problem of MP

is that, given the set of elementary functions selected at each iteration, the amplitudes of the coefficients are not optimised to represent the image. OMP adds an intermediate least squares optimisation step. Given the subset of indices selected in iteration k, $I^{(k)}$, we define the $N \times k$ matrix $\mathbf{\Phi}_{I^{(k)}}$ formed by all columns ϕ_i from $\mathbf{\Phi}$ such that $i \in I^{(k)}$. Then, in all OMP iterations, additionally to the MP steps, the following problem is solved:

$$\hat{\mathbf{a}}_I^{(k)} = \arg \min_{\mathbf{a}_I \in \mathbb{R}^k} \|\mathbf{x} - \mathbf{\Phi}_{I^{(k)}} \mathbf{a}_I\|_2.$$

To update the residual we now use: $\mathbf{r}^{(k)} = \mathbf{x} - \mathbf{\Phi}_{I^{(k)}} \hat{\mathbf{a}}_I^{(k)}$. Not only this method provides better compaction results than MP, but it also converges faster [24]. However, as it often happens with greedy strategies, OMP gets stacked in local optima that frequently are not satisfactory for the problem at hand. A number of modifications have been proposed based mostly on recursive searches, sometimes also in hierarchical trees, trying to explore the maximum possible number of combinations at each step (see, for example, [60, 61, 62, 63]). Another disadvantage of OMP, even more serious in the variants mentioned, is that selecting only one coefficient at each step is unfeasible in terms of computation time for most commonly used dictionaries in image processing. There exist more efficient variants of OMP which selects more than one coefficient at each step, either with a fixed (e.g., [64]) or variable (e.g., [65, 25]) step size. In this Thesis, we refer to these methods with the name of maybe the more extended of them, Stage-wise OMP (StOMP) [25].

OMP has been applied with different degree of success to several applications, as for example noise removal [23, 66], video coding [67, 68, 69], colour image compression [70] or audio signals separation [71, 58]. However, all these applications either need few steps of the algorithm or they used orthogonal dictionaries. Only StOMP-like techniques can be applied, in most practical situations, with redundant dictionaries commonly used in image processing.

In a previous work to this Thesis [26], we presented a method for removing spatial quantization artifacts through finding a sparse approximation to the observation. The resulting method selects the significant coefficients via direct thresholding the amplitudes of vector $\mathbf{a}^{LS}$, which can be also seen as a greedy strategy.

9.3.2. Convex relaxation problem and *Basis Pursuit*

As we have already seen, using the Euclidean norm to obtain solutions (either approximate or exact ones) to the linear system of equations (9.1) is

not efficient in terms of obtaining sparse solutions. On the other hand, the ℓ_0-norm is not convex not even continuous, what makes the optimisation problem difficult to solve. This disadvantage, in practice, is not well solved by greedy heuristics. Can we find an intermediate way allowing to profit from the advantages of both approaches? In some cases the answer is provided by the ℓ_1-norm. In this case, $p = 1 < 2$, so it promotes sparse solutions; but, as it is convex, global optima are achievable in polynomial time. Note also that, as ℓ_1-based techniques optimise all the coefficients at the same time, we can reach approximation levels in practice that OMP cannot. This variant is termed *convex relaxation problem*, and it is formulated, analogously to Equation (9.4), as:

$$\hat{\mathbf{a}}^1(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}. \tag{9.5}$$

This problem has been frequently formulated, since the 50s, in terms of a linear program (LP). For example, in [27], a *Simplex* method is proposed to get minimum ℓ_1-norm solutions formulating the problem as an LP. Nevertheless, this technique was not systematically exploited until the microprocessor speed and the memory capacity of computers were boosted in the last three decades or so.

Basis Pursuit (BP) and *Basis Pursuit De-noising* (BPDN) methods [13, 46] respectively solve the convex representation and approximation problems through an LP equivalent to the Equation (9.5), by using Interior-Point Methods. Both BP and BPDN have had so much diffusion that are nowadays taken as synonymous for the convex relaxation problem.

The appearance of BP and BPDN was preceded by two significant advances: 1) the surprising discovery that we can nearly optimally estimate a piece-wise smooth function from a noisy observation, by solving a convex relaxation problem using the appropriate wavelet basis and a value of the λ parameter related to the variance of the noise [32]; and 2) the development of LASSO, which proposed the convex relaxation to solve the problem of subset selection in linear regression [73]. The technique known as *Least Angle Regression* [65] was later adapted to solve the LASSO formulation. The great disadvantage of this and other techniques (for example, FOCUSS [28]) is that they need to explicitly deal with the dictionary matrix. This is not practicable in most image processing scenarios, where the matrix often have hundred of thousands rows and columns. However, we can use very efficient techniques to solve the product of the big matrix corresponding to the representation used with a vector corresponding to the image. Both BP and iterative methods seen in the next section take advantage of these tools. A brief but complete history of the different approximations to the convex relaxation problem can be found in [74].

Apart from coding, the main areas where this kind of methods have been applied are statistical regression [73, 28] and noise removal [13, 46], although it is possible to find other applications are, for example, missing pixels recovery [75].

Recently, a great sense of expectancy has been generated with a new application for the convex relaxation problem, which is named *Compressed Sensing* (e.g., [76, 77]). This technique is based on the observation that a relatively small number of random projections of a sparse signal suffice to recover from them a good approximation of that signal [78]. It has been proposed as a powerful alternative to the traditional Nyquist Sampling Theorem, when the signal can be expressed as linear combination of few elements.

It is also possible to use intermediate quasi-norms ($0 < p < 1$) when regularising the problem (9.2) (see, for example, [79, 16]). Although these norms do not lead to convex cost functions, and therefore the global optimum is difficult to calculate, several authors have pointed the fact that the marginal distribution of the coefficients of the linear transform of a natural image under a wavelet filter bank is appropriately modeled using these norms (see, for example, [80, 20]). This approach to the sparse approximation problem is very interesting, but its study is beyond the scope of this Thesis.

9.3.3. Iterative shrinkage

We have already said that application of greedy methods to sparse approximation of natural images is not completely satisfactory, because they are too expensive in computational terms and they get trapped in unfavourable local optima. On the other hand, methods based on a direct approach to the convex relaxation problem require too much computation for natural images of usual size.

During the last years new efficient techniques have been developed in order to solve the problem (9.4). They are based on iteratively apply shrinkage operations (coefficient-wise) combined with linear projections. By shrinkage we understand those operations that decrease the amplitude of the coefficients in the representation, possibly setting to zero the ones below and amplitude threshold. We call these techniques *Iterative Shrinkage Methods*[2].

The iterative shrinkage methods have been widely used. For example, [81] already used similar techniques for image segmentation. In [8] it is proved that, in presence of observations degraded by additive white Gaussian noise, the simple method of applying a (hard or soft) thresholding

[2]Other possible name is *Thresholded Landweber Iteration*.

to an orthogonal linear transform of the image achieves, for some signal models, optimal results in terms of MSE.

The optimality of these shrinkage operations depends on the orthogonality of the dictionary matrix[3]. But a single shrinkage, by itself, does not lead to optimal solutions when using redundant transforms. Even though, it has been widely used in all types of redundant representations (see [82] and references therein). Shrinkage, within an iterative scheme, can provide very efficient and reasonable good approximate solutions.

More recently, other authors [80, 83, 84] simplified the ideas in [72, 8] and re-stated the problem as in Equation (9.2), starting form a Bayesian formulation to find the *Maximum A Posteriori* (MAP). The Iterative Shrinkage Method, applied frequently to solve this problem, can be described with the following iterations:

$$\mathbf{a}^{(k+1)} = S_p\left(\mathbf{a}^{(k)} + \mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{a}^{(k)}), \theta\right), \tag{9.6}$$

where $S_p(\mathbf{a}, \theta)$ indicates a certain shrinkage operation on a vector $\mathbf{a}$ with threshold θ, which in our case is function of the parameter λ. The linear operation that is the argument of the shrinkage comes from minimising the fidelity term in the Equation (9.2), for a given vector $\mathbf{a}$, and it is the orthogonal projection of the vector onto the affine subspace of perfect reconstruction (see Section 10.1).

Up to our knowledge, the first method proposing an iterative thresholding technique to solve problem (9.2) was [79, 16], which use ℓ_p-norms with $0 < p \leq 1$. The method is derived from a formulation based on *Expectation-Maximisation* (EM). Later on, in [30], the same method was derived in the case $p = 1$, using extra terms added to the cost function without changing its minima. Also [82] derives a similar algorithm using $p = 1$, but from a different perspective, because it tries to maximise the sparseness of the representation. The method used in all these works is based on alternating a soft thresholding with a linear projection. It is commonly named *Iterative Soft-Thresholding* (IST). It corresponds to iterate with Equation (9.6) using soft-thresholding, defined as $S_1(\mathbf{a}, \theta) = \mathbf{b}$, where:

$$b_i = \begin{cases} sign(a_i) \cdot (|a_i| - \theta), & |a_i| > \theta \\ 0, & |a_i| \leq \theta. \end{cases} \tag{9.7}$$

Here, $sign(\cdot)$ indicates the sign function. The value of the threshold results in $\theta = \frac{1}{2\lambda}$. Convergence of the method was proven in [30, 39]. These techniques have also been used to separate the different morphological components of a signal, using several dictionaries, each having the property of having sparse

[3]It also depends on using the Euclidean norm as error term.

response to a different family of input signals (Morphological Component Analysis, MCA) [48, 21]. Other authors have derived similar methods from different points of view (e.g., [85]). See [86] for a review on iterative shrinkage-based methods.

Other iterative algorithms exist, as the one described in [87] using soft thresholding as a gradient descent operation, and using a linear search to find the optimal step size for each iteration. Another example is [83], that applies soft thresholding in redundant representations within a variational formulation to remove noise and for compression; or the application of the generic optimisation technique called *Iterative Re-weighted Least Squares* [88], which reformulates the Equation (9.5) as a quadratic programming problem (requiring direct manipulation of the transform as a large-scale matrix). We want to emphasise, among them, the gradient descent method recently proposed in [40]. This method is based on the same type of operations than IST, but the fixed threshold is changed by a threshold adapted in each iteration to keep constant the ℓ_1-norm. This method is also derived in Chapter 10 of this Thesis, when using the ℓ_1-norm. In addition to this, it applies an optimised step size of the gradient descent to accelerate convergence.

An alternative method to IST can be derived using hard-thresholding instead, which is known as Iterative Hard-Thresholding (IHT). In this case, the operation used in the iterations (9.6) is $S_0(\mathbf{a}, \theta) = \mathbf{b}$, with:

$$b_i = \begin{cases} a_i, & |a_i| > \theta \\ 0, & |a_i| \leq \theta. \end{cases} \tag{9.8}$$

And the value of the threshold results in $\theta = \lambda^{-\frac{1}{2}}$. The first paper (heuristically) proposing such a method was [15]. It is also heuristically proposed in other works, as [41]. In [29] it has been derived using surrogate functions, and, moreover, its convergence to a local minimum of the sparse approximation problem has been proved[4].

In [17] a similar heuristic is proposed, but instead of applying a fixed threshold, the number of non-zero coefficients after each hard-thresholding operation is fixed. This is a similar method to what we derive in Chapter 10 of this Thesis, but here we derive it as solution to an optimisation method. This modification provides better compaction results. A further improvement on these methods can be found in [89], and uses an adaptive threshold depending on the energy of each sub-band of the representation.

Alternative solutions to soft and hard-thresholding have been proposed. For example, Firm-Shrinkage [90] tries to improve the results obtained,

[4]This work is parallel to our derivation of the method and proof of convergence proposed in Chapter 11 and already published in [12].

in [72], using both thresholding types. In [91, 92] a variational formulation for IST, IHT and Firm-Shrinkage is shown.

Several authors have compared the performance of both soft and hard-thresholding. Unless exceptional cases, such as [84], most of them have experienced that hard outperforms soft-thresholding [84, 36, 93, 94, 95].

Many authors have experienced a great improvement in the general performance of this type of algorithms when the threshold is decreased at each iteration (dynamic thresholds). This is one of the ideas involved in the proximal-points methods [42]. These methods solve iteratively a succession of problems formulated using the Equation (9.5) with increasing values of λ. The dynamic version of IST is found in the MCA variant [21, 22, 96] but no theoretical justification is provided. Also heuristically, [40] proposes to increase dynamically the radius of the ℓ_1-ball where each soft-thresholding operation is projected. With respect to heuristic versions of dynamic IHT, they can be found in [97, 15, 17]. [98] presents a method based on substituting the ℓ_0-norm by a equivalent continuous function. They use a Gaussian function leading to an algorithm different from ℓ_0-GM.

Despite their recent introduction, methods based on iterative shrinkage have already proved to be very powerful for a number of applications. For example, [22] uses IST and IHT to recover missing pixels in the image. Also several papers use IST to approach the classic restoration problem (blurring plus noise, e.g., [31, 99, 100]). Moreover, other applications can be found, as medics imaging [101] or video coding [102]. Finally, we want to emphasise the application of this type of techniques to Compressed Sensing [103, 78].

9.4. Equivalence conditions when minimising ℓ_1 and ℓ_0-norms

In previous sections we have reviewed the most common techniques to solve the sparse approximation problem of Equation (9.4). We have seen that the global solution cannot be found in practice, and that three main approaches have been proposed: greedy heuristics, convex relaxation methods and iterative shrinkage methods. Next question is: Do these methods offer equally good solutions? In this section we review some surprising results proving that, under certain conditions, both greedy methods (OMP) and convex relaxation methods reach the global optimum to the sparse *representation* problem, and provide a solution with an error proportional to the level of noise when dealing with the sparse *approximation* problem.

In the foundation of these results we find the concept of mutual coherence

of a matrix. This is defined as $M(\mathbf{\Phi}) = \sup\{\langle \phi_i, \phi_j \rangle; \forall i \neq j\}$. There is also a stronger constraint, associated to a different measure of the richness of a dictionary, called $Spark(\mathbf{\Phi})$ or Kruskal range. It is defined as the minimum number of matrix columns forming a linear dependent set. It has been stated the following relation between $Spark(\mathbf{\Phi})$ and mutual coherence: $Spark(\mathbf{\Phi}) \geq \frac{1}{M(\mathbf{\Phi})}$.

The first step to establish equivalence conditions between problems (9.4) and (9.5) was to prove that, if a solution is sparse enough, it is the only global optimum to the problem (9.3) (see [104, 105] and also [106, 107, 108, 109, 110]). These results are interesting because they allowed for the first time to have a simple way to check if solutions obtained with different methods were optimal or not. The condition to check was established as $\|\hat{\mathbf{a}}^0\|_0 < \frac{Spark(\mathbf{\Phi})}{2}$. But more general results were still missing, because there was no known method to obtain effectively the global optimum.

It has been empirically shown in [13, 46] (using small 1-D standard discrete functions) that the solution to the convex relaxation problem is sparser than the minimum Euclidean norm solution. Defining the mutual coherence, for two matrices of equal size, as $M(\mathbf{\Phi}_A, Phimtx_B) = \sup\{|\phi_a; \phi_b| : \phi_a \in \mathbf{\Phi}_A, \phi_b \in \mathbf{\Phi}_B\}$, in [105] it is proven that, if the solution to the sparse representation problem holds that $\|\hat{\mathbf{a}}^0\|_0 < \frac{1}{2}(1 + \frac{1}{M(\mathbf{\Phi})})$, when $\mathbf{\Phi}$ is formed by concatenating two mutually incoherent dictionaries (those leading to a small value of $M(\mathbf{\Phi}_A, \mathbf{\Phi}_B)$), then $\hat{\mathbf{a}}^0$ is the unique global solution and it can be obtained through the minimisation of the ℓ_1-norm. Later on, [111] improved this upper bound setting it to $\|\hat{\mathbf{a}}^0\|_0 < \frac{0{,}9142}{M(\mathbf{\Phi})}$. This result can be also extended to redundant dictionaries [112, 113, 114]. In these works the uniqueness upper bound for general dictionaries were decreased to $\|\hat{\mathbf{a}}^0\|_0 < \frac{Spark(\mathbf{\Phi})}{2}$, what is twice tighter than the upper bound for the convex relaxation methods to reach the global optimum of the sparse representation problem. Finally, [32] relaxed the conditions to prove that, if a signal has a representation with fewer than τN non-zero coefficients, where $\tau > 0$ is a real proportionality factor, then the solution to the convex relaxation problem is equal to the sparse representation problem. However, it is not made clear how to calculate ρ for each dictionary.

In most practical situations it is not reasonable to assume that the observed coefficients perfectly represent the signal. Then, it is more interesting the scenario where an ideal signal has a sparse approximation, but we only observe a version degraded with white additive noise. [34] studies the algorithms based on convex relaxations in the same conditions that [112, 113, 114]. Under these sparseness bounds, and if the dictionary has the property of being mutually incoherent, then the convex relaxation algorithms are globally stable. That is, the error made is proportional to

the noise level even under an arbitrary amount of noise. It is also shown that, under certain conditions, the support of the results of these methods is contained within the ideal selection existing for the original signal. Similar results were also derived in [114, 74]. We refer to [49] to find a more complete review of these works.

On the other hand, in [33, 34] it is shown that greedy techniques, such as OMP, find the global solution in the same conditions as BP for the sparse approximation problem, with the difference that OMP is locally stable. That is, under a small quantity of noise we can recover the ideal sparse representation with an error that increases, at worst, proportionally to the noise level. However, in [115, 116] it is shown that, in practice, OMP gets better results, and it is also faster. We refer to [116] to see these results in detail. Nowadays, there are no similar results stating the conditions under which the iterative shrinkage methods reach the global optimum to the sparse representation or approximation problems.

Capítulo 10

Sparse approximation using alternating projections

In this chapter we present a simple and robust optimisation non-linear method providing sub-optimal solutions to the sparse approximation problem (Equation (9.4)). It is based on, given a Parseval frame transforming the pixels of the image to a redundant transformed domain, and given values for parameters p and R, look for the vector with ℓ_p-norm equal to R which best approximates the image, in terms of the MSE in the reconstruction. The method consists of applying alternating orthogonal projections[1] onto the set of vectors of the transformed domain with ℓ_p-norm equal or less than R and the set of vectors representing perfectly the image. We call this method ℓ_p-AP (for *Alternated Projections*). We show that it converges to the global optimum of the cost function when $p \geq 1$, and to a local optimum if $0 \leq p < 1$. Here we will focus on the cases $p = 0$ and $p = 1$. We will show that, even being sub-optimal, ℓ_0-AP clearly outperforms ℓ_1-AP (which is equivalent to other Basis Pursuit-like methods). We will also see how to de-bias the coefficients of the solution given by ℓ_1-AP, through a LS-optimization of the active coefficients. We obtain slightly better compaction results than those of ℓ_0-AP. Finally, we will see that ℓ_0-AP also outperforms other existing strategies, as greedy heuristics and iterative shrinkage methods based on using fixed thresholds. Previously, [17] had already proposed a heuristical method equivalent to ℓ_0-AP, whilst a very similar method to ℓ_1-AP appears in [40], developed simultaneously and independently to our work.

In Section 10.1 we describe the method ℓ_p-AP, explaining in detail the particular cases $p = 0$ and $p = 1$. In Section 10.2 we explain a method

[1]Here we use the term *orthogonal projection* in a wide sense, involving any minimum Euclidean distance projection.

that, given a set of indices, find the vector whose support is that set and best approximates the image. After that, we describe the implementation details in Section 10.3, to discuss the results of the compaction experiments in Section 10.4. Section 10.5 concludes this chapter.

10.1. ℓ_p-AP method

For clarity sake, we start rewriting the problem of approximation minimising a general ℓ_p-norm:

$$\hat{\mathbf{a}}^p(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_p^p + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}, \tag{10.1}$$

where $\|\mathbf{a}\|_p = (\sum_{i=1}^{M} |a_i|^p)^{\frac{1}{p}}$ denotes the ℓ_p-norm of $\mathbf{a}$. If we assume a determined value of λ, then $\hat{\mathbf{a}}^p(\lambda)$ will have a determined ℓ_p-norm, that we note[2] $R(\lambda)$. Then, to solve the Equation (10.1) for a given λ value is equivalent to minimise the approximation error for a given norm of the solution, $\|\hat{\mathbf{a}}^p(\lambda)\|_p = R$:

$$\hat{\mathbf{a}}^p(\lambda) = \hat{\mathbf{a}}^p(R) = \arg\min_{\mathbf{a}\in\mathbb{R}^M} \|\mathbf{\Phi a} - \mathbf{x}\|_2^2 \text{ s.t. } \|\mathbf{a}\|_p^p = R. \tag{10.2}$$

An ℓ_p-ball with radius R, centred at the origin, is formed by all those vectors with ℓ_p-norm less or equal than R, $B_p(R) = \{\mathbf{a} \in \mathbb{R}^M : \|\mathbf{a}\|_p^p \leq R\}$. Then, we solve here the problem:

$$\hat{\mathbf{a}}^p(R) = \arg\min_{\mathbf{a}\in B_p(R)} \|\mathbf{\Phi a} - \mathbf{x}\|_2. \tag{10.3}$$

Although, strictly talking, $\hat{\mathbf{a}}^p(R) \neq \hat{\mathbf{a}}^p(\lambda)$ (because those vectors with lp-norm lower than R are feasible), in practice, as we will see forward, we use a method that provides, under certain conditions, solutions holding the constraint of the optimization in Equation (10.2).

Equation (10.3) can be solved using several techniques. We have chosen to use the Alternating Projections Method [37, 38] due to its simplicity and convergence properties. This method consists of orthogonally alternately projecting onto two or more sets until reaching convergence. When the involved sets are convex and they have intersection, the method converges to the orthogonal projection of the input vector onto the intersection of the sets. When they are convex but they have empty intersection, the method converges to a limit cycle of minimum distance between both sets. When one or more of them are non convex, the limit cycle is reached in a local

[2]Trough this Thesis, we remove the dependency of R on λ for clarity sake.

minimum of that distance[3]. See [117] for a more complete discussion of the convergence properties when non-convex sets are involved.

In order to apply the Alternating Projections Method, we have to define two sets. First of them is the set of solutions to the Equation (9.1), defined as $S(\mathbf{\Phi}, \mathbf{x}) = \{\mathbf{a} \in \mathbb{R}^M : \mathbf{\Phi a} = \mathbf{x}\}$. It is an affine sub-space of $\mathbb{R}^M$, and, thus, it is convex. The second set is the ℓ_p-ball of radius R, centred at the origin, $B_p(R)$, for a given p and R values. This set is convex only if $p \geq 1$. Here we assume that the starting vector for the iterations has a ℓ_p-norm larger than desired (as it happens in practice), which implies that the solution will lie on the boundary of the ℓ_p-ball and, as $B_p(R)$ is a closed set, the optimization constraint of Equation (10.2) holds.

We denote $P_C^{\perp}(\mathbf{v})$ to the orthogonal projection of a vector $\mathbf{v}$ onto a given set C. The orthogonal projection of $\mathbf{a}$ onto the affine sub-space $S(\mathbf{\Phi}, \mathbf{x})$ can be obtained easily, being:

$$P_{S(\mathbf{\Phi},\mathbf{x})}^{\perp}(\mathbf{a}) = \mathbf{a} + \mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi a}). \tag{10.4}$$

This result can be interpreted in terms of adding to the vector $\mathbf{a}$ the difference between the minimum Euclidean-norm solution, $\mathbf{a}^{LS} = \mathbf{\Phi}^T\mathbf{x}$, and the analysis vector of the reconstruction using $\mathbf{a}$ ($\mathbf{\Phi}^T\mathbf{\Phi a}$).

On the other hand, the orthogonal projection onto $B_p(R)$, $P_{B_p(R)}^{\perp}(\mathbf{a})$, depends, obviously, on the value of p. Next, we explore in detail the cases $p = 0$ and $p = 1$.

Finally, the ℓ_p-AP method is implemented through the following iterations:

$$\begin{aligned} \hat{\mathbf{a}}^p(R)^{(0)} &= P_{B_p(R)}^{\perp}(\mathbf{a}^{LS}), \\ \hat{\mathbf{a}}^p(R)^{(k+1)} &= P_{B_p(R)}^{\perp}(P_{S(\mathbf{\Phi},\mathbf{x})}^{\perp}(\hat{\mathbf{a}}^p(R)^{(k)})). \end{aligned}$$

We have chosen to stop the iterations when $\|\hat{\mathbf{a}}^p(R)^{(k+1)} - \hat{\mathbf{a}}^p(R)^{(k)}\|_2 < \delta$ for a $\delta > 0$ (see implementation details in Section 10.3). Now we study the $p = 0$ and $p = 1$ cases in detail.

10.1.1. ℓ_0-AP

10.1.1.1. Projection onto the ℓ_0-ball of given radius

When $p = 0$, it is straightforward to derive that $P_{B_0(R)}^{\perp}(\mathbf{a})$ is a hard-thresholding preserving the R largest coefficients in amplitude:

$$P_{B_0(R)}^{\perp}(\mathbf{a}) = \mathbf{a}^h,$$

[3]In this case, it can eventually happen that the orthogonal projection onto the non-convex sets is not unique, but this is a theoretical problem without practical consequences.

where:

$$a_i^h = \begin{cases} a_i, & |a_i| > \theta_h(\mathbf{a}, R) \\ 0, & |a_i| \leq \theta_h(\mathbf{a}, R). \end{cases}$$

Here, $\theta_h(\mathbf{a}, R)$ is the lowest threshold between those preserving the $R - n_0$ largest coefficients in amplitude, being n_0 the smallest non-negative integer guarantying that a solution exists. Thus, $n_0 = 0$ if there are no repeated amplitudes in the interval of interest. Following the previous definition, in practice the threshold is set to the amplitude of the $R + 1$-th largest coefficient in amplitude in vector $\mathbf{a}$.

This method can be also seen as a particular case of the method described in [17], but with the difference that in that paper the method was not formally justified as an optimisation method.

10.1.1.2. ℓ_0-AP scheme and convergence

Top panel of Figure 10.1 shows an illustration of ℓ_0-AP using few dimensions ($N = 2$, $M = 3$, $R = 1$).

We will prove next that this method converges to a local minimum, in the image domain, of the MSE of the reconstruction for the vectors of the ℓ_0-ball. Substituting following the Equation (10.4):

$$\|\mathbf{a} - P_{S(\mathbf{\Phi},\mathbf{x})}^{\perp}(\mathbf{a})\|_2 = \|\mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{a})\|_2 = \|\mathbf{x} - \mathbf{\Phi}\mathbf{a}\|_2, \tag{10.5}$$

where the last step is true because $\mathbf{\Phi}^T$ is a Parseval frame. Given that $\hat{\mathbf{a}}^0(R)$ is a local minimum in $B_0(R)$ of the distance to $S(\mathbf{\Phi}, \mathbf{x})$, then it exists a $\delta > 0$ such that for all $\mathbf{a} \in B_0(R)$, if $\|\mathbf{a} - \hat{\mathbf{a}}^0(R)\|_2 < \delta$, then $\|\mathbf{a} - P_{S(\mathbf{\Phi},\mathbf{x})}^{\perp}(\mathbf{a})\|_2 \geq \|\hat{\mathbf{a}}^0(R) - P_{S(\mathbf{\Phi},\mathbf{x})}^{\perp}(\hat{\mathbf{a}}^0(R))\|_2$. Using (10.5) we have that $\|\mathbf{x} - \mathbf{\Phi}\mathbf{a}\|_2 \geq \|\mathbf{x} - \mathbf{\Phi}\hat{\mathbf{a}}^0(R)\|_2$. That is, $\hat{\mathbf{a}}^0(R)$ is a local minimum in vector $\mathbf{a}$ and within set $B_0(R)$ of the Euclidean distance between $\mathbf{\Phi}\mathbf{a}$ and $\mathbf{x}$.

Regarding the convergence properties, we have observed that the method evolves fast towards the solution during the first iterations, and then the convergence speed decreases, as shown in Figure 10.2 for images *Barbara*, *Boat* and *House* from our test set (see Appendix B). In the figure we see that the convergence speed also depends on the degree of sparseness imposed (the sparser, the faster). In this Thesis, we are interested in exploring the performance of the methods at convergence, and this has required making thousands of iterations for each experiment. However, in a practical implementation, less iterations can be made for obtaining satisfactory results. In this work, we have established the stopping criterion based on the PSNR increase every 10 iterations. Dotted curve corresponds to the increase rate used as tolerance (it would be a straight line if the figure

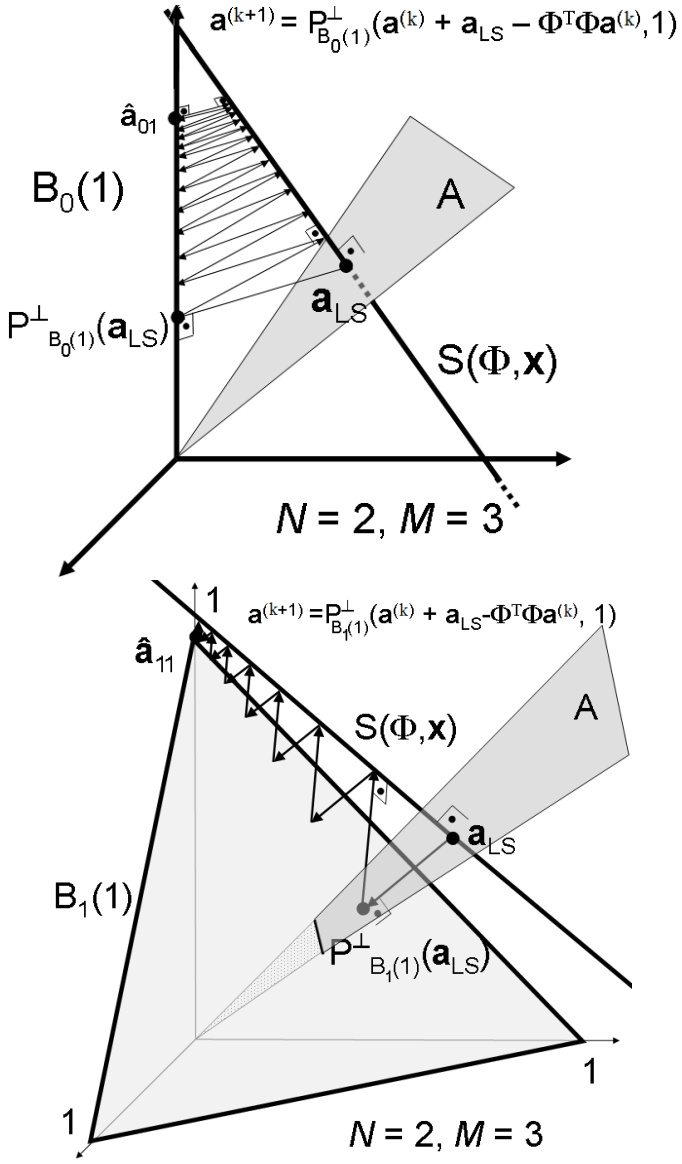

Figura 10.1: **Top**, *graphical explanation of the ℓ_0-AP method.* **Bottom**, *same for ℓ_1-AP. Only a face of the ball is shown for clarity.*

was not in logarithmic coordinates). In subsection 10.3.2 we will see more details about the stopping criterion.

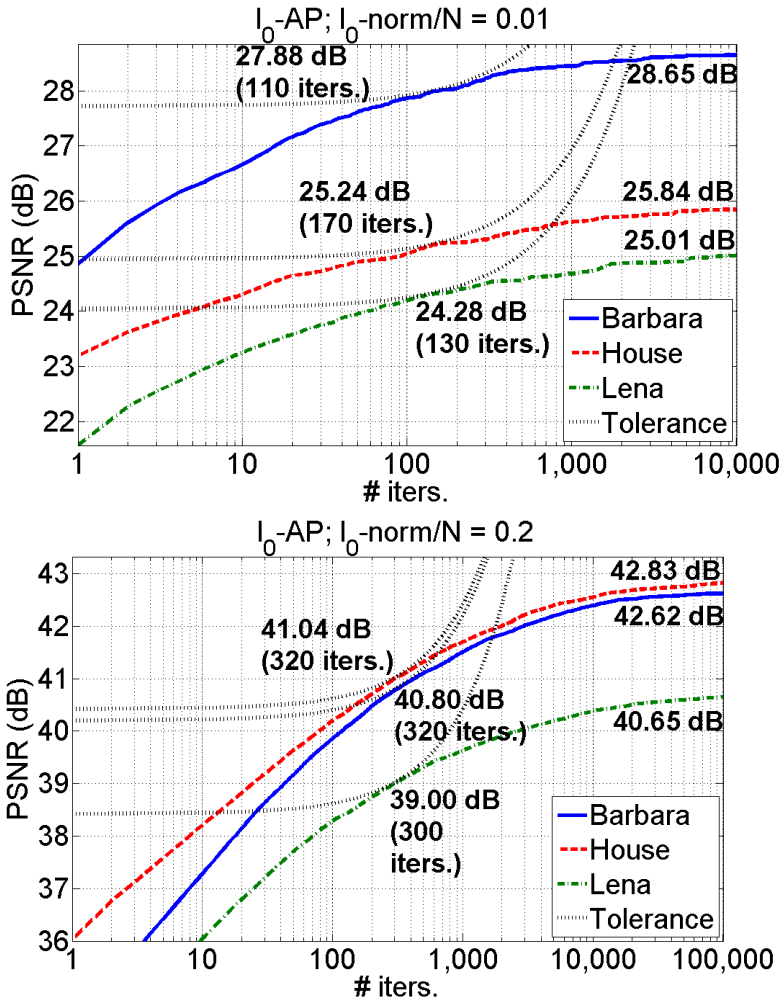

Figura 10.2: *Logarithmic plot of the approximation quality (PSNR, in dB) vs. the number of iterations for ℓ_0-AP for three images and two sparseness levels. The representation used here is DT-CWT. The number at the end of the curves is the PSNR at convergence. The numbers accompanying the tangency point (indicated by the intersection with the dotted curves) are the PSNR and the number of iterations obtained when the stopping criterion is reached.*

10.1.2. ℓ_1-AP

10.1.2.1. Projection onto the ℓ_1-ball of given radius

When $p = 1$, it can be proved that the orthogonal projection of a vector $\mathbf{a}$ onto the ℓ_1-ball of given radius R, that we denote as $\mathbf{a}^s = P^{\perp}_{B_1(R)}(\mathbf{a})$, is a soft-thresholding operation. This has been proved before, for example in [39, 40], but our alternative proof will also provide an iterative method to find the threshold associated to this operation.

We first assume that $\|\mathbf{a}\|_1 > R$ (otherwise the projection onto $B_1(R)$ would be the identity). In addition, we use the obvious fact that any projection onto a ℓ_p-ball preserves the sign of the coefficients in the original vector ($sign(\mathbf{a}^s) = sign(\mathbf{a})$). The problem is then reduced to project the vector formed by the components $\{|a_1|, |a_2|, ..., |a_M|\}$, that we denote $\mathbf{a}^{abs}$, onto the positive hyper-quadrant of $B_1(R)$. Once we have that projection, we restore the sign of each element to obtain the projection of $\mathbf{a}$ onto $B_1(R)$.

The positive hyper-quadrant of $B_1(R)$ can be defined as the intersection of two convex sets. The first one is the set of all those vectors whose components sum, as much, R:

$$F(R) = \{\mathbf{b} \in \mathbb{R}^M : \sum_{i=1}^{M} b_i \leq R\}.$$

Given a vector $\mathbf{c} \in \mathbb{R}^M$, the expression of the orthogonal projection onto this set is:

$$P^{\perp}_{F(R)}(\mathbf{c}) = \mathbf{c} - \delta,$$

where $\delta = \frac{\sum_{i=1}^{M}(c_i)-R}{M}$ if $\sum_{i=1}^{M} c_i > R$ and 0 otherwise.

The second set is the positive hyper-quadrant of the M-dimensional vector space:

$$G^+ = \{\mathbf{b} \in \mathbb{R}^M : \forall\ i = \{1, ..., M\}, b_i \geq 0\}.$$

The orthogonal projection onto this set is defined as:

$$P^{\perp}_{G^+}(\mathbf{c}) = \mathbf{D}\mathbf{c},$$

where $\mathbf{D}$ is a diagonal $M \times M$ matrix, such that $d_{ii} = 1$ if $c_i > 0$ and 0 otherwise.

Following the alternated projections theory, the orthogonal projection of a vector $\mathbf{a}$ onto the intersection of $F(R)$ and G^+, that we name $\mathbf{a}^{pro} = P^{\perp}_{F(R)\bigcap G^+}(\mathbf{a})$, is defined as:

$$\mathbf{a}^{pro} = \lim_{n\to\infty}[P^{\perp}_{G^+}(P^{\perp}_{F(R)}(\cdots n \cdots P^{\perp}_{G^+}(P^{\perp}_{F(R)}(\mathbf{a}^{abs}))\cdots n \cdots))], \quad (10.6)$$

The orthogonal projection of $\mathbf{a}$ onto $B_1(R)$ is finally obtained as:

$$\mathbf{a}^s = sign(\mathbf{a}) \cdot \mathbf{a}^{pro}. \quad (10.7)$$

We prove next that the expression obtained in Equation (10.7) is a soft-thresholding. First, for $k = \{1, ..., n\}$, we denote $\delta^{(k)}$ the term substracted in

the k-th application of the orthogonal projection onto $F(R)$ in the iterations of Equation (10.6). We also denote $\mathbf{D}^{(k)}$ the mask applied in the k-th application of the orthogonal projection onto G^+. Then:

$$\mathbf{a}^{pro} = \lim_{n\to\infty}[\mathbf{D}^{(n)}(\cdots\mathbf{D}^{(2)}(\mathbf{D}^{(1)}(\mathbf{a}^{abs} - \delta^{(1)}) - \delta^{(2)})\cdots - \delta^{(n)})].$$

This can be expressed as:

$$\mathbf{a}^{pro} = \mathbf{a}^{abs} - \mathbf{d}, \tag{10.8}$$

where each element d_i is defined as:

$$d_i = \begin{cases} \theta_s(\mathbf{a}, R), & |a_i| > \theta_s(\mathbf{a}, R) \\ |a_i|, & |a_i| \leq \theta_s(\mathbf{a}, R), \end{cases}$$

and where $\theta_s(\mathbf{a}, R) = \sum_{k=1}^{n} \delta^k$. Consequently, if we substitute the expression of Equation (10.8) into Equation (10.7), we obtain:

$$\mathbf{a}^s = sign(\mathbf{a}) \cdot (\mathbf{a}^{abs} - \mathbf{d}),$$

which is the definition of a soft-thresholding. That is, $\mathbf{a}^s = S_1(\mathbf{a}, \theta_s(\mathbf{a}, R))$ (see Equation (9.7)).

Note that this proof provides a method based on alternating projections for, given $\mathbf{a}$, finding the value of the threshold that leads to the desired value of the ℓ_1-norm after the shrinkage. This method starts by removing the sign from $\mathbf{a}$, projecting the result onto the intersection of $F(R)$ and G^+, by using alternating projections, and finally restoring the original sign to every element of that projection. In practice, this method converges linearly in very few iterations. Next we develop a method where each iteration requires less calculation, so the final method is easier to implement.

Firstly, let's express the ℓ_1-norm of the projected vector (that is, R) as a function of the threshold[4] $\theta_s(\mathbf{a}, R)$. For that purpose, we define the set of indices corresponding to those coefficients in $\mathbf{a}$ whose amplitudes are above a threshold θ: $\Upsilon(\mathbf{a}, \theta) = \{i \in \{1, ..., M\} : |a_i| > \theta\}$. Then, we can write:

$$\begin{aligned} R &= \sum_{\Upsilon(\mathbf{a},\theta_s)} (|a_i| - \theta_s) \\ R &= \left(\sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i|\right) - card(\Upsilon(\mathbf{a}, \theta_s)) \cdot \theta_s, \end{aligned}$$

[4]For notation clarity, in the following derivation we have removed the dependency of θ_s upon $\mathbf{a}$ and R.

where $card(\cdot)$ indicates the cardinality of a set. We can express the previous iterations as:

$$\theta_s = \frac{\left(\sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i|\right) - R}{card(\Upsilon(\mathbf{a},\theta_s))}. \tag{10.9}$$

Note that the right term depends on θ_s, but nevertheless we can solve this equation iteratively using the following iterations:

$$\begin{aligned} \theta_s^{(0)} &= 0, \\ \theta_s^{(k+1)} &= \frac{\left(\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i|\right) - R}{card(\Upsilon(\mathbf{a},\theta_s^{(k)}))}. \end{aligned} \tag{10.10}$$

The iterations end when $\|\theta_s^{(k+1)} - \theta_s^{(k)}\|_2$ is below a tolerance threshold (see subsection 10.3.2 for more details on the stopping criterion).

Next, we prove that iterations (10.10) converge to θ_s. We note first that $R(\theta_s)$ is a strictly decreasing function and, then, so it is $\theta_s(R)$. This implies that Equation (10.9) has an unique solution in θ_s. If we find $\theta_s^{(k+1)} = \theta_s^{(k)}$ then that value holds Equation (10.9), so we know that if the iterations converge then they do to the unique solution, θ_s. Thus, to prove the convergence to θ_s, it is left to prove that the succession $\theta_s^{(k)}$ converges. This can be made by proving that 1) $\theta_s^{(k)}$ is monotonically increasing, and that 2) it is upper bounded by θ_s. This is what we do next.

We start by observing that $\theta_s^{(0)} = 0 \leq \theta_s$. Assuming that $\theta_s^{(k)} \leq \theta_s$, then:

$$\sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} |a_i| \leq \sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} \theta_s,$$

where $\Gamma(\mathbf{a},\theta_1,\theta_2) = \{i \in \{1,...,M\} : \theta_1 < |a_i| \leq \theta_2\}$. From here we obtain that:

$$\begin{aligned} \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - \sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i| &\leq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s - \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s, \\ \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\leq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s, \\ \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\leq card(\Upsilon(\mathbf{a},\theta_s^{(k)})) \cdot \theta_s, \\ \frac{\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R}{card(\Upsilon(\mathbf{a},\theta_s^{(k)}))} &\leq \theta_s, \\ \theta_s^{(k+1)} &\leq \theta_s. \end{aligned}$$

Now we see that, as $\sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} |a_i| \geq \sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} \theta_s^{(k)}$, and as $\sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s \geq \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s^{(k)}$, then:

$$\sum_{\Gamma(\mathbf{a},\theta_s^{(k)},\theta_s)} |a_i| + \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s \geq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s^{(k)},$$

and we have the following inequalities:

$$\begin{aligned}
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - \sum_{\Upsilon(\mathbf{a},\theta_s)} |a_i| + \sum_{\Upsilon(\mathbf{a},\theta_s)} \theta_s &\geq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s^{(k)}, \\
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\geq \sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} \theta_s^{(k)}, \\
\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R &\geq card(\Upsilon(\mathbf{a},\theta_s^{(k)})) \cdot \theta_s^{(k)}, \\
\frac{\sum_{\Upsilon(\mathbf{a},\theta_s^{(k)})} |a_i| - R}{card(\Upsilon(\mathbf{a},\theta_s^{(k)}))} &\geq \theta_s^{(k)}, \\
\theta_s^{(k+1)} &\geq \theta_s^{(k)}.
\end{aligned}$$

Consequently, the succession is monotonically increasing, so the proof is complete.

10.1.2.2. ℓ_1-AP scheme and convergence

Figure 10.1 (bottom panel) illustrates the behaviour of ℓ_1-AP with $N = 2$, $M = 3$, and $R = 1$. Only a face of $B_1(1)$ is shown for visibility sake.

It is easy to prove that ℓ_1-AP provides the global minimum for the distance, in the image domain, from the reconstruction from vectors in $B_1(R)$ to image $\mathbf{x}$. We first note that $\hat{\mathbf{a}}^1(R)$ is the global minimum in $B_1(R)$ of the Euclidean distance to $S(\mathbf{\Phi}, \mathbf{x})$ (because both sets are convex). Then, for every $\mathbf{a} \in B_1(R)$, we have that $\|\mathbf{a} - P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\mathbf{a})\|_2 \geq \|\hat{\mathbf{a}}^1(R) - P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\hat{\mathbf{a}}^1(R))\|_2$. Applying the Equation (10.4) and being $\mathbf{\Phi}^T$ a Parseval frame, we obtain that $\|\mathbf{x} - \mathbf{\Phi}\mathbf{a}\|_2 \geq \|\mathbf{x} - \mathbf{\Phi}\hat{\mathbf{a}}^1(R)\|_2$. That is, $\mathbf{\Phi}\hat{\mathbf{a}}^1(R)$ is the global minimum, for all $\mathbf{a} \in B_1(R)$, of the Euclidean distance of $\mathbf{\Phi}\mathbf{a}$ to $\mathbf{x}$.

Figure 10.3 illustrates the convergence properties of ℓ_1-AP. The interpretation is similar to that of Figure 10.2. As there are no local solutions to avoid, the convergence is more regular that using ℓ_0-AP and fewer iterations are required to converge. We have included an example, in the bottom panel, where perfect reconstruction is achieved.

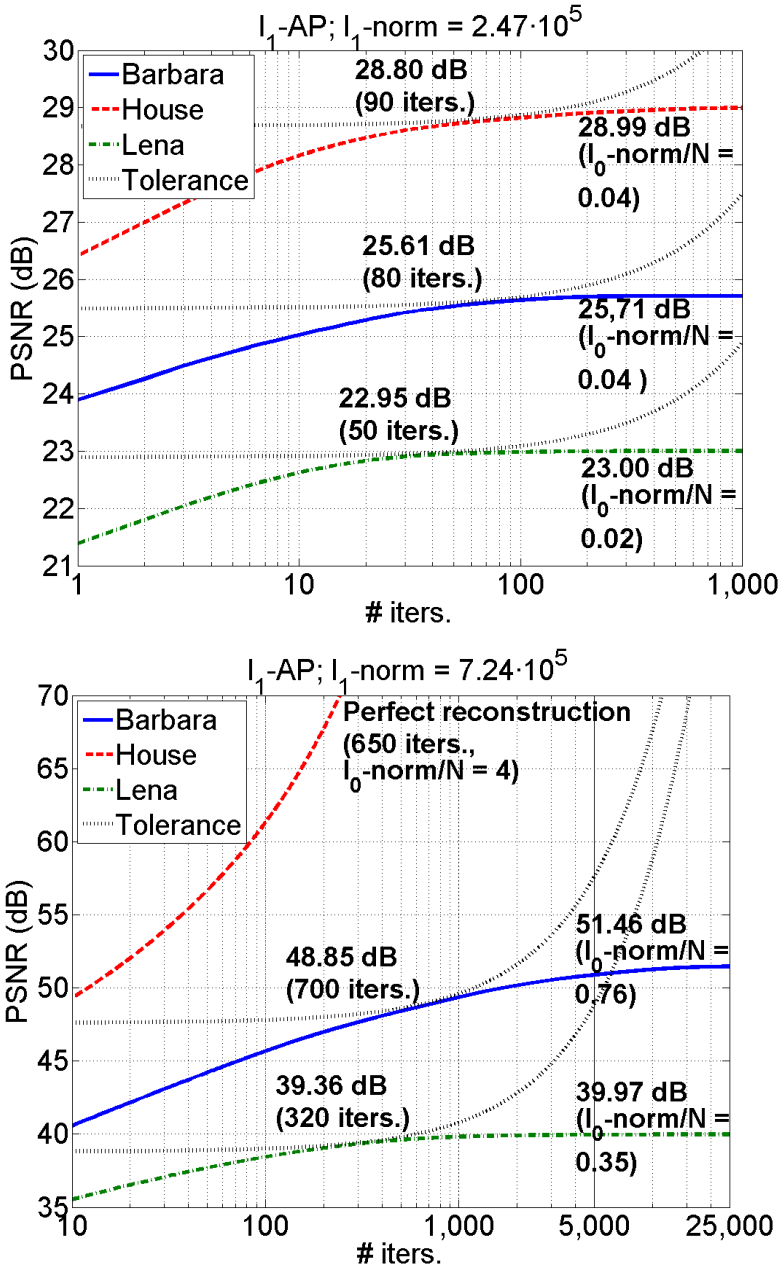

Figura 10.3: *Convergence curves in semi-logarithmic scale for ℓ_1-AP, using three images and two sparseness levels. Details are similar to Figure 10.2. It is also indicated the ℓ_0-norm, normalised by N, of the solution at convergence.*

10.2. Mean square error minimisation for a given selection of coefficients

As ℓ_0-AP iterations are performed, the selection of coefficients becomes more and more stable, and the final solution becomes LS-optimal for

that selection[5]. As it was already pointed in [17], when, in the limit, the number of active coefficients becomes fixed, the two involved sets (vector subspace generated by the selected atoms and affine subspace of perfect reconstruction) are convex, and so the iterations converge to the global optimum, for that selection, linearly.

However, this is not true when we use a generic ℓ_p-norm, because the projection onto the ℓ_p-ball is not, in general, a hard-thresholding operation. As our final target is to solve the sparse approximation problem of Equation (9.4), we should use some in order to improve the quality of the approximation for a given set of selected functions. We use here a method based on alternating projections which has been previously used by a number of authors, as [13, 15, 26, 34, 100].

Given a set I of R indices extracted from $\{1, ..., M\}$, we define $\mathbf{\Phi}_I$ as a $N \times R$ matrix formed by columns ϕ_i of $\mathbf{\Phi}$ such that $i \in I$. Then, we want to find:

$$\hat{\mathbf{a}}_I = \arg \min_{\mathbf{a}_I \in \mathbb{R}^R} \|\mathbf{\Phi}_I \mathbf{a}_I - \mathbf{x}\|_2,$$

which translates into $\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^\sharp \mathbf{x}$, being $\mathbf{\Phi}_I^\sharp$ is the pseudo-inverse of $\mathbf{\Phi}_I$. Note that $\mathbf{\Phi}_I^\sharp = \mathbf{\Phi}_I^T [\mathbf{\Phi}_I \mathbf{\Phi}_I^T]^{-1}$ if $R > N$ and $\mathbf{\Phi}_I^\sharp = [\mathbf{\Phi}_I^T \mathbf{\Phi}_I]^{-1} \mathbf{\Phi}_I^T$ if $R \leq N$. When dealing with images, the size of $\mathbf{\Phi}_I$ makes the calculation of the pseudo-inverse a completely intractable task. Instead, we follow these iterations:

$$\begin{aligned} \mathbf{a}^{(0)} &= \mathbf{D}_I \mathbf{\Phi}^T \mathbf{x}, \\ \mathbf{a}^{(k+1)} &= \mathbf{D}_I [\mathbf{a}^{(k)} + \mathbf{\Phi}^T (\mathbf{x} - \mathbf{\Phi} \mathbf{a}^{(k)})]. \end{aligned} \tag{10.11}$$

where $\mathbf{D}_I$ is a diagonal $M \times M$ matrix such that $d_{ii} = 1$ if $i \in I$ and 0 otherwise. In Appendix D we show that this method effectively solves for the pseudo-inverse in $\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^\sharp \mathbf{x}$.

10.3. Implementation

10.3.1. Representations

To test the methods, we initially used four different Parseval frames (DT-CWT [45], Curvelets [118], Steerable Pyramid [7] and a redundant version of the Haar Wavelets [119]). From them, we chose those two giving the

[5]Note that the method is still suboptimal because the selection of elementary functions is not optimal in general.

best averaged compaction performance. These are DT-CWT and Curvelets[6]. Redundancy factor for DT-CWT is 4, and for Curvelets is $\approx 7{,}2$.

In order to make a homogeneous treatment of the two representations, we have divided DT-CWT coefficients in real and imaginary parts. On the other hand, to optimise the approximation in the extremely high sparseness range, we have inserted, in both representations, an extra scale composed only by the global mean of the image. This way we adapt to the fact that, usually, the best approximation to a natural image, using only one coefficient, is the global mean.

MATLAB® code for DT-CWT is available in [120]. We have also used MATLAB® code for the Curvelets implementation (*CurveLab 2.0* [121]).

10.3.2. Convergence and stopping criterion

The stopping criterion for ℓ_p-AP is traduced, in our implementation, in using two tolerance constants. The first one controls the PSNR increase in order to decide if convergence has been reached. The method stops when the increase after 10 iterations is below 0,02 dB. This stopping criterion is represented as dotted curves in Figures 10.2 and 10.3. These curves would be straight and tangents to the convergence curves if the horizontal axis was plotted in a linear scale. We have experienced that this criterion, typically, provides differences with respect to the PSNR at convergence below 1 dB in the high sparseness range and below 2 dB in the low sparseness range. These differences are even lower for ℓ_1-AP (favouring, thus, to this method in their comparison).

Note that, if the radius of the ℓ_p-ball is big, the method achieves perfect reconstruction of the image. In this case, the increase of PSNR is, as stated by theory, linear. To detect this situation we have used a second tolerance criterion, controlling the PSNR increase after every 10 iterations and stopping them when the difference between the last two increases is below a constant (10^{-6} for ℓ_0-AP and 10^{-4} for ℓ_1-AP).

We optimise the threshold in each ℓ_0-AP iteration through a golden search. This requires an extra tolerance parameter controlling the size of the search interval, which we have set to 0,1. In ℓ_1-AP, we have used the method described in subsection 10.1.2, taking for the stopping criterion the difference between the desired radius for $B_1(R)$ and that obtained after each iteration. We have experienced that, eventually, this iterative method provides exactly the radius required. However, to reduce computation in practice, we have also chosen 0,1 for this tolerance.

[6]The experiments were actually made with all the representations, and all the results are qualitatively similar to those presented here.

10.4. Results and discussion

In the following experiments we compare the compaction performance of our methods with respect to some reference algorithms in the field. Results have been collected for a wide sparseness range, and for our set of standard test images. We have used PSNR to measure the approximation error. We have used a logarithmic scale for the vertical axis, though PSNR is already a logarithmic measure. This may be unusual, but we think that it is justified, in this case, because of the great improvement achieved in the visualisation. Regarding the sampling of the curves, each marker corresponds to averaging the results of the corresponding method for all test images, and the intermediate values have been linearly interpolated.

10.4.1. Some previous methods

Our first experiment compares some widely used sparse approximation strategies. We have the following two objectives: a) compare Iterative Shrinkage Methods, with fixed threshold, in their two variants: hard and soft-thresholding; and b) compare the direct and accumulative strategies for selecting coefficients in greedy methods.

Regarding the former, we have implemented the methods IHT and IST, as described in subsection 9.3.3. We remind that these methods iterate between a thresholding operation and the projection onto the affine space of perfect reconstruction (Equation (10.4)), using a fixed threshold. We have used the same stopping criterion as in our implementation of ℓ_p-AP (see subsection 10.3.2). Thus, our implementation of these methods only differs from our implementation of ℓ_p-AP in using a fixed threshold instead of a fixed number of non-zero coefficients after each thresholding.

To compare greedy heuristics, we have implemented StOMP [25] and the method we presented in [26], called here DT+OP (from Direct Thresholding plus Least Squares-Optimisation). To choose the threshold used by StOMP, we previously set how many coefficients will be selected at each iteration. On the other hand, DT+OP applies the threshold directly (and only once) upon the linear representation of the image for each sample. Both methods use Equations (10.11) to LS-optimise the quality of the reconstruction after the thresholding. Here, we have also used the same stopping criteria as for ℓ_p-AP.

Figure 10.4 shows graphically some numerical results of this experiment. Top panel shows fidelity results using 8-scale DT-CWT, and bottom one using 6-scale Curvelets. This figure shows that the compaction performance of IHT is, on average, clearly better than IST for medium-high levels of sparseness. In low levels, the number of local minima is considerably

increased and therefore IHT has more probabilities of getting trapped in non-favorable local minima. Figure also shows that the results obtained with DT+OP outperform our implementation of StOMP, except in the very high sparseness range (of little practical relevance). This indicates that directly selecting the coefficients is better than accumulating them by means of the correlation with the residual. Among the compared methods, IHT provides the best results. Previously, some authors have also pointed that hard thresholding outperforms soft one when compacting the energy [84, 93, 94, 18], but no careful and systematic comparisons using natural images were presented.

10.4.2. Comparison of ℓ_0-AP, ℓ_1-AP and previous methods

This second experiment compares ℓ_0-AP to ℓ_1-AP. Because the result of ℓ_1-AP is not LS-optimal for the selection of atoms from the dictionary, we also compare with the result of de-biasing these coefficients with Equations (10.11). We label this method as ℓ_1-AP+OP. In addition, we have included IHT and StOMP as representatives of the iterative shrinkage and greedy methods, respectively.

Figure 10.5 shows the result of this experiment. Top panel shows fidelity results with 8-scale DT-CWT, and bottom one with 6-scale Curvelets. We can see that ℓ_0-AP clearly outperforms ℓ_1-AP, even though the latter is optimally minimising the ℓ_1-norm for each sparseness level. We also see that ℓ_1-AP+OP improves drastically the results of ℓ_1-AP, providing slightly better results than ℓ_0-AP. This shows that the selection of coefficients made by ℓ_1-AP is slightly better, in general, than that of ℓ_0-AP, specially in the low sparseness range. This, as before, seems to be a natural consequence of ℓ_0-AP getting trapped in non-favourable local minima, whose number rapidly increases when the sparseness gets low.

We can also see that ℓ_0-AP significantly improves the results of IHT and StOMP. It is interesting to note that fixing the radius of the ℓ_p-ball provides much better results than fixing the threshold.

Tables 10.1 and 10.2 show the results plotted in Figure 10.5.

Figure 10.6 compares visually the methods using the *Einstein* image[7] using $0{,}0765 \cdot N$ Curvelets coefficients. From top to bottom, left column shows the original image, the result of ℓ_1-AP (30,85 dB) and that of ℓ_1-AP+OP (33,52 dB). Note the great visual improvement obtained when

[7]For every experiment in this Thesis using *Einstein* image, we have removed its black border, by replication of adjacent rows and columns. This makes it a more representative natural image.

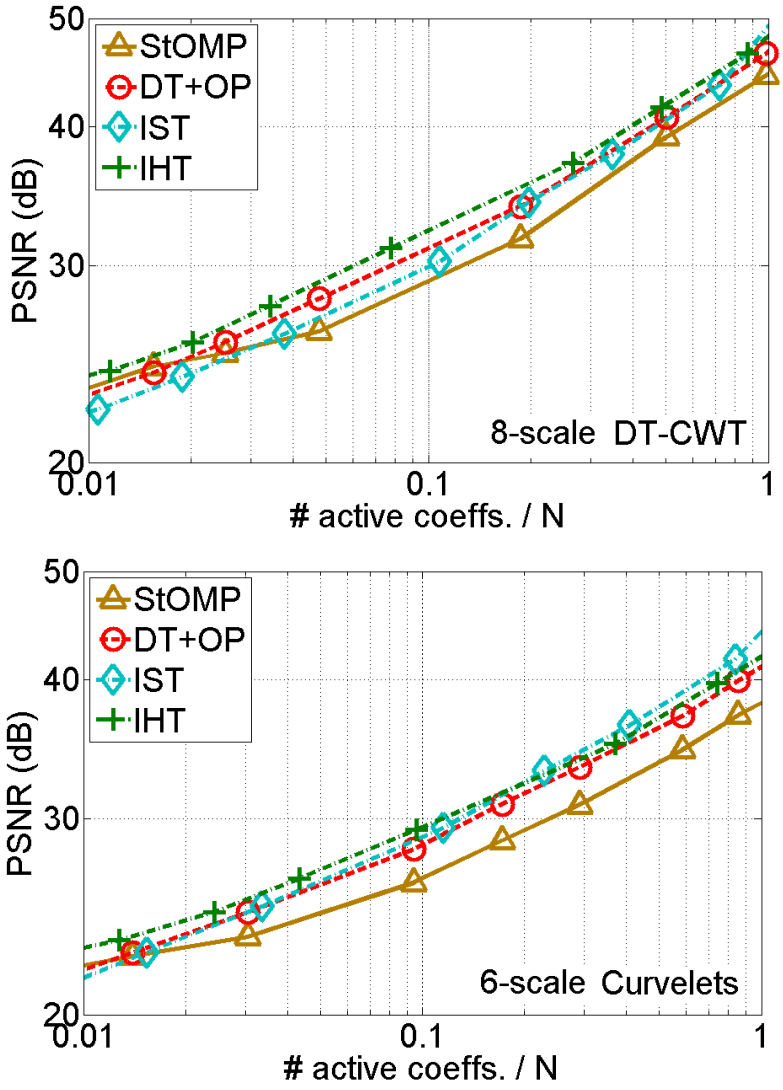

Figura 10.4: *Averaged compaction results (fidelity of the approximation as PSNR, in dB) for our test set using StOMP, DT+OP, IHT and IST.* **Top**, *using 8-scale DT-CWT.* **Bottom**, *using 6-scale Curvelets.*

post-optimising the selected coefficients. Right column shows StOMP (28,66 dB), IHT (29,10 dB) and ℓ_0-AP (32,98 dB). Although more than half dB below in terms of PSNR, there is no significant visual difference between ℓ_0-AP and ℓ_1-AP+OP in this example. Not shown here, this difference becomes even smaller for lower PSNR values.

As we have already pointed, ℓ_0-AP is equivalent to [17] when a fixed number of coefficients, and no extra heuristics, are used. They also use DT-CWT, but they apply the threshold to the magnitudes of complex

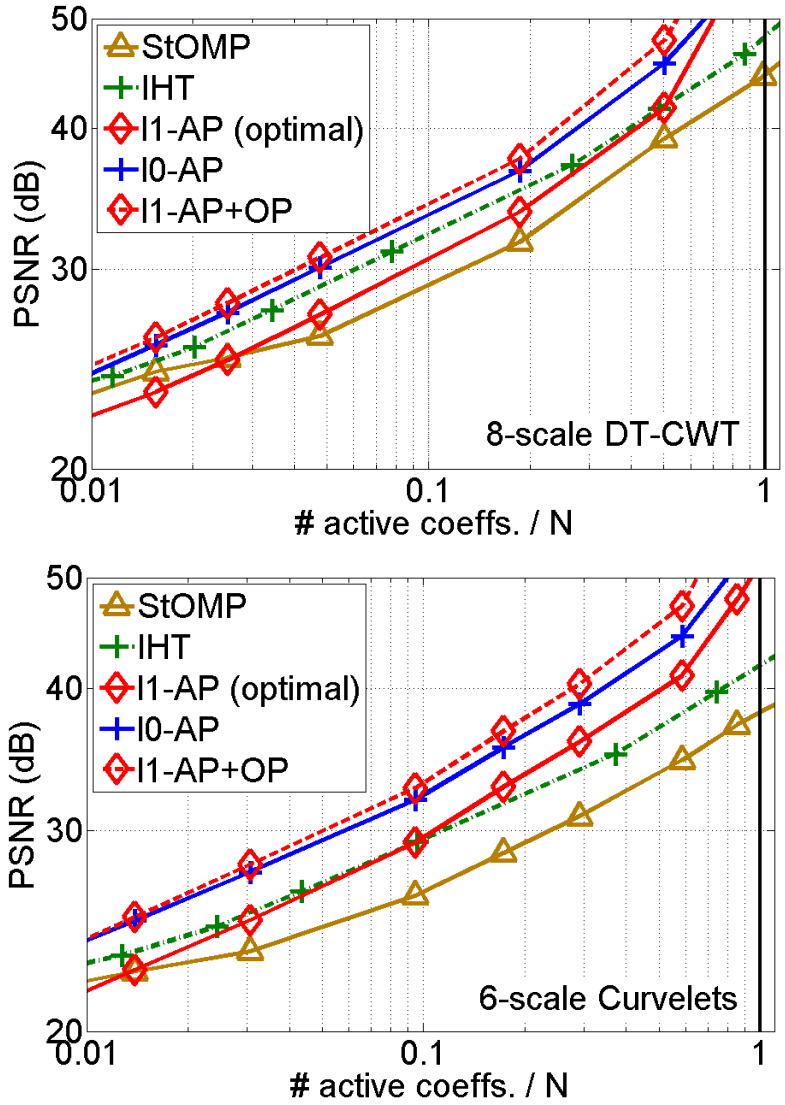

Figura 10.5: *Compaction results, averaged in our test set, of methods ℓ_0-AP, ℓ_1-AP, ℓ_1-AP+OP, IHT and DT+OP.* **Top**, *using 8-scale DT-CWT.* **Bottom**, *using 6-scale Curvelets.*

coefficients. Then, to compare properly their results with ours, we have doubled the given number of selected coefficients in their results. We used 5-scale DT-CWT, as these authors do, and using 24000 fixed coefficients and the 512×512 *Lena* image[8], obtaining a result 2,02 dB above theirs (39,09 vs. 37,07 dB).

In [17] it is also presented a dynamic version increasing the number of active complex coefficients used at each iteration (from 12000 to 24000

[8]We thank Prof. Kingsbury for helping us replicating their experiments.

DT-CWT/Curvelets		R/N		
Image	Method	0.00305	0.00944	0.02914
Barbara	StOMP	25,75/24,02	27,91/27,26	35,99/32,56
	IHT	28,18/25,98	33,25/30,51	40,58/34,30
	ℓ_1-AP	26,39/25,75	30,21/31,02	38,65/38,38
	ℓ_0-AP	*29,23/28,61*	*33,38/34,29*	*41,76/41,51*
	ℓ_1-AP+OP	**29,95/29,10**	**34,12/34,94**	**43,09/43,08**
House	StOMP	28,64/25,52	30,81/29,23	37,88/34,22
	IHT	31,19/28,64	34,76/32,69	40,25/37,77
	ℓ_1-AP	29,60/27,98	32,79/33,32	39,56/39,35
	ℓ_0-AP	*32,09/30,45*	*35,18/35,63*	*43,00/41,87*
	ℓ_1-AP+OP	**32,61/31,19**	**35,79/36,69**	**44,41/43,78**
Boat	StOMP	24,17/22,46	26,35/24,70	32,53/29,23
	IHT	25,46/24,07	30,07/27,62	34,69/31,17
	ℓ_1-AP	24,10/23,41	27,53/27,00	34,56/33,62
	ℓ_0-AP	*26,48/26,15*	*30,43/29,97*	*38,00/36,73*
	ℓ_1-AP+OP	**26,92/26,19**	**31,05/30,24**	**39,08/38,02**
Lena	StOMP	24,95/23,10	27,31/25,88	34,42/30,46
	IHT	27,09/24,86	*32,17*/28,63	39,38/32,54
	ℓ_1-AP	25,50/24,78	28,92/28,72	36,89/35,69
	ℓ_0-AP	*27,72/27,10*	31,63/*31,27*	*40,29/38,66*
	ℓ_1-AP+OP	**28,49/27,64**	**32,50/32,15**	**41,41/40,40**
Peppers	StOMP	24,48/22,97	26,85/25,66	33,14/29,69
	IHT	25,80/24,53	*31,57*/28,41	38,43/32,28
	ℓ_1-AP	24,36/24,47	28,46/28,80	35,99/34,89
	ℓ_0-AP	*27,43/26,85*	31,47/*30,88*	*38,81/37,35*
	ℓ_1-AP+OP	**27,82/27,43**	**32,26/32,13**	**40,12/39,17**

Cuadro 10.1: *Detailed comparison of the methods using* 8*-scale DT-CWT and* 6*-scale Curvelets in our test set. Bold numbers indicate the method providing the best fidelity results for each image and sparseness level. Cursive numbers indicate the second best. Each column corresponds to a number of selected coefficients, whose value is normalised by* N*. The precise sparseness value for each normalised row correspond respectively to* 2001*,* 6189 *and* 19096 *active coefficients. We have extracted directly the PSNR values from the experiments, except for IHT, where the values have been linearly interpolated.*

in 30 iterations in the experiment they describe). In this case they obtain 38,68 dB in the approximation, still 0,41 dB below our result. However, it is easy to check that this difference is caused by the extra flexibility of our implementation which activate independently the real and imaginary parts

DT-CWT/Curvelets		R/N		
Image	Method	0.05851	0.08552	1.4873
Barbara	StOMP	42,88/36,91	45,02/38,56	49,86/43,03
	IHT	44,73/39,69	47,69/42,78	53,44/47,16
	ℓ_1-AP	45,39/44,16	50,24/50,20	$>$ **100**/$>$ **100**
	ℓ_0-AP	*48,31/47,37*	*51,76/53,00*	*61,73/64,86*
	ℓ_1-AP+OP	**52,03/51,33**	**56,88/61,48**	$>$ **100**/$>$ **100**
House	StOMP	42,78/37,52	45,49/41,29	51,65/44,32
	IHT	45,85/42,02	50,01/43,95	55,98/47,94
	ℓ_1-AP	46,22/43,78	50,09/50,94	$>$ **100**/$>$ **100**
	ℓ_0-AP	*50,92/46,52*	*54,56/53,96*	*67,38/63,13*
	ℓ_1-AP+OP	**53,18/49,17**	**57,05/60,94**	$>$ **100**/$>$ **100**
Boat	StOMP	37,73/32,67	40,20/35,00	45,56/38,01
	IHT	40,61/36,07	44,29/38,27	52,50/42,61
	ℓ_1-AP	40,71/38,73	45,86/45,29	$>$ **100**/57,46
	ℓ_0-AP	*45,50/42,54*	*50,22/49,66*	*63,29/58,70*
	ℓ_1-AP+OP	**47,76/45,14**	**52,90/55,64**	$>$ **100**/**71,97**
Lena	StOMP	41,07/34,39	43,31/36,93	48,48/40,95
	IHT	43,65/38,07	46,15/41,61	51,57/45,47
	ℓ_1-AP	43,71/41,30	48,14/49,70	$>$ **100**/$>$ **100**
	ℓ_0-AP	*47,54/44,74*	*51,16/51,02*	*61,60/60,94*
	ℓ_1-AP+OP	**50,67/47,47**	**55,09/58,18**	$>$ **100**/$>$ **100**
Peppers	StOMP	39,10/33,41	41,91/36,31	47,61/39,61
	IHT	42,38/37,22	45,52/40,31	51,60/44,74
	ℓ_1-AP	41,92/39,75	49,84/46,17	$>$ **100**/58,62
	ℓ_0-AP	*45,76/43,04*	*52,13/50,40*	*63,78/61,45*
	ℓ_1-AP+OP	**48,12/45,66**	**56,05/56,30**	$>$ **100**/**72,91**

Cuadro 10.2: *Continuation of Table 10.1. Sparseness values in the normalised columns correspond, respectively, to* 38342, 56048 *and* 97471 *active coefficients.*

of the complex coefficients. Actually, if we use non-separated coefficients in our implementation of ℓ_0-AP, then our result is 1,31 dB *below* their dynamic version (37,37 vs. 38,68 dB). Better results can be achieved by using dynamic threshold and separation of real and imaginary part (see Chapter 11). The method presented in [89] improves heuristically the results of ℓ_0-AP.

10.4.3. Computational load

The time per iteration is dominated in all the methods by one analysis and one synthesis operation. In addition to this, the search for the threshold in ℓ_p-AP also takes a significant amount of time. Other methods like DT+OP and IHT do not require a threshold search, so they are relatively faster. Even so, the time consumed by the methods depends more critically on the number of iterations before reaching the stopping criterion. Table 10.3 shows that ℓ_0-AP requires more iterations than ℓ_1-AP. This difference is partly due to the tolerance used to detect the perfect reconstruction is tighter for ℓ_0-AP than for ℓ_1-AP (see subsection 10.3.2).

It is important to note, as we did in Section 10.1, that most of the real applications do not require so many iterations as shown in these experiments. In this chapter we did not aim to achieve a good compromise between performance and computation time, but we want to explore the quality ceiling of each method to appropriately compare them. However, as we have also experienced (see Chapter 11), methods based on dynamic thresholding (e.g., [97, 15, 17, 21, 40]) are intrinsically faster than those based on a fixed threshold or a fixed number of selected coefficients.

For our experiments, we have used an Intel® $Core^{TM}2$ Duo processor, with 1.66 GHz and with 2 GB RAM. As examples of execution time over 256×256 images, ℓ_0-AP takes around 7 minutes using DT-CWT and around 1 hour using Curvelets. On the other hand, ℓ_1-AP takes around 3 minutes using DT-CWT and 30 minutes using Curvelets. Again, these running times are not representative of a real application, for which much fewer iterations would be applied.

	♯ Iterations	
Methods	DT-CWT	Curvelets
IHT	180	231
DT+OP	188	174
ℓ_1-AP	263	360
ℓ_1-AP+OP	333	440
ℓ_0-AP	495	920

Cuadro 10.3: *Averaged number of iterations in our test set using* 8*-scale DT-CWT and* 6*-scale Curvelets, for the different methods compared.*

10.5. Conclusions

In this chapter, we have presented an optimisation method, which we call ℓ_p-AP, based on minimising the MSE of the reconstruction of an image using a Parseval frame and given a maximum ℓ_p-norm for that vector in that representation. Given p and R, the method consists of alternatively orthogonally projecting between the ℓ_p-ball of radius R, centred at the origin, and onto the set of vectors reconstructing perfectly the image. A global optimum is achieved when $p \geq 1$, and a local one when $0 \leq p < 1$. We have applied this method to the sparse approximation problem. We have focused on $p = 0$ and $p = 1$. The case of ℓ_0-AP translates into a heuristical algorithm previously proposed in [17]. On the other hand, ℓ_1-AP is similar to the method, developed in a parallel and independent work, in [40].

Through systematic experiments, we have shown that ℓ_0-AP clearly outperforms ℓ_1-AP in terms of energy compaction of natural images using widely used pyramidal representations, no matter ℓ_1-AP is optimal for the convex relaxation problem. Moreover, this behaviour is consisten throughout the representations studied. This result shows that the conditions for achieving a global optimum to the sparse approximation problem by using convex relaxation are not held when using natural image and typical representations. Nevertheless, we can improve ℓ_1-AP results by LS-optimising *a posteriori* the amplitudes of the selected coefficients. Applying this, we have shown that the selection of coefficients made by ℓ_1-AP is slightly superior to that of ℓ_0-AP. In the next chapter, however, we will show that this selection is still far from optimal.

We have also compared iterative to shrinkage methods based on fixed thresholds and greedy strategies, showing that ℓ_0-AP also outperforms IHT, IST, our implemented version of StOMP, and DT+OP. We would need more exhaustive test to establish the superiority of ℓ_0-AP over greedy methods in general, but the huge computational effort required by more strict greedy algorithms prevented us from doing this comparison. Among the methods mentioned before (excluding ours), we have seen that IHT achieves the best compaction results.

Although not compared in detail here, but in the next chapter, methods based on dynamically adjusting the threshold through iterations provide, until this date, the best compaction performance. But, up to now, these methods have not been mathematically formulated, as we have done here with ℓ_p-AP. It is easy to adapt our method to iteratively increase the number of selected coefficients (as in [17]). An additional fact is that, for some restoration tasks (as, for example, spatial quantisation artifacts, see Chapter 13) we have experienced that not always sparser solutions are used.

Another additional advantage of our method is that we use less parameters than other similar ones [15, 17, 21]. However, it still requires to establish a radius for the ℓ_p-ball. This disadvantage is overcome in the method proposed in next chapter.

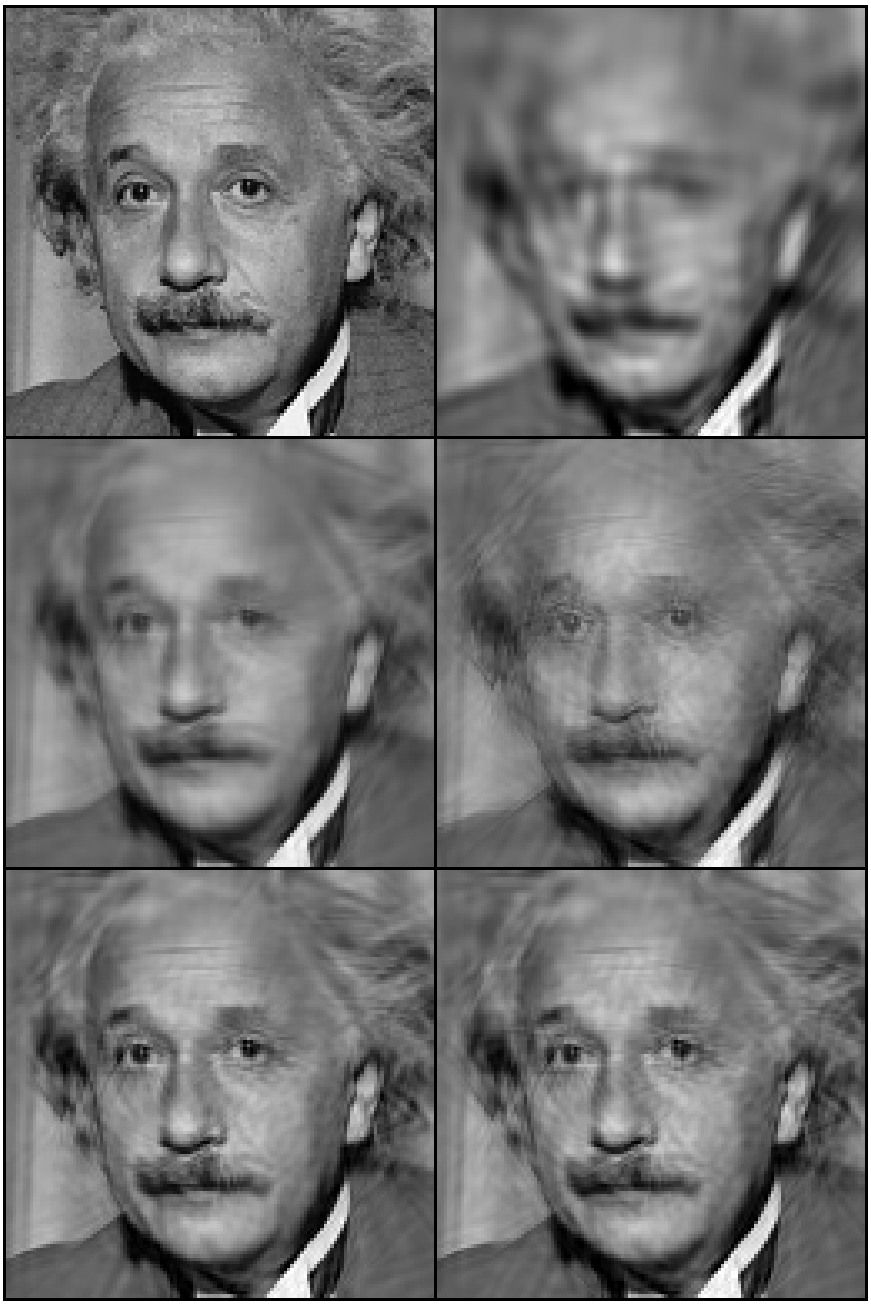

Figura 10.6: *Visual comparison of the methods using* $0{,}0765 \cdot N$ *Curvelets coefficients and the Einstein image, , where* N *is the number of pixels in the image. Results are cropped to* 128×128*, starting at pixel (*71, 41*), to improved the visibility.* **Left column**, *from top to bottom: original image and results of* ℓ_1*-AP (*30,85 *dB) and* ℓ_1*-AP+OP (*33,52 *dB).* **Right column**, *from top to bottom: results from StOMP (*28,66 *dB), IHT (*29,10 *dB) and* ℓ_0*-AP (*32,98 *dB).*

Capítulo 11

Sparse approximation using gradient descent

In this chapter we mathematically derive another method to solve the sparse approximation problem. It is more accurate and efficient than the one described in the previous chapter, but it maintains the advantage of being a solution to an explicit optimisation problem. It is designed from the next question: Is it possible to make gradient descent in the criterion to be minimised in the Equation (9.4)? The answer, due to the discontinuous nature of the ℓ_0-norm, is "not directly". However, we will write an equivalent continuous criterion which allows to calculate the gradient direction. Then we will obtain a generalised version of IHT and proof that the fixed point of its iterations is a local minimum of the cost function at hand.

Moreover, to avoid getting trapped by unfavourable local minima, we will apply a deterministic annealing technique similar to other non-convex global optimisation algorithms [122, 123, 21]. We name the resulting method ℓ_0-GM. We show through experiments that ℓ_0-GM is competitive with current state-of-art in terms of energy compaction, outperforming both ℓ_0-AP and our LS-optimized version of ℓ_1-AP (ℓ_1-AP+OP).

We derive analogously the IST method throughout the gradient descent in a function equivalent to the criterion to be minimised in the convex relaxation problem (Equation (9.5)). We also derive a convex variant of ℓ_0-GM, which we name ℓ_1-GM. We will show that it achieves comparable results to other convex relaxation methods, and we will describe the practical cases where it should be used.

We have that ℓ_0-GM is a dynamic thresholding method. The idea of decreasing the threshold as iterations are executed is not new [15, 17, 21, 93, 94, 19]. Nevertheless, up to our knowledge, this is the first time that it has been formally derived as a direct solution to the sparse approximation

problem. In addition, nobody has analysed, in a certain depth, the reasons why this solution behaves so well.

We start by reformulating the sparse approximation cost function (discontinuous and unconstrained) in a continuous and constrained form (Section 11.1). Then, in Section 11.2 we derive the generalised IHT as local solution to the sparse approximation problem. We justify the use of a decreasing threshold in Section 11.3. In Section 11.4 implementation details of ℓ_0-GM are given, and in Section 11.5 we compare the energy compaction capacity of ℓ_0-GM to the methods studied in previous chapter. Finally, we derive the IST and ℓ_1-GM methods in Section 11.6, and compare them to ℓ_1-AP. Section 11.7 concludes this chapter.

11.1. An alternative formulation with a continuous cost function

We repeat here, for convenience, the sparse approximation problem formulation of Equation (9.4):

$$\hat{\mathbf{a}}^0(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}. \tag{11.1}$$

The associated cost function is not only non-convex, but it is also discontinuous. This prevents a direct calculation of its gradient. Next we derive a new equivalent continuous and constrained function, whose gradient can be calculated. We start from the following formulation:

$$(\hat{\mathbf{a}}, \hat{\mathbf{b}}) = \arg\min_{\mathbf{a},\mathbf{b}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2 \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\}, \tag{11.2}$$

and prove the equality $\hat{\mathbf{a}} = \hat{\mathbf{a}}^0(\lambda)$. Firstly we express Equation (11.2) as:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda \min_{\mathbf{b}}\{\|\mathbf{b} - \mathbf{a}\|_2^2 \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\}\}. \tag{11.3}$$

Note that the inner minimisation is the orthogonal projection of $\mathbf{a}$ onto the affine subspace $S(\mathbf{\Phi}, \mathbf{x})$ of perfect reconstruction of $\mathbf{x}$. This projection was already defined in Equation (10.4). We repeat its expression here for convenience:

$$P^{\perp}_{S(\mathbf{\Phi},\mathbf{x})}(\mathbf{a}) = \mathbf{a} + \mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi a}).$$

Substituting it in Equation (11.2) we obtain:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{\Phi}^T(\mathbf{\Phi a} - \mathbf{x})\|_2^2\}.$$

Given that $\mathbf{\Phi}$ is a Parseval frame, it finally yields:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\} = \hat{\mathbf{a}}^0(\lambda),$$

as we wanted to prove. In the next step, in order to obtain a continuous and constrained cost function only depending on $\mathbf{b}$, we start from Equation (11.2) and swap the minimization variables with respect to Equation (11.3):

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\min_{\mathbf{a}}\{\|\mathbf{a}\|_0 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2\} \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\}.$$

It is easy to see that, in this case, minimizing this cost function for vector $\mathbf{a}$ is equivalent to minimizing independently for each index. We express the cost as $c(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{M} c'(a_i, b_i)$, where:

$$c'(a, b) = \begin{cases} 1 + \lambda(b - a)^2, & |a| > 0 \\ \lambda b^2, & |a| = 0. \end{cases}$$

Given $\mathbf{b}$, it is easy to see that if the value $\tilde{a}_i(b_i)$ minimizing $c'(a_i, b_i)$ is not zero, then $\tilde{a}_i(b_i) = b_i$, and $c'(\tilde{a}_i(b_i), b_i) = 1$. Then, we have:

$$c(\tilde{\mathbf{a}}(\mathbf{b}), \mathbf{b}) = \sum_{i=1}^{M} \min(1, \lambda b_i^2).$$

Figure 11.1 shows a one-dimensional illustration of this minimum (with $\lambda = 1$). Given some λ value, we note θ the value holding $\lambda\theta^2 = 1$. Therefore:

$$\theta = \lambda^{-\frac{1}{2}},$$

and we have that:

$$\tilde{a}_i(b_i) = \begin{cases} b_i, & |b_i| > \theta \\ 0, & |b_i| \leq \theta. \end{cases}$$

This is a hard-thresholding operation with threshold θ, which we note $\tilde{\mathbf{a}}(\mathbf{b}) = S_0(\mathbf{b}, \theta)$. Substituting $S_0(\mathbf{b}, \theta)$ for $\mathbf{a}$, in Equation (11.2):

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\|S_0(\mathbf{b}, \theta)\|_0 + \lambda\|\mathbf{b} - S_0(\mathbf{b}, \theta)\|_2^2 \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\}.$$

When evaluating this criterion for each coefficient in $\mathbf{b}$, one of the two terms (fidelity or sparseness) is zero. Thus, we can express the same as:

$$\begin{aligned} \hat{\mathbf{b}} &= \arg\min_{\mathbf{b}}\{C_0(\mathbf{b}, \theta) \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\}, \qquad (11.4) \\ \hat{\mathbf{a}} &= S_0(\hat{\mathbf{b}}, \theta), \end{aligned}$$

where:

$$C_0(\mathbf{b}, \theta) = \sum_{i=1}^{M} \min\left(1, \left(\frac{b_i}{\theta}\right)^2\right). \qquad (11.5)$$

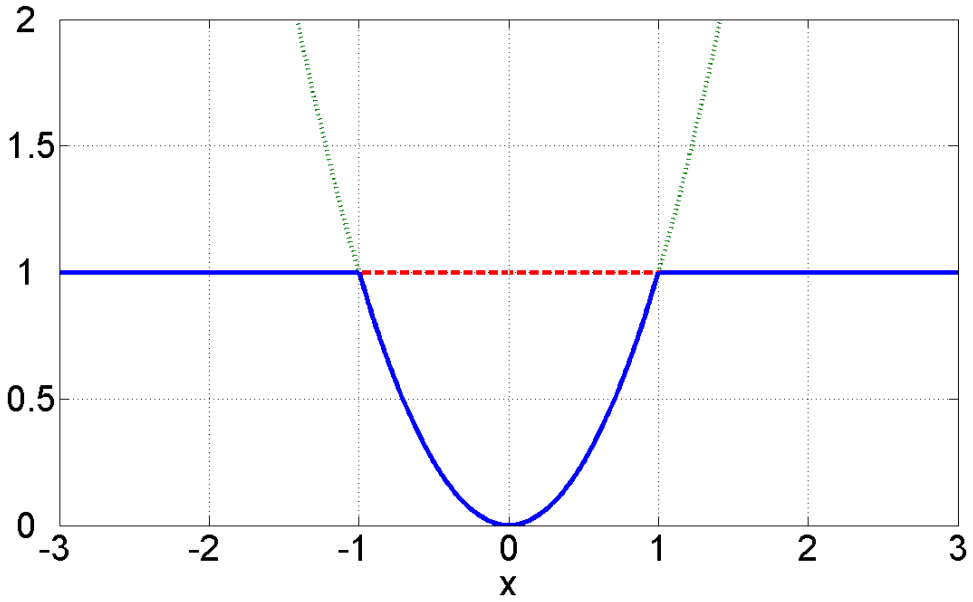

Figura 11.1: *Bold line shows the minimum between* $y(x) = 1$ *(dashed) and* $y(x) = x^2$ *(dotted).*

11.2. Local minimisation with ℓ_0-norm: IHT

The gradient of the new (unconstrained) cost function is $\nabla C_0(\mathbf{b}, \theta) = \mathbf{c}$, where:

$$c_i = \begin{cases} 0, & |b_i| > \theta \\ \frac{2}{\theta^2} b_i, & |b_i| \leq \theta. \end{cases}$$

This can be expressed more compactly as:

$$\nabla C_0(\mathbf{b}, \theta) = \frac{2}{\theta^2} \left(\mathbf{b} - S_0(\mathbf{b}, \theta) \right),$$

The projection of this gradient onto the affine subspace of perfect reconstruction, $S(\mathbf{\Phi}, \mathbf{x})$, is:

$$\nabla^{S(\mathbf{\Phi},\mathbf{x})} C_0(\mathbf{b}, \theta) = (\mathbf{I} - \mathbf{\Phi}^T \mathbf{\Phi}) \nabla C_0(\mathbf{b}, \theta).$$

Every iteration of the gradient descent method is:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \alpha \nabla^{S(\mathbf{\Phi},\mathbf{x})} C_0(\mathbf{b}^{(k)}, \theta).$$

As this projection is the component of the gradient in the null space of $\mathbf{\Phi}$, $\mathbf{b}^{(k)}$ always provides perfect reconstruction, no matter which value of α we use. Substituting the gradient expression we obtain:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \frac{2\alpha}{\theta^2} (\mathbf{I} - \mathbf{\Phi}^T \mathbf{\Phi}) \left(\mathbf{b}^{(k)} - S_0(\mathbf{b}^{(k)}, \theta) \right).$$

A necessary and, in our case, sufficient condition to reach a local minimum of the cost function is that:

$$\nabla^{S(\mathbf{\Phi},\mathbf{x})} C(\mathbf{b}^*, \theta) = \mathbf{0}.$$

This is the convergence condition of the previous iterations. This means that, if those iterations converge ,they do it to a local minimum of the cost function in Equation (11.5).

Note that the choice of $\alpha = \alpha_0 = \frac{1}{2\lambda} = \frac{\theta^2}{2}$ minimises the unconstrained cost function of Equation (11.5) for a single descent step, resulting in:

$$\mathbf{b}^{(k+1)} = S_0(\mathbf{b}^{(k)}, \theta) + \mathbf{\Phi}^T \left(\mathbf{x} - \mathbf{\Phi} S_0(\mathbf{b}^{(k)}, \theta)\right),$$

that is, the same Iterative Hard Thresholding (IHT) method described in subsection 9.3.3.

We have shown that this procedure provides, when converging, a local minimum in the classical sparse approximation criterion (Equation (11.1)). However, in general, choosing the α value which minimises in one step the unconstrained cost function (α_0) is not optimal in terms of convergence speed. We have hand-optimised the convergence speed by using $\alpha \sim 1{,}85\alpha_0$.

Recently, we have known that [29] also proved, in a parallel and independent way to our work, that the convergence point of the IHT iterations is a local minimum of the cost function[1]. However, they prove, in addition, that the iterations converge indeed, provided that the eigenvalues of $(\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi})$ are between 0 and 1, where $\mathbf{I}$ is the $M \times M$ identity matrix.

Figure 11.2 shows some convergence curves using fixed thresholds and different α values, and using 8-scale DT-CWT as representation[2], whose redundancy factor is 4. We can see that, although the striking simplicity of this method, doing gradient descent for a given λ value until convergence is too expensive in computational terms. In addition, we know that the local minimum obtained is clearly worse than that of ℓ_0-AP (see Section 10.4).

11.3. Global minimisation with ℓ_0-norm: ℓ_0-GM

We propose next an efficient alternative to IHT and ℓ_0-AP, inspired by deterministic global optimisation techniques, which drastically reduces

[1]This work was published in April 2007, while ours [12] appeared in August same year.

[2]Except when indicated, this is the representation used throughout the chapter. We have experienced that other representations, as Curvelets, provide a qualitatively similar behaviour.

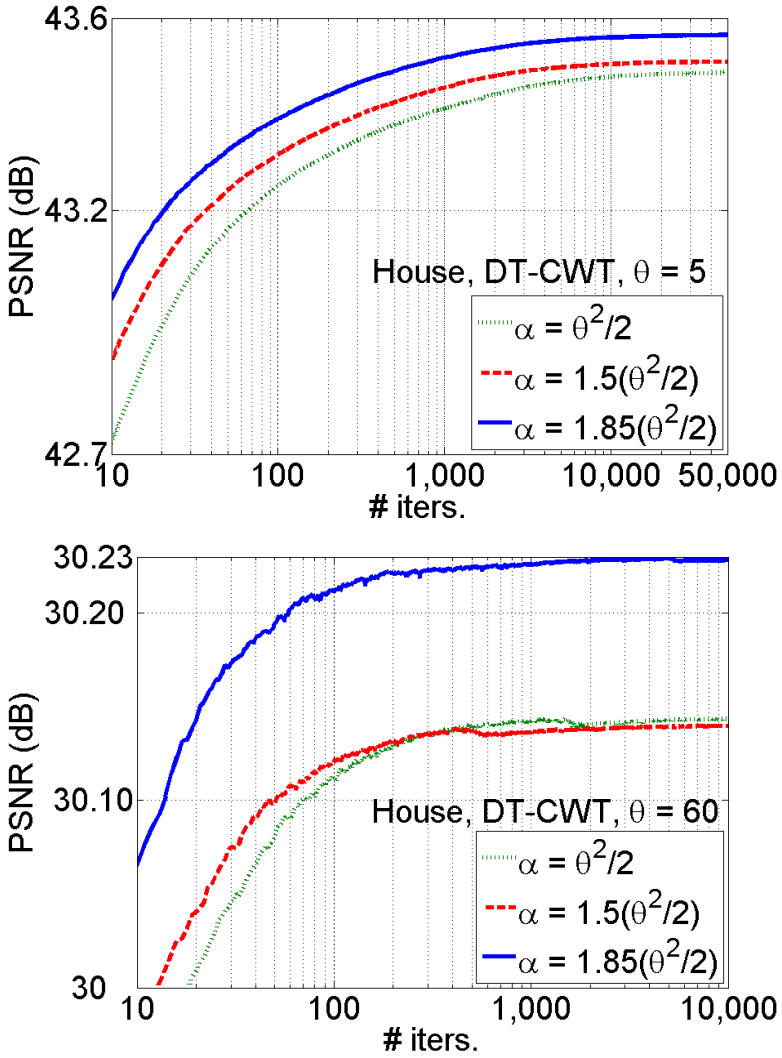

Figura 11.2: **Top**, *IHT convergence curves using a low threshold ($\theta = 5$) and three different α values. We have used* House *image and 8-scale DT-CWT.* **Bottom**, *same result for a higher threshold ($\theta = 60$).*

computational cost compared to IHT, while increasing the energy compaction capacity.

The cost function in Equation (11.5) can be re-written as:

$$C_0(\mathbf{b}, \theta) = \sum_{i=1}^{M} (1 - h(b_i/\theta)), \tag{11.6}$$

where $h(x) = \text{máx}(1 - x^2, 0)$ is the inverted parabolic arc centred at zero, going from -1 to 1, and reaching a maximum value of one at zero, with an

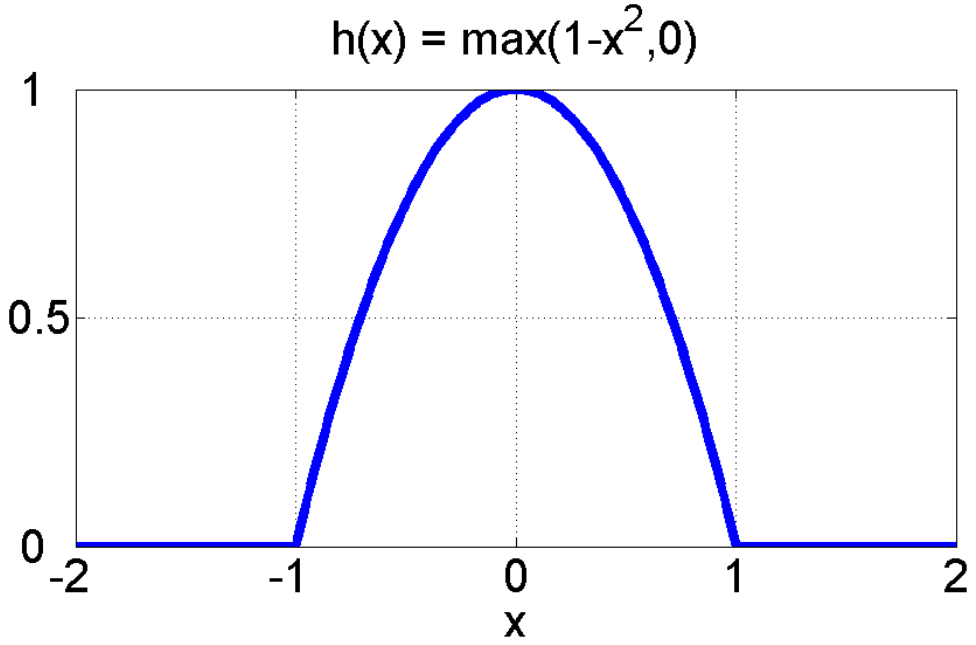

Figura 11.3: 1-*D smoothing function: an inverted parabola in the interval* $[-1, 1]$*, centred at* 0 *and with maximum* 1*. Outside that interval is* 0*.*

amplitude of one. A plot of this function is shown in Figure 11.3. We can re-write the optimisation problem in Equation (11.4) as follows:

$$\begin{aligned} \hat{\mathbf{b}} &= \arg\max_{\mathbf{b}} C'(\mathbf{b}, \theta), \\ \hat{\mathbf{a}} &= S_0(\hat{\mathbf{b}}, \theta), \\ C'(\mathbf{b}, \theta) &= \sum_{i=1}^{M} h(b_i/\theta) = M - C_0(\mathbf{b}, \theta). \end{aligned}$$

And it is easy to express this in terms of a infinitely sharp cost function, $C_\delta(\mathbf{b})$, convolved with a smoothing kernel:

$$C'(\mathbf{b}, \theta) \propto C_\delta(\mathbf{b}) * H(\mathbf{b}/\theta),$$

where $*$ represents the convolution operator; $H(\mathbf{b}) = \prod_{i=1}^{M} h(b_i)$, $C_\delta(\mathbf{b}) = \sum_{i=1}^{M} \delta(b_i)$, and the proportionality factor is $A(\theta)^{-M+1}$, con $A(\theta) = \int_{-\theta}^{\theta} h(x/\theta) = 4\theta/3$. The scale factor, $A(\theta)^{-M+1}$, results from integrate $H(\mathbf{b}/\theta)$ along all dimensions except for the one of the corresponding delta (in which dimension a one-dimensional convolution is performed), and it is irrelevant in terms of the minimisation in vector $\mathbf{b}$.

As Figure 11.2 illustrates, it is faster to find a local optimum when θ is high, or, equivalently, λ is low, which corresponds to a smooth cost

function. Moreover, having a good candidate for the global optimum for a given λ, we can expect a good result by searching from it the nearest optimum corresponding to a similar, slightly higher, λ. From here we derive the following method. Starting from a small λ, we do gradient descent until reaching convergence, then set a slightly higher λ, do again gradient descent from the previous convergence point, and so on until reaching the desired λ value. We call this method ℓ_0-GM (from Gradual Minimisation). A faster and simpler approximated version is to increase slowly λ at each iteration, so drastically reduce the number of iterations. In fact, both versions become equivalent in the limit when the increase of λ at each iteration becomes infinitesimal. In terms of the threshold θ, we start from the highest possible threshold (highest amplitude in $\mathbf{a}^{LS}$) and slowly decrease it at each *de-smoothing* iteration, until reaching the desired value. In Figure 11.4 we illustrate the concept guiding this method with an example of a function of multiple minima smoothed until getting a function with one single minimum. The path joining all the minima throughout the different scales is drawn. In this example there is continuity of the global minima as a function of the scale, which is a necessary condition for ℓ_0-GM to reach the global optimum. This, in general, does not happen in real cases. We have seen that this method finds the global optimum for extremely high sparseness levels (around few tens of DT-CWT coefficients). Nevertheless, global optimality conditions are beyond the scope of this Thesis, where we are more interested in the methods behaviour under practical conditions.

The idea of smoothing a cost function to avoid getting trapped by unfavourable local minima is closely related to other deterministic annealing schemes, such as [122, 123]. Some authors had already proposed this idea as an heuristic to obtain algorithms promoting the energy compaction, by using either soft-thresholding [21] or hard-thresholding [15, 93]. But the referred authors did not propose their algorithms as means to solve well-founded optimisation problems.

Figure 11.5 shows, on the one hand, some convergence trajectories (dashed lines) of IHT for different fixed thresholds (which corresponds to search for a local optimum by doing gradient descent with a fixed λ). On the other hand, it shows two trajectories (circles and solid lines) corresponding to exponentially decrease the threshold with the rule $\theta^{(k)} = \theta^{(0)}\beta^k$ for two different β values. The closer to 1 is β, the better is the compaction, but also the slower is the convergence. We, as other authors [17, 21], have experienced that, in practice, exponential decreasing of the threshold provides a better compromise between computational cost and quality of the result than other decreasing functions, as linear. By decreasing dynamically the threshold we are not only dramatically reducing the required number of iterations for

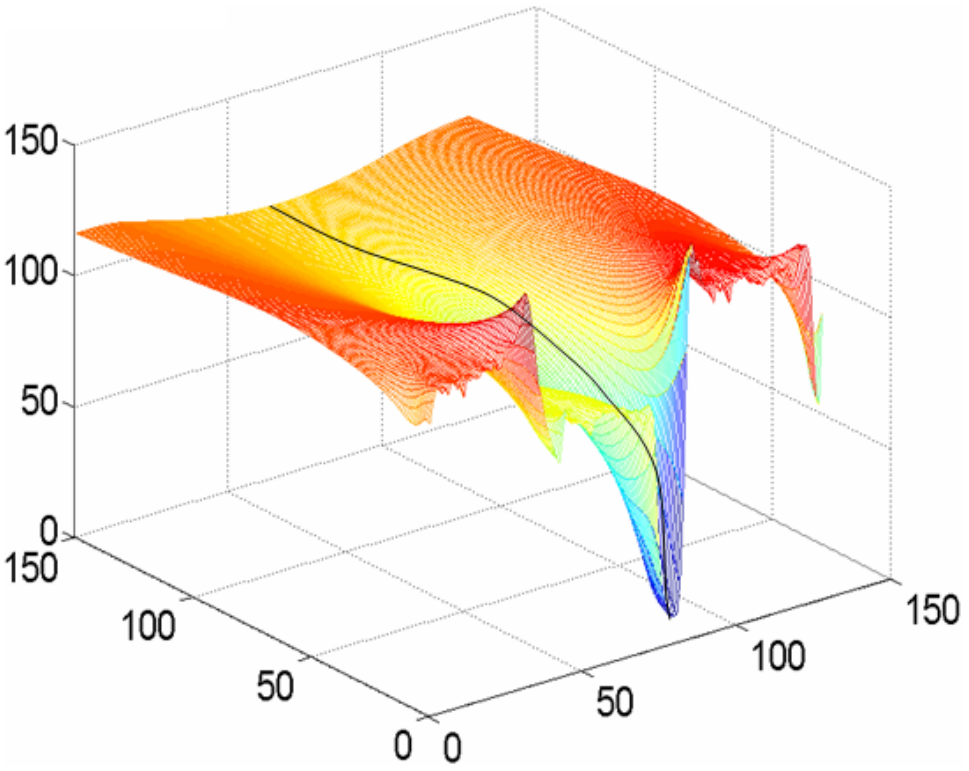

Figura 11.4: 1-*D Function with multiple minima progressively smoothed until obtaining only one. The black continuous line indicates the path joining the global optima through the scale of the smoothing kernel. We have used here as smoothing kernel a normalised (in area) version of* $h(x)$ *(See Figure 11.3).*

reaching convergence, but we are also significantly improving the achievable fidelity for any given sparseness level.

Top panel of Figure 11.6 shows a family of fidelity-sparseness curves for different β values. An ideal curve would have an asymptote to perfect reconstruction in N. Note that ℓ_0-GM approximates this asymptote as β gets closer to 1. This is even more significant if we consider that achieving a global optimum for low sparseness levels is much more difficult than for high sparseness, because the number of local optima increases very rapidly with λ.

11.3.1. Using a single solution for all the sparseness levels

If we optimise using ℓ_0-GM for a set of λ values, we end up having multiple solutions, one for every value taken by the threshold in its

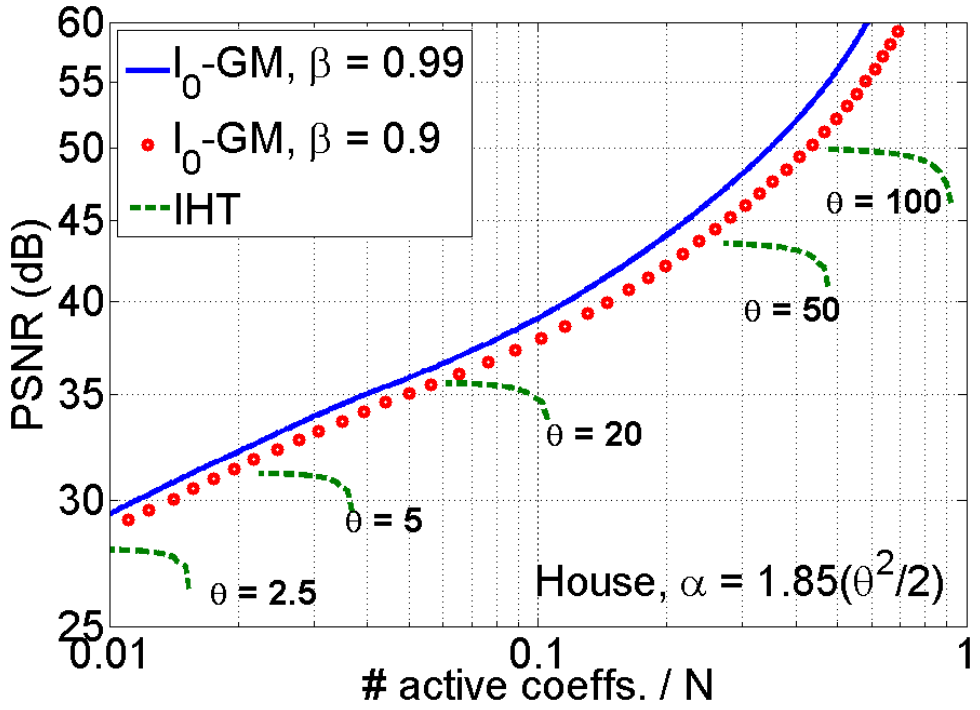

Figura 11.5: *Fidelity-sparseness results of ℓ_0-GM, using $\beta = 0{,}9$ (circles, $1{,}5 \cdot 10^2$ iterations) and $\beta = 0{,}99$ (solid, $1{,}5 \cdot 10^3$ iterations), compared to IHT, using several thresholds (dashed, 10^5 iterations). We use* House *image and DT-CWT with 8-scales.*

descending path. Which criterion should we apply to choose a particular solution? Is it possible to find a θ_0 value whose associated minimum of the cost function, $C_0(\hat{\mathbf{b}}(\theta_0), \theta_0)$, can be extended as $C_0(\hat{\mathbf{b}}(\theta_0), \theta)$ to approximate the minimum of the cost function for other thresholds θ (that is, such that $C_0(\hat{\mathbf{b}}(\theta_0), \theta) \approx C_0(\hat{\mathbf{b}}(\theta), \theta)$, for all $\theta > \theta_0$)? The answer, surprisingly enough, is ¿es". This problem has an important practical impact, because in that case we could use $\hat{\mathbf{a}}'(\theta) = S_0(\hat{\mathbf{b}}(\theta_0), \theta)$ as an almost equally good substitute of $\hat{\mathbf{a}} = S_0(\hat{\mathbf{b}}(\theta), \theta)$, that does not require to use, and store[3], $\hat{\mathbf{b}}(\theta)$.

We have explained before how, to get a good solution for some $\lambda_i > \lambda_j$, it is good to start from the solution associated to λ_j and, from there, refine it until reaching the solution associated to λ_i. This seems to imply that good solutions for high λ values should be reasonable good for lower values. In bottom panel of Figure 11.6 we show the fidelity-sparseness curve obtained by *a posteriori* thresholding the solution obtained using a very high λ value. As we can see, for $\beta < 0{,}99$, the results are even better than the curve

[3]However, this does not prevent us from computing $\hat{\mathbf{b}}(\theta)$, if $\theta > \theta_0$, because the computation of $\hat{\mathbf{b}}(\theta_0)$ in ℓ_0-GM requires the computation of $\hat{\mathbf{b}}(\theta)$, for all $\theta > \theta_0$ (in practice, a dense enough sampling of the interval $[\theta_{max}, \theta_0]$)

obtained throughout the execution of the direct ℓ_0-GM method. This means that only an optimisation solution, for a determined sparseness level (the lowest one) is enough to have a good approximate solution to all λ values considered. This is a practical advantage, because it means that we do not need to store all $\hat{\mathbf{b}}(\theta)$ to choose a threshold level θ, corresponding to a certain λ, in real time. This allows, for example, for adapting to a variable channel bandwidth in communications, and it provides, in general, a flexible approach to quickly shift the trade-off between fidelity and sparseness.

11.4. Implementation

We have experimented with several Parseval frames, as we did with ℓ_p-AP method. Although the qualitative conclusions of the experiments are similar using any of them, we have chosen DT-CWT to show the experiments in this chapter. Together with Curvelets, it offers the best compaction results among the compared representations. In addition to this, the MATLAB® implementation available [120] is much faster than the one for Curvelets [121].

Similarly to the previous chapter, the complex coefficients of DT-CWT have been separated in real and imaginary parts, in order to make a homogeneous treatment of them. Moreover, it has also been added an extra scaled composed by one only coefficient storing the global mean of the image.

We have checked that, in our ℓ_0-GM implementation, the best results are obtained when the decrease interval of θ is as large as possible. Then, the threshold is initialised to the second largest amplitude of the linear response to the image (to choose at least one coefficient in the first iteration). Then it is decreased until reaching the desired value. This value depends on the application. Following the previous section, we choose a small final value for θ, then obtaining a solution with good performance at every sparseness level.

11.5. Results and discussion for ℓ_0-GM

Figure 11.7 compares the compaction performance of the following methods: ℓ_0-GM with $\beta = 0{,}99$, ℓ_0-AP, post LS-optimised ℓ_1-AP (ℓ_1-AP+OP), IHT and StOMP. See more details about these methods in Chapter 10. The improvement on the behaviour of ℓ_0-GM with respect to the rest is very remarkable. These plot show clearly that we can obtain, in the conditions of this experiment, a much better local minimum to the sparse approximation problem by directly minimising the ℓ_0-norm than solving

the convex relaxation problem, even LS-optimising the coefficients for the selected support. One important difference of ℓ_0-GM with the methods based on alternating projections is that we can sweep all the sparseness levels in the same execution of the method, instead of making a lot of iterations for each level, each time. Tables 11.1 and 11.2 show the numerical data of Figure 11.7.

Other strategies exist in the literature for the dynamic thresholding, and depending on the precise case, they can give slightly better results than those of ℓ_0-GM. For example, in [17], the number of preserved coefficients at each iteration is linearly increased.

Figure 11.8 shows a visual comparison of sparse approximation with the different methods using $0{,}04 \cdot N$ DT-CWT coefficients and *Einstein* image. We see that ℓ_0-GM preserves significantly better the perceptually relevant information of the original.

11.6. Gradient descent for minimisation of ℓ_1-norm: IST & ℓ_1-GM

11.6.1. Alternative formulation of the convex cost function

We described the convex relaxation problem in Equation (9.5), which we repeat here for convenience:

$$\hat{\mathbf{a}}^1(\lambda) = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{\Phi a} - \mathbf{x}\|_2^2\}, \tag{11.7}$$

The associated cost function, in contrast to the ℓ_0-norm case, is convex and, thus, continuous. Nevertheless, we are interested in doing a similar transformation as for that case. The proof of that solution $\hat{\mathbf{a}}$ of the problem:

$$(\hat{\mathbf{a}}, \hat{\mathbf{b}}) = \arg\min_{\mathbf{a},\mathbf{b}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2 \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\} \tag{11.8}$$

is equivalent to $\hat{\mathbf{a}}^1(\lambda)$ is analogous to the one ℓ_0-norm case. We can express $\hat{\mathbf{b}}$ as:

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\min_{\mathbf{a}}\{\|\mathbf{a}\|_1 + \lambda\|\mathbf{b} - \mathbf{a}\|_2^2\} \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\}. \tag{11.9}$$

First, we find the generic expression minimising the inner cost function given $\mathbf{b}$. This cost can be decomposed as a summation of a coefficient for each

		♯ active coeffs./N		
Image	Method	0,00868	0,02536	0,04761
Barbara	StOMP	24,48	25,55	26,45
	IHT	23,16	27,25	29,89
	ℓ_0-AP	24,89	28,67	31,89
	ℓ_1-AP+OP	*25,24*	*29,38*	*32,60*
	ℓ_0-GM	**26,18**	**30,47**	**33,93**
House	StOMP	26,40	28,27	29,82
	IHT	24,08	30,35	32,91
	ℓ_0-AP	27,33	31,50	34,16
	ℓ_1-AP+OP	*27,75*	*32,01*	*34,73*
	ℓ_0-GM	**28,85**	**33,18**	**35,65**
Boat	StOMP	20,75	23,89	24,89
	IHT	16,82	24,74	27,09
	ℓ_0-AP	21,57	25,76	27,86
	ℓ_1-AP+OP	*21,96*	*26,12*	*28,37*
	ℓ_0-GM	**23,77**	**27,32**	**30,13**
Lena	StOMP	23,21	24,67	26,02
	IHT	17,27	26,22	28,93
	ℓ_0-AP	23,62	27,20	30,09
	ℓ_1-AP+OP	*24,17*	*27,96*	*30,95*
	ℓ_0-GM	**25,19**	**29,29**	**32,34**
Peppers	StOMP	21,43	24,19	25,28
	IHT	16,76	25,15	27,93
	ℓ_0-AP	22,61	26,39	29,17
	ℓ_1-AP+OP	*22,81*	*26,85*	*29,74*
	ℓ_0-GM	**24,03**	**28,43**	**31,57**

Cuadro 11.1: *Fidelity (PSNR, in dB) for several sparseness levels, using the images in our test set and five different methods, and using* 8*-scale DT-CWT. Bold numbers indicate the method providing the best approximation for each image and sparseness level, and italic indicate the second best. Columns correspond to* 569*,* 1662 *and* 3120 *active coefficients. We have directly extracted the PSNR values from the experiments, except for IHT, where they have been linearly interpolated.*

element of the involved vectors, so the vector $\tilde{\mathbf{a}}^s(\mathbf{b})$ minimising the inner criterion in Equation (11.9) is:

$$\tilde{\mathbf{a}}^s(\mathbf{b}) = \min_{\mathbf{a}}\{\sum_{i=1}^{M} c(a_i, b_i)\},$$

		♯ active coeffs./N		
Image	Method	0,1865	0,5021	0,9869
Barbara	StOMP	34,58	42,39	46,63
	IHT	35,68	44,12	48,56
	ℓ_0-AP	40,39	47,63	55,64
	ℓ_1-AP+OP	*41,57*	*51,22*	$>$ **100**
	ℓ_0-GM	**43,05**	**54,93**	$>$ **100**
House	StOMP	33,86	41,83	47,89
	IHT	38,14	44,61	50,75
	ℓ_0-AP	38,40	49,25	58,98
	ℓ_1-AP+OP	*39,20*	*51,25*	*62,31*
	ℓ_0-GM	**43,47**	**56,12**	$>$ **100**
Boat	StOMP	29,06	36,42	42,05
	IHT	32,07	39,30	45,65
	ℓ_0-AP	33,52	43,29	58,91
	ℓ_1-AP+OP	*34,25*	*45,02*	$>$ **100**
	ℓ_0-GM	**38,09**	**50,10**	$>$ **100**
Lena	StOMP	31,70	39,48	45,06
	IHT	34,33	42,30	48,21
	ℓ_0-AP	36,85	45,66	55,51
	ℓ_1-AP+OP	*37,79*	*47,95*	*63,72*
	ℓ_0-GM	**40,97**	**52,97**	$>$ **100**
Peppers	StOMP	31,60	38,50	43,17
	IHT	33,98	41,19	48,66
	ℓ_0-AP	37,20	44,84	57,09
	ℓ_1-AP+OP	*38,41*	*47,13*	$>$ **100**
	ℓ_0-GM	**39,93**	**51,63**	$>$ **100**

Cuadro 11.2: *Continuation of Table 11.1. Columns correspond, respectively, to* 12221, 32905 *and* 64682 *active coefficients.*

where $c(a,b) = |a| + \lambda(b-a)^2$. The derivative in a of this function is $\frac{\partial c(a,b)}{\partial a} = d + 2\lambda(a-b)$, where:

$$d = \begin{cases} 1, & a > 0 \\ -1, & a < 0 \\ 0, & a = 0. \end{cases} \tag{11.10}$$

For the case $a > 0$, we have that:

$$\frac{\partial c(a,b)}{\partial a} = 1 + 2\lambda(a-b).$$

Equaling to zero we obtain:

$$a = b - \frac{1}{2\lambda},$$

from where it follows that $b > \frac{1}{2\lambda}$, given that $\lambda > 0$ by definition. For the case $a < 0$, analogously, we obtain:

$$a = b + \frac{1}{2\lambda},$$

and then $b < -\frac{1}{2\lambda}$. Joining these two cases we have that:

$$a = sign(b) \cdot (|b| - \frac{1}{2\lambda}),$$

when $|b| > \frac{1}{2\lambda}$.

On the other hand, when $|b| \leq \frac{1}{2\lambda}$, the value of a minimising the associated cost function changes the sign with respect to b. Given that for every quadrant we only consider values in the same quadrant, this implies that the minimum is at zero.

Thus, by applying these results in our problem, we obtain that the vector $\tilde{\mathbf{a}}^s(\mathbf{b})$ is the result of a soft-thresholding operation of $\mathbf{b}$ with threshold $\theta = \frac{1}{2\lambda}$. We denote this operation as $\tilde{\mathbf{a}}^s(\mathbf{b}) = S_1(\mathbf{b}, \theta)$. Substituting in Equation (11.8):

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b}}\{\|S_1(\mathbf{b}, \theta)\|_1 + \lambda\|\mathbf{b} - S_1(\mathbf{b}, \theta)\|_2^2 \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\},$$

And finally, given that the cost function of the previous expression is separable as a sum of independent terms for each coefficient index, we can write:

$$\begin{aligned} \hat{\mathbf{b}} &= \arg\min_{\mathbf{b}}\{C_1(\mathbf{b}, \theta) \text{ s.t. } \mathbf{\Phi b} = \mathbf{x}\}, \\ \hat{\mathbf{a}} &= S_1(\hat{\mathbf{b}}, \theta), \end{aligned}$$

where:

$$C_1(\mathbf{b}, \theta) = \sum_{i=1}^{M} \min\left(|b_i| - \frac{\theta}{2}, \frac{b_i^2}{2\theta}\right). \tag{11.11}$$

11.6.2. Cost function minimisation with a fixed threshold: IST

The derivation of the gradient descent-based method with the cost function $C_1(\mathbf{b}, \theta)$ is analogous to the one shown for function $C_0(\mathbf{b}, \theta)$. From Equation 11.11 we obtain:

$$\nabla C_1(\mathbf{b}, \theta) = \frac{1}{\theta}(\mathbf{b} - S_1(\mathbf{b}, \theta)),$$

and, after projecting onto the affine space of perfect reconstruction, $\nabla^{S(\mathbf{\Phi},\mathbf{x})} C_1(\mathbf{b}, \theta) = (\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi})\nabla C_1(\mathbf{b}, \theta)$, we end up with the following iterations:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \frac{\alpha}{\theta}(\mathbf{I} - \mathbf{\Phi}^T\mathbf{\Phi}) \left(\mathbf{b}^{(k)} - S_1(\mathbf{b}^{(k)}, \theta)\right).$$

A necessary (and also sufficient, in this case) condition to reach the global minimum of the function $C_1(\mathbf{b}, \theta)$ is that $\nabla^{S(\mathbf{\Phi},\mathbf{x})} C_1(\mathbf{b}^*, \theta) = \mathbf{0}$. Note that this is the convergence condition for the previous iterations. Choosing $\alpha = \alpha_0 = \frac{1}{2\lambda} = \theta$ leads us to IST method:

$$\mathbf{b}^{(k+1)} = S_1(\mathbf{b}^{(k)}, \theta) + \mathbf{\Phi}^T \left(\mathbf{x} - \mathbf{\Phi} S_1(\mathbf{b}^{(k)}, \theta)\right).$$

It has been proved [30, 39] that this procedure provides the global minimum of the convex relaxation problem shown in Equation (11.7).

However, as in the case $p = 0$, in general the $\alpha = \theta$ choice, though it minimises in α in one step the unconstrained cost function, is not optimal in terms of convergence speed. We have also experienced that we can obtain a faster convergence using $\alpha \sim 1{,}85\alpha_0$, though now the difference of using $\alpha = \alpha_0$ is very small. Figure 11.9 shows the corresponding convergence curves in the same conditions stated for IHT in Figure 11.2. The convergence here is much much faster and uniform than for IHT, because of the convexity of the cost function.

11.6.3. A more efficient convex minimisation: ℓ_1-GM

We can derive an equivalent, but often more efficient, alternative method to IST and ℓ_1-AP. For any θ value, the cost function $C_1(\mathbf{b}, \theta)$ is convex and therefore one can find its global minimum using IST or ℓ_1-AP. As the value of θ is increased (λ decreased), the quadratic term of the function dominates, which provokes a faster convergence (as seen in Figure (11.9)). Moreover, we know that, in this case, it does exist continuity along the global minima of the function for different λ values, and this condition holds

because there is only one minimum for each λ and because that minimum must be a continuous function of λ. This property ensures that, starting from the global optimum of a given λ we rapidly converge to the optimum for a slightly higher λ. From here we derive a method similar to ℓ_0-GM but minimising the ℓ_1-norm. That is, we fix a small λ, do gradient descent with IST until reaching convergence, then fix a slightly higher λ, we apply again IST from the previous convergence point, and so on until reaching to the desired value of λ. We call this method ℓ_1-GM. Similarly to the ℓ_0-GM case, a faster and simpler approximation consists of increasing very slowly λ at each iteration, so the number of iterations is significantly reduced.

Figure 11.10 compares the convergence trajectories of IST, using for different thresholds, with respect to two ℓ_1-GM trajectories corresponding to exponentially decreasing the threshold using the rule $\theta^{(k)} = \theta^{(0)}\beta^k$ for two different β values. We can see that, in practice, the result of the exponential decrease with $\beta = 0{,}99$ needs less iterations ($1{,}5 \cdot 10^3$ vs. 10^3 for IST) to provide a quasi-optimal result for many values of λ, executing in total as many iterations as IST takes for only one sparseness value. Furthermore, using $\beta = 0{,}9$, we achieve a good approximation to the optimal result with a much more reduced number of iterations. Also in this case, as other authors (e.g., [42]), we have experienced that the exponential decay of the threshold provides a better compromise between computational cost and quality of the result than other decreasing functions, as linear, for example.

Figure 11.11 shows a family of fidelity-sparseness for different β values. We have also indicated the results of ℓ_1-AP as a reference. We can appreciate that the fidelity obtained with ℓ_1-GM approximates better the result of ℓ_1-AP as β gets closer to 1.

11.6.4. Practical advantages of ℓ_1-GM

We have observed that ℓ_1-AP is faster if the solution has a medium-high sparseness level (equivalently, a medium-low λ value), whereas ℓ_1-GM is better in case of having high λ values. Figure 11.12 shows a comparison, using *Barbara* image and 8-scale DT-CWT, of the iterations needed to reach a close to optimal result for different sparseness level by methods ℓ_1-AP (as described in Chapter 10) and ℓ_0-GM (with $\alpha = \theta$ and $\beta = 0{,}99$). Note that, for low sparseness level, ℓ_1-GM is faster. This case appears often in practice, when trying to look for exact sparse representations, or when solving image restoration problems, if some localised information has been lost (see Chapter 13). That is, when the goal is to match, total or partially, an observation.

11.7. Conclusions

In this chapter we have derived an optimisation method, based on iterative shrinkage and with dynamic adjusting of the threshold, to solve the sparse approximation problem. In contrast to existing heuristics (for example [17, 21, 19]) our approximation is fully justified in theory and it is formulated as a classical optimisation problem solution.

Our first step has been to reformulate the sparse approximation problem to obtain an equivalent optimisation problem, but using a constrained continuous function, instead of the discontinuous and unconstrained original cost function, which prevented us to apply classical optimization tools. Then, we have derived a solution for the problem by applying gradient descent on this function, and projecting each iteration onto the set of vectors holding the constraint, $S(\mathbf{\Phi}, \mathbf{x})$. The resulting method is a generalisation of IHT for the case $p = 0$. Finally, we have proposed the ℓ_0-GM method, which is based on a dynamic update of the threshold, while doing gradient descent of the cost function. This method has been justified as a type of deterministic annealing equivalent based on expressing the cost function as the result of convolving tan infinitely sharp reference cost function with a decreasingly smooth kernel.

Our experiments show that ℓ_0-GM is not only more efficient and requires less iterations than other methods (ℓ_0-AP, ℓ_1-AP+OP, IHT) but that it also provides much better compaction results. In fact, its performance, when the number of selected coefficients is close to the number of pixels in the image is close to optimal (asymptotical). This method is comparable to the state-of-the-art in sparse approximation performance. These results show that, under the practical conditions presented in this Thesis, trying to solve directly for the sparse approximation problem leads to better local minima than solving for the convex relaxation problem.

Analogously, we have derived generalised IST and ℓ_1-GM from the gradient descent in a constrained equivalent version of the cost function associated to the convex relaxation problem. Both methods provide the optimal solution to the problem (ℓ_1-GM when $\beta \to 1$). We have seen that using ℓ_1-GM is recommended for those applications where we are constrained to preserve some part (of all) of the observation.

We will use the same ideas presented here with other norms. Although the mathematics behind using intermediate quasi-norms ($0 < p < 1$) can be more complicated, but it would probably be worthy in order to improve the compaction result.

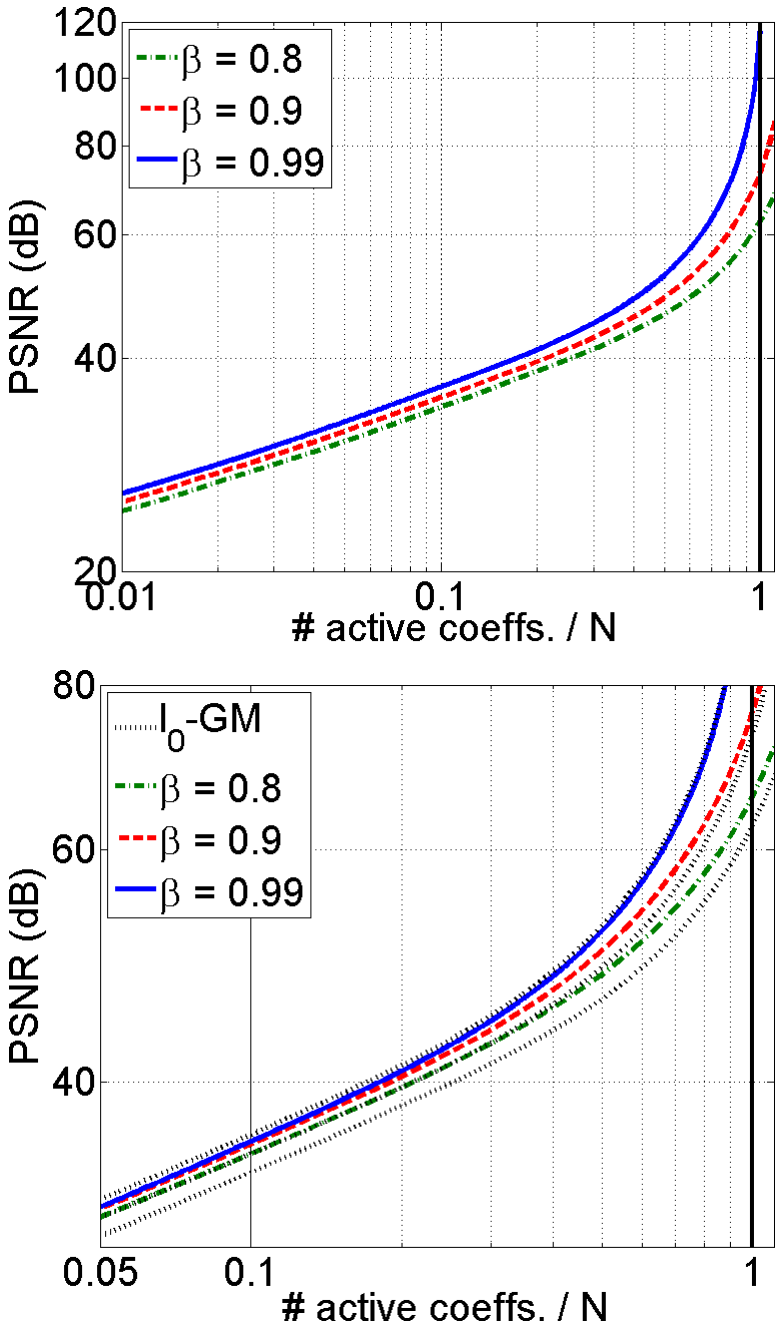

Figura 11.6: **Top**, *sparse approximation fidelity averaged in our test set using* ℓ_0*-GM with* $\alpha = 1{,}85(\theta^2/2)$*, three different* β *values and 8-scale DT-CWT.* **Bottom**, *quality of the reconstruction from the highest amplitude coefficients of the vector obtained using* ℓ_0*-GM for a very high* λ *value (very low sparseness), and the same* β *values. Dotted curves correspond to that of top panel. The vertical axis has been re-scaled to improve visibility.*

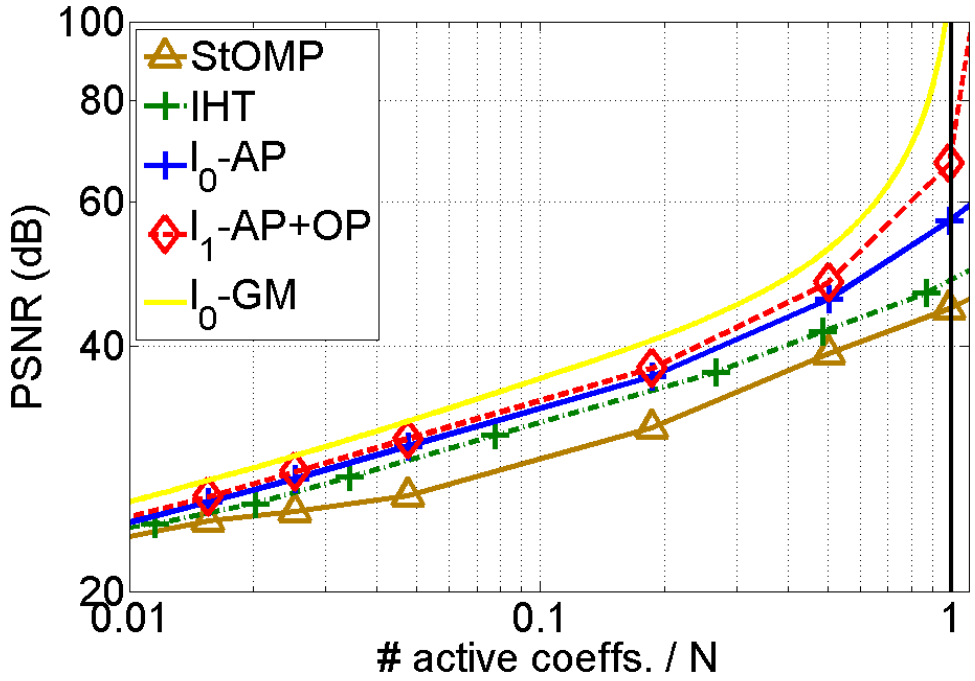

Figura 11.7: *ℓ_0-GM sparse approximation results averaged in our test set compared to other methods previously seen (StOMP, IHT, ℓ_0-AP and ℓ_1-AP+OP).*

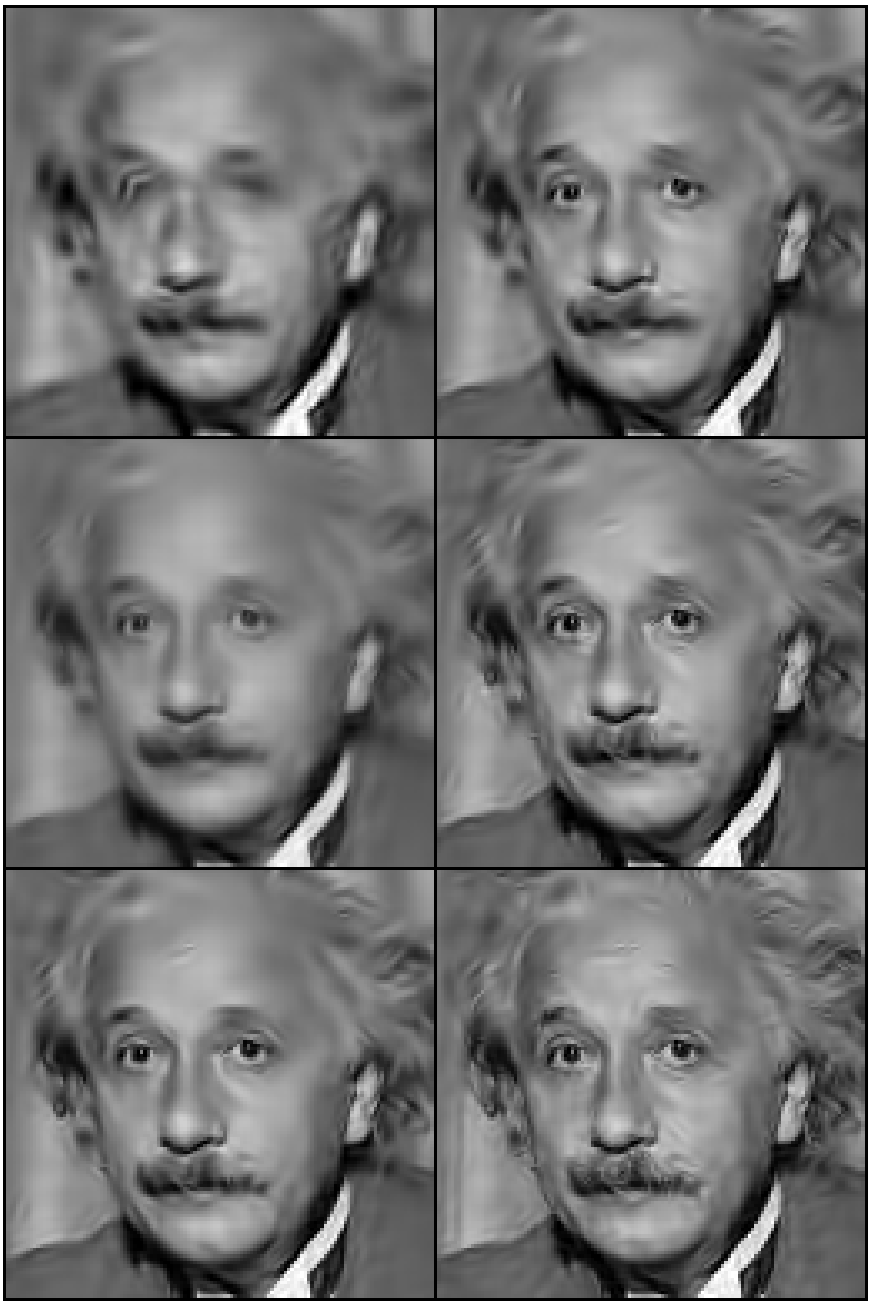

Figura 11.8: 64×64 *crop of the reconstruction of* Einstein *image using* $0{,}04 \cdot N$ *(*2605*) active DT-CWT coefficients, for several sparse approximation methods.* **Top-left**, *result of StOMP, implemented as described in Section 10.4.1 (*28,98 *dB).* **Top-right**, *IHT (*31,20 *dB).* **Centre-left**, ℓ_1*-AP (*29,70 *dB).* **Centre-right**, ℓ_0*-AP (*31,97 *dB).* **Bottom-left**, ℓ_1*-AP+OP (*32,38 *dB).* **Bottom-right**, ℓ_0*-GM (*33,28 *dB).*

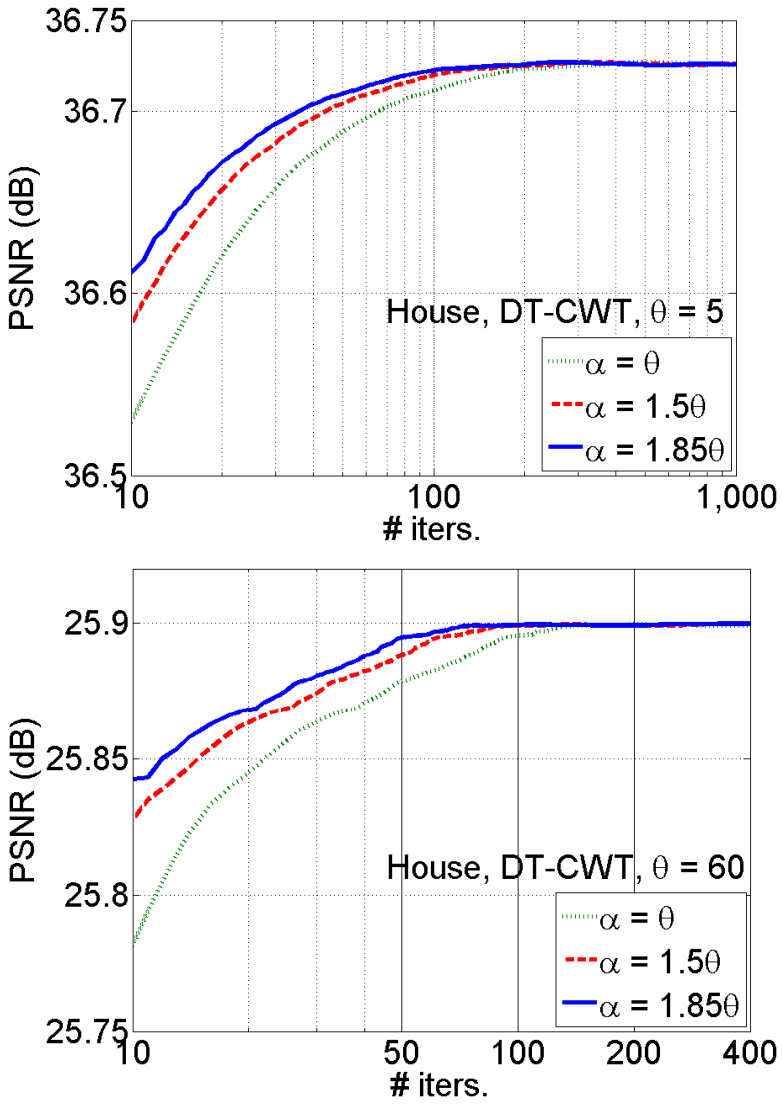

Figura 11.9: **Top**, *convergence curves for IST with a low threshold ($\theta = 5$) and three different α values. We have used* House *image and 8-scale DT-CWT.* **Bottom**, *same result for a higher threshold ($\theta = 60$).*

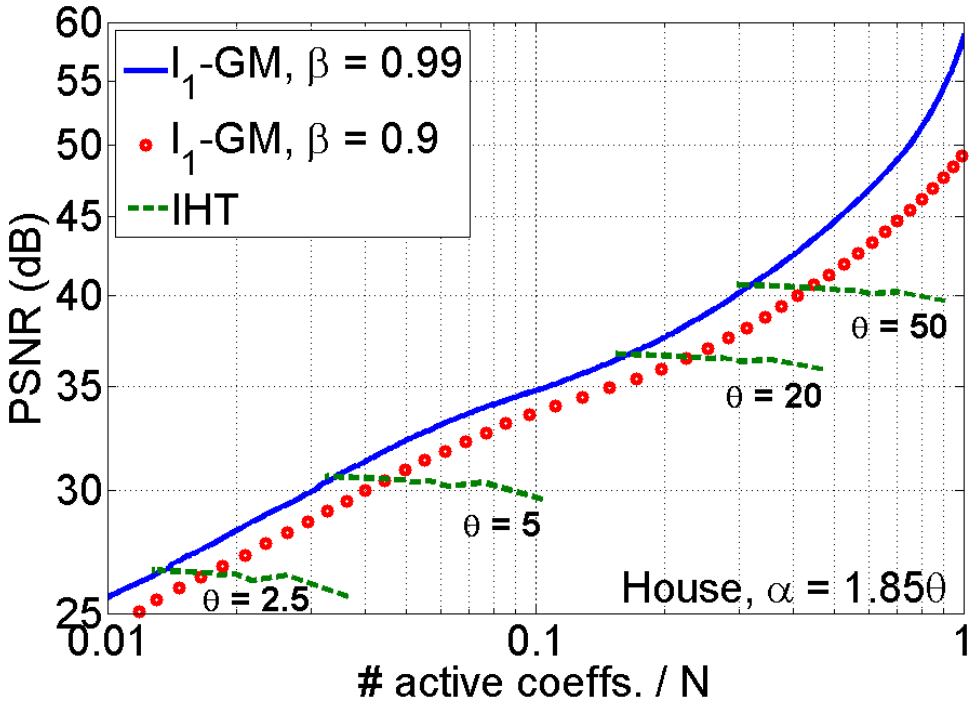

Figura 11.10: *Fidelity-sparseness results of ℓ_0-GM, using $\beta = 0{,}9$ (circles, $1{,}5 \cdot 10^2$ iterations) and $\beta = 0{,}99$ (solid, $1{,}5 \cdot 10^3$ iterations), compared to IHT, using several thresholds (dashed, 10^3 iterations). We use* House *image and DT-CWT with 8-scales.*

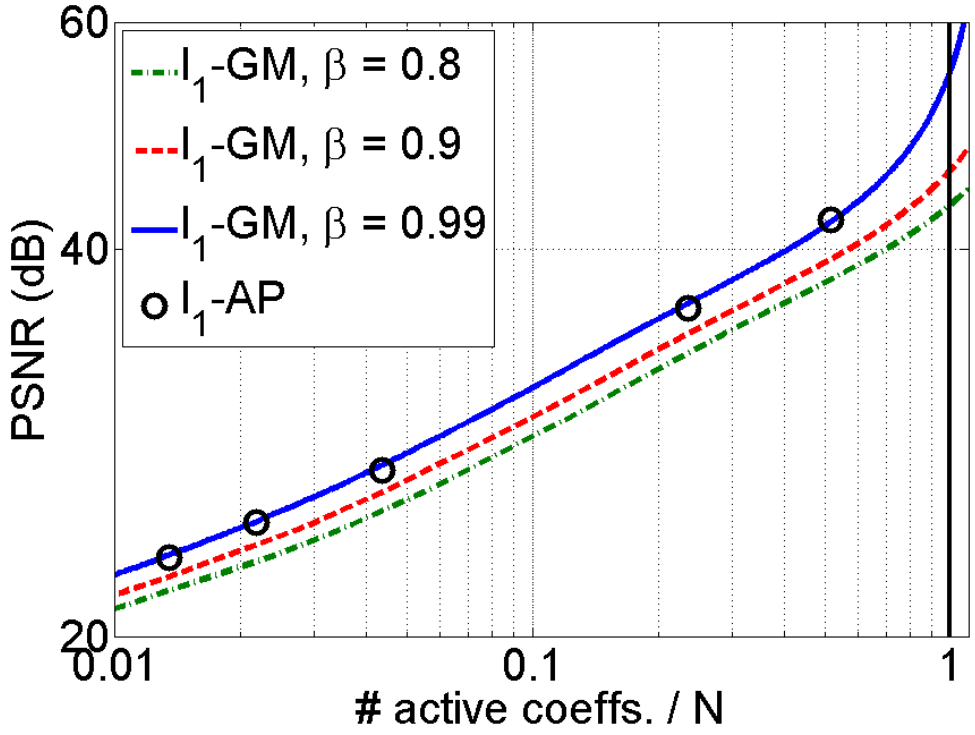

Figura 11.11: *Averaged sparse approximation results in the test set using ℓ_1-GM with $\alpha = 1{,}85\theta$, different β values and using DT-CWT with 8 scales. We also show the result of ℓ_1-AP.*

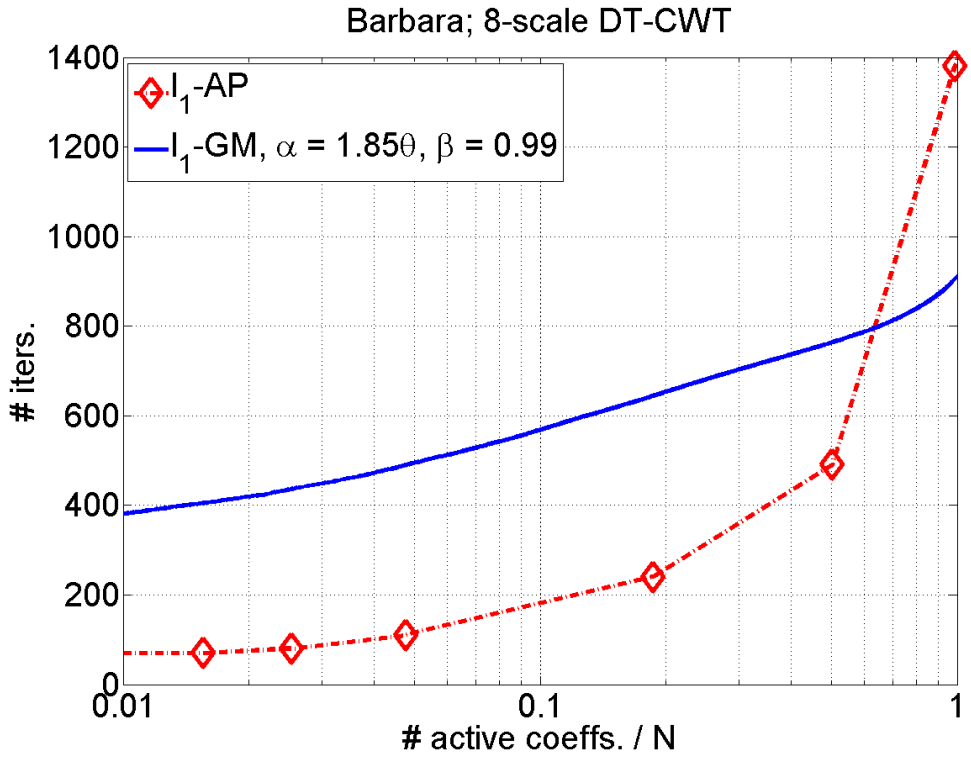

Figura 11.12: *Iterations needed to provide nearly optimal result for different sparseness level using ℓ_1-AP and ℓ_1-GM ($\alpha = 1{,}85\theta$, $\beta = 0{,}99$). We use* Barbara *image and 8-scale DT-CWT.*

Capítulo 12

Application to image restoration

We now consider that we have an incomplete observation. For example, it has lost some pixels, chromatic components, bits, resolution, etc. Our goal is estimating that missing information. We approach the problem by maximising the fidelity to the observation regularised by an *a priori* model based on statistical properties of natural images.

Our fidelity model is based on the concept of consistency. We say that an image is consistent with a degraded observation when, applying the same degradation to the image; we obtain again the given observation. Consequently, to apply this in practice, the degradation should be perfectly reproducible from the observed image. In some cases, it is not possible to identify the precise degradation suffered by some given observation (e.g., white Gaussian noise). But, in some others, it is possible (e.g., missing pixels, bits, chromatic components, resolution, etc.). We call the latter *a posteriori* deterministic degradations.

Our *a priori* model is based on favouring the sparseness of the estimation. This is justified by the observation that most of the degradations decrease the sparseness of the representation (e.g., wavelets) with respect to the original image [124, 125, 26]. Figure 12.1 shows an example. Left column corresponds to a crop of *Peppers* image (top) and a high-frequency sub-band of the linear response using DT-CWT with this image (bottom). Right column corresponds to randomly missing 40 % of the image pixels (top) and the corresponding sub-band (bottom). We note that the energy is less concentrated in the degraded sub-band.

The two observations made about natural images in Chapter 2 (energy compaction of the linear response and sparseness increase using non-linear methods) lead us to two different variants to describe the *a priori* knowledge

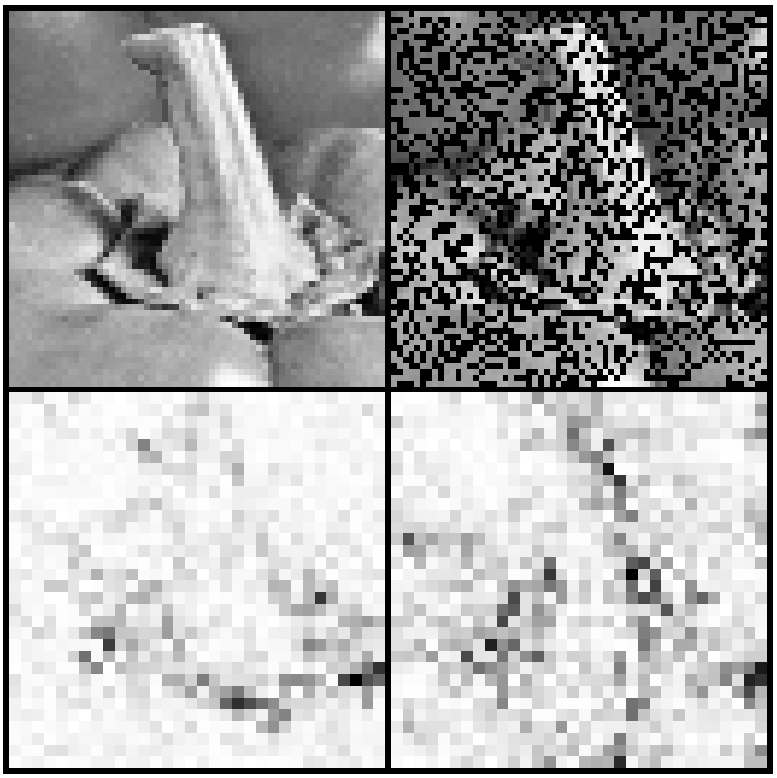

Figura 12.1: **Top-left**, Peppers *crop, starting at row* 111, *column* 91. **Bottom-left**, *same crop of the high frequency sub-band of the linear response to* Peppers *using* 8*-scales DT-CWT, corresponding to orientation* -45^o*. We have previously doubled the size of this sub-band, through pixel replication, in order to match the image size.* **Top-right**, *degraded image by setting to zero, randomly,* 40 % *of the pixels.* **Bottom-right**, *corresponding sub-band.*

we have about them. On the one hand, we can assume that the original image can be expressed as a linear combination of few representation vectors. This implies a sparse vector of synthesis coefficients. Many authors have previously used this concept to approach image restoration [46, 31, 66]. We call this Synthesis-sense Sparseness (SS).

Although the use of the SS approach is perfectly legitimate and reasonable successful in practice, one could object the lack of a direct empirical basis. The traditional Bayesian approximation to image restoration is based on building *a priori* models of the image reflecting the typical behaviour of the signals in many previous observations. However, the synthesis coefficients of an optimally sparse representation cannot be observed directly, and, often, they cannot be exactly calculated in practice (because, in general, the global optimum solution is not available).

Following this reasoning, it seems conceptually more consistent to use an

a priori statistical model based on direct observations, describing the typical distribution of the coefficients of the linear transformation of natural images. We call this Analysis-sense Sparseness (AS). This is a natural extension of many previous works that, under different points of view, have used sparse density models for the linearly transformed image (e.g., [80, 20]). Moreover, some authors have implemented practical methods based on AS for image processing, with very positive results (e.g., [21, 22]). As it will be seen in the next chapter, we have experienced a generally better restoration performance using AS than SS, in agreement with [126].

In this chapter we show how to apply the methods presented in previous chapters to restoration of *a posteriori* deterministic degradations. We have observed, for each method, which sparseness type is better in practice. Consequently, we have used ℓ_p-AP method for SS-restoration and ℓ_p-GM for AS.

We start by explaining and formulating the consistency set with a generic degraded observation (Section 12.1). Then we formulate the SS restoration problem (Section 12.2). Next we show how to adapt ℓ_p-AP to solve it (Section 12.3). Then we formulate the AS problem (Section 12.4), and show how to adapt ℓ_p-GM to solve it (Section 12.5).

12.1. Consistency with an observation

We have a degraded image $\mathbf{y} \in \mathbb{R}^N$. Consider that the degradation consists of missing some identifiable pieces of information (e.g., bits, pixels, chromatic components, etc.). This could be bits, pixels, chromatic components, etc. We assume that we can exactly know, given $\mathbf{y}$, which elements of the original image are preserved (*a posteriori* deterministic degradations). Then, we can replicate the associated degradation, noted as:

$$\mathbf{y} = f_{\mathbf{y}}(\mathbf{x}). \tag{12.1}$$

We define the consistency set for observation $\mathbf{y}$, $R(\mathbf{y})$, as all those images which, after being degraded with $f_{\mathbf{y}}(\mathbf{x})$, result in the same observation. Mathematically:

$$R(\mathbf{y}) = \{\mathbf{x} \in R^N : \ f_{\mathbf{y}}(\mathbf{x}) = \mathbf{y}\}.$$

12.2. Formulation using synthesis-sense sparseness

If we are dealing with a redundant representation domain, we have that:

$$\mathbf{y} = f_{\mathbf{y}}(\mathbf{\Phi a}),$$

where $\mathbf{a}$ is a synthesis vector whose reconstruction provides the original image. The *Maximum A Posteriori* estimate of $\mathbf{a}$ is given by:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\log p(\mathbf{a}) + \lambda\|\mathbf{y} - f_{\mathbf{y}}(\mathbf{\Phi a})\|_2^2\}.$$

where $p(\mathbf{a})$ is a prior for the representation coefficients. As it is common in the literature when using, as in our case, wavelets (e.g., [3, 80, 84, 31]), we assume independent coefficients and heavy-tailed priors, such as the Generalized Gaussian density :

$$p(\mathbf{a}) \propto \exp\{-k\|\mathbf{a}\|_p^p\}.$$

When $0 \leq p \leq 1$, this distribution is *sparse*, in the sense of having a probability density function concentrating most of the coefficients around zero, and having a small proportion of them with relatively high amplitudes. The logarithm of this prior is proportional to the p-th power of the ℓ_p-norm of the vector plus some irrelevant constant ($\log p(\mathbf{a}) \propto \|\mathbf{a}\|_p^p + A$). Then, our optimisation problem is set up as follows:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}}\{\|\mathbf{a}\|_p^p + \lambda\|\mathbf{y} - f_{\mathbf{y}}(\mathbf{\Phi a})\|_2^2\},$$

Note that this is analogous to the ℓ_p-norm minimisation problem in Equation (3.1), but the fidelity is measured in terms of the residual between the observation and the degradation of the estimation. Therefore, the cost function to be minimised is formed by the addition of two terms, one corresponding to the sparseness of the approximation, and the other to the quadratic distance to the consistency set $R(\mathbf{y})$. The parameter λ controls the relative importance of each term in the final solution. In practice, we require our estimation to belong to $R(\mathbf{y})$, same as the original image. Equivalently, we require $\hat{\mathbf{a}}$ to be inside the set $S(\mathbf{y})$ of synthesis vectors representing images in $R(\mathbf{y})$:

$$S(\mathbf{y}) = \{\mathbf{a} \in \mathbb{R}^M : \ \mathbf{\Phi a} \in R(\mathbf{y})\}.$$

Then, we set λ to infinite, which yields the following problem:

$$\begin{aligned} \hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} \|\mathbf{a}\|_p^p \text{ s.t. } \mathbf{a} \in S(\mathbf{y}), \\ \hat{\mathbf{x}} &= \mathbf{\Phi}\hat{\mathbf{a}}. \end{aligned} \tag{12.2}$$

12.3. Estimation using ℓ_p-AP and synthesis-sense sparseness

The solution to Equation (12.2) has a certain ℓ_p-norm, $\|\hat{\mathbf{a}}\|_p^p = R^*$. This value can be found by solving the problem:

$$R^* = \min\{R \in \mathbb{R}^* : \ B_p(R) \bigcap S(\mathbf{y}) \neq \emptyset\}.$$

The intersection between the corresponding sets, $B_p(R^*)$ and $S(\mathbf{y})$, will have more than one element in general. Among them, we choose the closest one to the observation:

$$\hat{\mathbf{a}} = P^{\perp}_{S(\mathbf{y}) \bigcap B_p(\hat{R}^*)}(\mathbf{\Phi}^T \mathbf{y}).$$

It is easy to see that we can use ℓ_p-AP (see Chapter 3) to solve this problem, only substituting the set $S(\mathbf{\Phi}, \mathbf{x})$ with the set $S(\mathbf{y})$. Then, we obtain the following iterations:

$$\begin{aligned} \hat{\mathbf{a}}^{(0)} &= P^{\perp}_{B_p(\hat{R}^*)}(\mathbf{a}^{LS}), \\ \hat{\mathbf{a}}^{(k+1)} &= P^{\perp}_{B_p(\hat{R}^*)}(P^{\perp}_{S(\mathbf{y})}(\hat{\mathbf{a}}^{(k)})). \end{aligned} \tag{12.3}$$

Iterations end when $\|\hat{\mathbf{a}}^{(k+1)} - \hat{\mathbf{a}}^{(k)}\|_2 < \delta$, for $\delta > 0$. The proof that the fixed point of these iterations is a local minimum to the distance to $S(\mathbf{y})$ is completely analogous to that shown in Section 3.1 for the case of sparse approximation.

We derive now the expression of the orthogonal projection of a vector $\mathbf{b}^o$ onto $S(\mathbf{y})$:

$$\hat{\mathbf{b}}^p_{S(\mathbf{y})} = P^{\perp}_{S(\mathbf{y})}(\mathbf{b}^o) = \arg\min_{\mathbf{b}}\{\|\mathbf{b} - \mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{b} \in S(\mathbf{y})\}. \tag{12.4}$$

$S(\mathbf{y})$ is orthogonal to the set A of linear responses to images, defined as:

$$A = \{\mathbf{b} : \ \exists \mathbf{x} \in \mathbb{R}^N, \ \mathbf{\Phi}^T \mathbf{x} = \mathbf{b}\}.$$

Then, analogously to the case of the affine space of perfect reconstruction of an image (see Equation (3.4)), we have that:

$$\hat{\mathbf{b}}^p_{S(\mathbf{y})} = \mathbf{b}^o + \mathbf{\Phi}^T \mathbf{\Phi}[\hat{\mathbf{b}}^p_{S(\mathbf{y})} - \mathbf{b}^o].$$

We now define $S_A(\mathbf{y})$ as the set of linear responses whose reconstruction is consistent with the observation:

$$S_A(\mathbf{y}) = \{\mathbf{b} \in R^M : \ \exists \mathbf{x} \in R(\mathbf{y}), \ \mathbf{\Phi}^T \mathbf{x} = \mathbf{b}\}.$$

We have that $\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = P^{\perp}_{S_A(\mathbf{y})}(\hat{\mathbf{b}}^p_{S(\mathbf{y})}) = \mathbf{\Phi}^T\mathbf{\Phi}\hat{\mathbf{b}}^p_{S(\mathbf{y})}$, and thus:

$$\hat{\mathbf{b}}^p_{S(\mathbf{y})} = \mathbf{b}^o + \hat{\mathbf{b}}^p_{S_A(\mathbf{y})} - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o. \tag{12.5}$$

To solve $\hat{\mathbf{b}}^p_{S_A(\mathbf{y})}$, we have to derive an expression of the orthogonal projection onto $S_A(\mathbf{y})$ in terms of our observation $\mathbf{b}^o$. We have that:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \arg\min_{\mathbf{b}}\{\|\mathbf{b}^o - \mathbf{b}\|_2^2 \text{ s.t. } \mathbf{b} \in S_A(\mathbf{y})\}.$$

We can express:

$$\mathbf{b}^o - \mathbf{b} = (\mathbf{b}^o - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o) + (\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o - \mathbf{b}).$$

These two bracketed differences are orthogonal vectors, as the first one belongs to the null space of $\mathbf{\Phi}$, whereas the second one has no null component in $\mathbf{\Phi}$ (that is, $\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b} = \mathbf{b}$, because $\mathbf{b} \in S_A(\mathbf{y})$). Then, we can write:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \arg\min_{\mathbf{b}}\{\|\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o - \mathbf{b}^o\|_2^2 + \|\mathbf{b} - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{b} \in S_A(\mathbf{y})\}.$$

As the first summation term is independent from $\mathbf{b}$, it can be ignored in the minimisation, resulting in:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \arg\min_{\mathbf{b}}\{\|\mathbf{b} - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{b} \in S_A(\mathbf{y})\}.$$

We know that, for every vector $\mathbf{b} \in S_A(\mathbf{y})$, another vector $\mathbf{x} \in R(\mathbf{y})$ exists such that $\mathbf{b} = \mathbf{\Phi}^T\mathbf{x}$. Thus, substituting in previous expression, we get:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \mathbf{\Phi}^T[\arg\min_{\mathbf{x}}\{\|\mathbf{\Phi}^T\mathbf{x} - \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{x} \in R(\mathbf{y})\}].$$

And because $\mathbf{\Phi}^T$ is a Parseval frame:

$$\hat{\mathbf{b}}^p_{S_A(\mathbf{y})} = \mathbf{\Phi}^T[\arg\min_{\mathbf{x}}\{\|\mathbf{x} - \mathbf{\Phi}\mathbf{b}^o\|_2^2 \text{ s.t. } \mathbf{x} \in R(\mathbf{y})\}].$$

The minimisation in $\mathbf{x}$ corresponds to the orthogonal projection of $\mathbf{\Phi}\mathbf{b}^o$ onto the set of images consistent with the observation, $P^{\perp}_{R(\mathbf{y})}(\mathbf{\Phi}\mathbf{b}^o)$, so we obtain:

$$P^{\perp}_{S_A(\mathbf{y})}(\mathbf{b}^o) = \mathbf{\Phi}^T P^{\perp}_{R(\mathbf{y})}(\mathbf{\Phi}\mathbf{b}^o). \tag{12.6}$$

And, substituting in Equation (12.5) we finally have that:

$$P^{\perp}_{S(\mathbf{y})}(\mathbf{b}^o) = \mathbf{b}^o + \mathbf{\Phi}^T(P^{\perp}_{R(\mathbf{y})}(\mathbf{\Phi}\mathbf{b}^o) - \mathbf{\Phi}\mathbf{b}^o).$$

Finding the orthogonal projection onto the consistency set $R(\mathbf{y})$ is trivial for a wide number of strictly reproducible *a posteriori* degradations, by simply forcing the reconstruction to preserve the desired values. Of course, the precise form of this projection depends on each degradation. We explain some cases in detail in Chapter 6.

12.4. Formulation using analysis-sense sparseness

We cannot reach, in general, strict sparseness when dealing with linear responses of natural images, because we cannot avoid the simultaneous response of several coefficients to the same feature. Instead, we can consider that most of the energy is concentrated in a small proportion of coefficients. Then, we can model the linear representation as a strictly sparse vector whose support corresponds to the highest responses in amplitude, called $\mathbf{a}$, plus a Gaussian correction term, noted $\mathbf{r}$. Then, if we define $S_A(\mathbf{y})$ as the set of linear responses whose reconstruction is consistent with the observation:

$$S_A(\mathbf{y}) = \{\mathbf{b} \in R^M : \exists \mathbf{x} \in R(\mathbf{y}),\ \mathbf{\Phi}^T\mathbf{x} = \mathbf{b}\},$$

then, we can write our optimization problem as:

$$\begin{aligned} (\hat{\mathbf{a}}, \hat{\mathbf{r}}) &= \arg\min_{\mathbf{a},\mathbf{r}}\{\|\mathbf{a}\|_p^p + \lambda\|\mathbf{r}\|_2^2 \text{ s.t. } (\mathbf{a}+\mathbf{r}) \in S_A(\mathbf{y})\}, \\ \hat{\mathbf{x}} &= \mathbf{\Phi}(\hat{\mathbf{a}} + \hat{\mathbf{r}}). \end{aligned} \tag{12.7}$$

12.5. Estimation using ℓ_p-GM and analysis-sense sparseness

Following a completely parallel way to that of the sparse approximation problem, when solving Equation (12.7) we derive an expression which only depends on a vector $\mathbf{b} = \mathbf{a} + \mathbf{r}$, and where the constraint set, $S(\mathbf{\Phi}, \mathbf{x})$, is substituted by the new constraint, $S_A(\mathbf{y})$. Note that this new set is no longer affine and that, therefore, we have to consider its curvature by projecting the cost function gradient onto its tangent hyperplane on every border point $\mathbf{b}$. This projection can be calculated as the limit:

$$\nabla^{S_A(\mathbf{y})} C_p(\mathbf{b}, \theta) = \lim_{\alpha \to 0} \frac{P^{\perp}_{S_A(\mathbf{y})}(\alpha \nabla C_p(\mathbf{b}, \theta))}{\alpha},$$

where $P^{\perp}_{S_A(\mathbf{y})}$ is the orthogonal projection onto $S_A(\mathbf{y})$ (Equation (12.6)), and where $C_p(\mathbf{b}, \theta)$ indicates, with $p = 0$ or $p = 1$, the cost functions defined in Equations (4.5) and (4.11), respectively. The gradient descent method is then formulated as:

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} - \alpha \nabla^{S_A(\mathbf{y})} C_p(\mathbf{b}^{(k)}, \theta). \tag{12.8}$$

However, it is more convenient in practice to use the following simpler calculations in the estimation loop:

$$\mathbf{b}^{(k+1)} = P^{\perp}_{S_A(\mathbf{y})}\left(\mathbf{b}^{(k)} - \alpha \nabla C_p(\mathbf{b}^{(k)}, \theta)\right),$$

which ensures that the updated vector belongs to the consistency set $S_A(\mathbf{y})$ for any value of α. This update rule is equivalent to that of Equation (12.8) if the projection is linear.

Because of the similar structure of the minimisation problem described in Equation (4.2), and that described in Equation (12.7), we can apply the same strategy to look for a global minimisation of the cost function when $p = 0$ or $p = 1$. This implies, in practice, that we get a significatively better restoration performance if we use an exponentially decaying threshold until reaching the desired value, and then we use that fixed threshold until convergence. We have empirically tested in different applications that the optimal final threshold in our optimisation is usually close to zero, as it was also indicated by [21, 22]. Thus, in the absence of any additional information source, an arbitrarily small threshold[1] is a suitable stopping criterion for the iterations.

[1]Too low thresholds demand more computation, so here there is again a trade-off between time and quality.

Capítulo 13

Some applications

In this chapter we present several applications to restoration problems of the methods derived in this Thesis. We focus on *a posteriori* deterministic degradations (see Chapter **??**). We motivate each problem and formulate the associated consistency set and the orthogonal projection onto it. Finally, we present some experiments showing that our methods are highly competitive for the applications studied.

In Section 13.1 our sparse approximation methods are applied to remove spatial quantisation artifacts. In Section 13.2 they are applied to estimation of missing pixels of the image. In Section 13.3 we study the interpolation of Bayer Color Filter Array mosaics. Finally, in Section 13.4 we approach the problem of increasing the details of images.

13.1. Removing quantisation artifacts

13.1.1. Introduction

Spatial quantisation is an indispensable part of the capture of images with digital devices. Usually, the artifacts derived from it, as false contours and destruction of low-contrast texture, are close or even below the visibility threshold. However, in a number of situations they can become evident. For example, when the local luminance range of an image is stretched to inspect low-contrast details, or when blurry and quantised images are deconvolved, especially if there is little noise from other sources. It can also be useful as a previous step to extract sensitive local features, like the gradient of the luminance. Other possible applications are interpolating level curves in topographic or barometric maps, or using a reduced number of bits per pixel when there are not enough resources to perform a more advanced compression of the image.

Surprisingly enough, until very recently the removal of quantisation artifacts in the image domain (from now on, de-quantising) has received little attention in the scientific literature. In contrast, the quantisation in the transformed domain has been widely used, specially in the context of post-processing compressed images (e.g., [127, 128, 129, 130, 131]).

Nevertheless, during the last years there has been a growing interest in approaching the problem in the image domain. Up to our knowledge, the first work was [132], which used de-quantising as a previous step to edge detection. Recently, other methods have been published, based on iterating between some filtering operation and the correction of the difference with the original [133, 134]. But this type of strategies, though they result in efficient algorithms, are too simple to provide satisfactory results. In parallel to these works, we presented a method based on promoting the sparseness on a representation with redundant wavelets [26]. The selection of coefficients was made, in this method, by directly thresholding the analysis wavelet coefficients, so we can classify this technique within the greedy heuristics. Last three referred methods are described in this section in detail.

In this section we compare the performance of ℓ_p-AP adapted to de-quantising in the two versions presented, $p = 1$ and $p = 0$. We see, through exhaustive experiments, that ℓ_0-AP significantly outperforms methods in [26, 133, 134], and also ℓ_1-AP.

13.1.2. Consistency set

In this case the consistency set is made of those images that, when quantised using the same observed quantization levels, result in the same observation. Therefore, being $\mathbf{y}$ a quantised observed image, the consistency set associated to it, noted as $R_Q(\mathbf{y})$, is defined as:

$$R_Q(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \; y_i - \frac{\delta_i}{2} < x_i \leq y_i + \frac{\delta_i}{2}, \; \forall i \in \{1, ..., N\}\},$$

where δ_i indicates the size of the quantisation interval associated to each pixel[1].

Given an image $\mathbf{x} \in \mathbb{R}^N$, the orthogonal projection onto $R_Q(\mathbf{y})$ is easily computed as $\mathbf{z} = P^{\perp}_{R_Q(\mathbf{y})}(\mathbf{x})$, where:

$$z_i = \begin{cases} x_i, & y_i - \frac{\delta_i}{2} < x_i \leq y_i + \frac{\delta_i}{2} \\ y_i - \frac{\delta_i}{2} + \epsilon, & x_i \leq y_i - \frac{\delta_i}{2} \\ y_i + \frac{\delta_i}{2}, & y_i + \frac{\delta_i}{2} < x_i, \end{cases}$$

[1]Here we have assumed uniform quantisation, for simplicity, but all the methods described here are easily applicable to other types of quantisation.

where $\epsilon \in \mathbb{R}^*$ (ideally infinitesimal) is an artifice added to achieve empty intersection between closed adjacent quantisation intervals.

13.1.3. Implementation

In the experiments shown in next subsection we have compared our methods with three recent algorithms applied to quantisation artifacts removal. We briefly describe next these three methods and our implementation of them. The values given to the different parameters have been hand-optimised for every method to stop the iterations at a similar approximation level to the final convergence.

Direct thresholding and optimisation. In [26] we describe a method for de-quantizing in the image domain, by enforcing a high degree of sparseness in its representation with a redundant wavelet-based dictionary. For this purpose we devise a linear operator that returns the minimum ℓ_2-norm image preserving a set of significant coefficients, and estimate the original by minimizing the cardinality of that subset, always ensuring that the result is compatible with the quantized observation. We implement this solution by alternated projections onto convex sets. To select the set of significant coefficients, we threshold directly the amplitudes of the linear representation of the image, using a threshold proportional to the estimated energy of each original sub-band.

This application is based on the method that we call Direct Thresholding and Optimisation (DT+OP, see Chapter 10), and by extension this will be the name given to it here. Its details can be seen in [26]. In the following experiments we have used 7-scale DT-CWT and we have assumed that there is intersection between the sets when, in less than 30 iterations, the mean square difference of the subsequent projected vectors onto both sets is less or equal to 0,5.

Constrained Diffusion, In [134] the method of Constrained Diffusion (CD) is presented, based on combining linear filtering with non-linear correction of the difference with the observation.

Our implementation of the method follows the steps explained in [134], that is, an iterative method initialising the estimate with the observation and having two steps: 1) Convolution of the estimated image with the following matrix:

$$\begin{pmatrix} 0 & \frac{1}{5} & 0 \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ 0 & \frac{1}{5} & 0 \end{pmatrix}, \tag{13.1}$$

and 2) projection onto the consistency set with the observation. The

iterations ed when the mean square difference of the estimation of one iteration with respect to the previous one is less than 0,5.

Regularised Constrained Iterative Restoration. In [133] it is described a method to remove quantisation artifacts, called Regularised Constrained Iterative Restoration (RCIR). The method is described applied to vector quantisation. However, when dealing with uniform quantisation, as in our case, it is reduced to a simple strategy based on minimising the ℓ_2-norm of the output of a high-pass filter, enforcing, at each step, again that the estimation belongs to the consistency set.

Our implementation results in an iterative method, initialised with the observation at the first estimation. It consists of two steps: 1) Subtract to the estimation its own convolution with a Laplacian spatial filter. 2) Project onto the consistency set. The stopping criterion for the iterations is similar to that of CD.

ℓ_p-AP. We use a binary search to find the radius of the smaller ℓ_p-ball having non-empty intersection with the set of vectors consistent with the observation (see Chapter **??**), whose interval is initialised between 0 and M in ℓ_0-AP and between 0 and $\|\mathbf{\Phi}^T\mathbf{y}\|_1$ in ℓ_1-AP. We consider that we have found the required radius when the search interval has a length less or equal to 5000, and we assume that it exists intersection when in less than 30 iterations the mean square difference between the projected vector onto one of the sets and the other is less or equal to[2] 0,3.

13.1.4. Results and discussion

13.1.4.1. ℓ_0-AP vs. ℓ_1-AP

Figure 13.1 compares the de-quantising performance of ℓ_0-AP vs. ℓ_1-AP, using 3-bits quantisation of *Einstein* image. Results are shown using both 8-scale DT-CWT and 6-scale Curvelets. We observe that the performance of ℓ_0-AP is much better, both in PSNR increase and visually: it manages to remove the quantisation discontinuities but preserves a high definition in the original edges of the image. We also see that results using Curvelets are slightly better in this case than those obtained using DT-CWT.

Figure 13.2 shows a similar example with a little texture image (*Peppers*). Note that the relative behaviour between the methods is qualitatively similar, although the difference in visual quality between ℓ_1-AP and ℓ_0-AP is not so big in this case as in the *Einstein* example. This is due to this image has less high-frequency texture, and then over-smooth

[2]The higher tolerance given to methods based on linear filtering favour them, because making more iterations will smooth too much the estimation, thus decreasing the PSNR.

estimations are having little visual impact. However, the difference in PSNR is still favourable to ℓ_0-AP. Note also that now the performance is better using DT-CWT than Curvelets.

Regarding the behaviour of ℓ_p-GM, we have experienced that it is not very satisfactory for both $p = 0$ and $p = 1$. Using AS it is not able to remove the quantisation artifacts; and using SS too smooth estimations are obtained. In the latter case, the compaction capacity of $\ell_0 - GM$ is much better than that of ℓ_0-AP, but this does not reflect in a parallel way on the method's performance.

13.1.4.2. ℓ_0-AP vs. existing methods

Now we compare the ℓ_0-AP performance vs. that of the methods previously described: RCIR [133], CD [134], and DT+OP [26]. Table 13.1 shows the performance average (obtaining the MSE for the average) of each method in our test set (see Appendix B) for all the range of possible quantisation bits in 8-bits images. We can see that methods based on enforcing sparseness, DT+OP and ℓ_0-AP, outperform clearly those based on simpler linear operations, except when the image is quantised with only one bit. Note that ℓ_0-AP is the best for those levels which, in practice, result in visible artifacts (low and medium range).

	PSNR (dB)						
♯ Bits	1	2	3	4	5	6	7
Observed	16,40	22,73	29,22	34,71	40,59	46,45	51,11
RCIR	*17,62*	24,25	29,75	34,37	39,20	44,56	51,87
CD	**17,88**	*24,33*	29,60	34,38	39,30	44,65	*51,89*
DT+OP	16,46	24,05	*30,83*	*35,67*	**40,61**	**45,28**	49,27
ℓ_0-AP	17,29	**24,74**	**31,49**	**35,91**	*40,21*	*45,11*	**52,03**

Cuadro 13.1: *PSNR (MSE averaged) using the images in our test set, quantised with all the possible range of bits and restored using methods RCIR, CD, DT+OP and ℓ_0-AP. First row corresponds to the averaged PSNR of the observed images. Bold numbers indicate the best result for each number of bits, and italic the second best.*

In addition, the visual appearance of the results of methods based on promoting sparseness is significantly better than those of their competitors, even for low number of quantisation bits. In Figures 13.3 and 13.4 we can see a visual comparison of the application of the methods to *Einstein* and *Peppers* images quantised with 3-bits. Among them, the best one, both visually and in PSNR, is ℓ_0-AP. Both RCIR and CD destroy too many high-frequency components without removing completely the artifacts. Using an

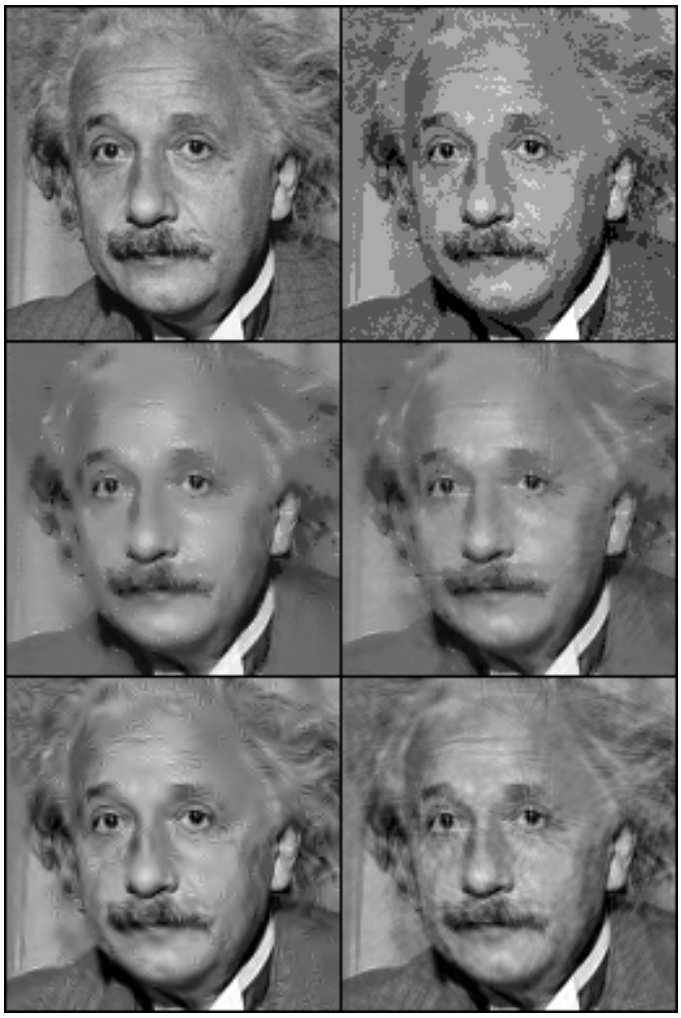

Figura 13.1: *Example of application of ℓ_1-AP and ℓ_0-AP to de-quantizing.* **Top-left**, *original* Einstein *image, cropped to* 128×128 *pixels.* **Top-right**, 3*-bits observed quantisation (PSNR:* 27,98 *dB).* **Centre-left**, ℓ_1*-AP result using* 8*-scale DT-CWT (*30,17 *dB).* **Centre-right**, ℓ_1*-AP result using* 6*-scale Curvelets (*30,61 *dB).* **Bottom-left**, ℓ_0*-AP result using* 8*-scale DT-CWT (*31,21 *dB).* **Bottom-right**, ℓ_0*-AP result using* 6*-scale Curvelets (*31,38 *dB).*

adaptive threshold allows DT+OP to remove isolated elementary functions (present in ℓ_0-AP). However, presence of *ringing* effect and poor performance in areas of high-frequency texture contribute to the fact that the visual effect (and also the PSNR) is better in ℓ_0-AP. Finally, we show in the last panel the result of ℓ_0-AP using a joint DT-CWT - Curvelets representation (see Appendix F). Note that increasing the richness of the dictionary, not only in number but also in type of elementary functions used, significantly

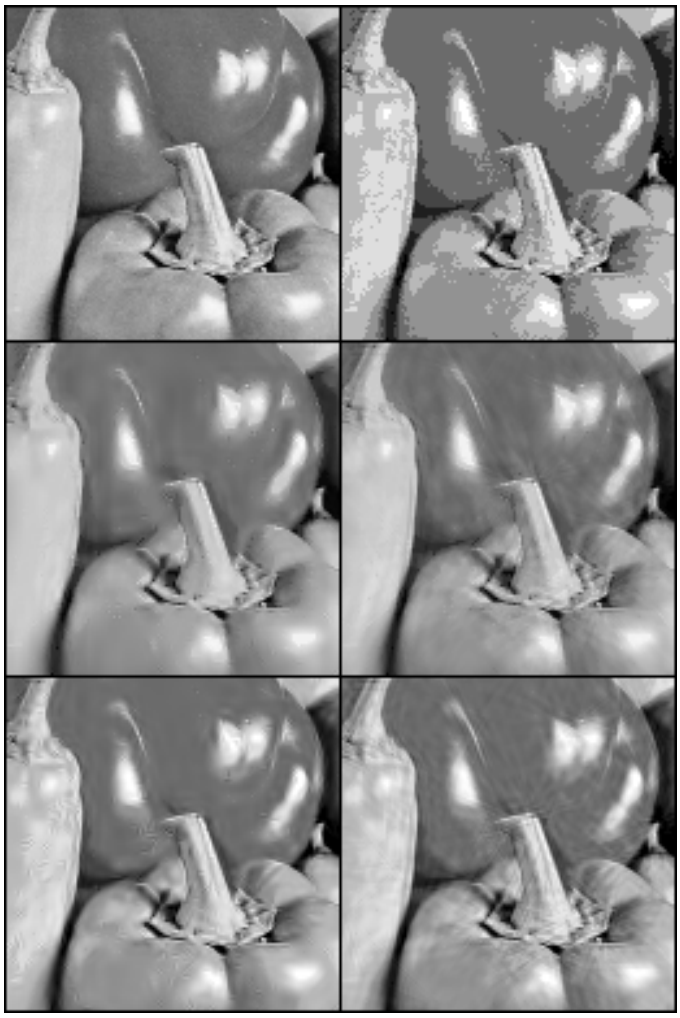

Figura 13.2: *Example of application of ℓ_1-AP and ℓ_0-AP to de-quantizing.* **Top-left**, *original* Peppers *image, cropped to* 128×128. **Top-right**, 3*-bits observed quantisation (PSNR:* 28,81 *dB).* **Centre-left**, ℓ_1*-AP result using* 8*-scale DT-CWT (*29,08 *dB).* **Centre-right**, ℓ_1*-AP result using* 6*-scale Curvelets (*29,50 *dB).* **Bottom-left**, ℓ_0*-AP result using* 8*-scale DT-CWT (*31,06 *dB).* **Bottom-right**, ℓ_0*-AP result using* 6*-scale Curvelets (*30,85 *dB).*

improves the result, increasing the PSNR and drastically reducing the isolated elementary functions appearing in the estimated image when only one representation is used. We have chosen these two dictionaries to fairly compare with the results shown in 13.1 using each representation separately. Moreover, using other dictionaries we can further improve the results. For example, using Curvelets and a version of the Steerable Pyramid [135] without high-pass-residual, the PSNR is 31,99 dB for *Einstein* and 31,46

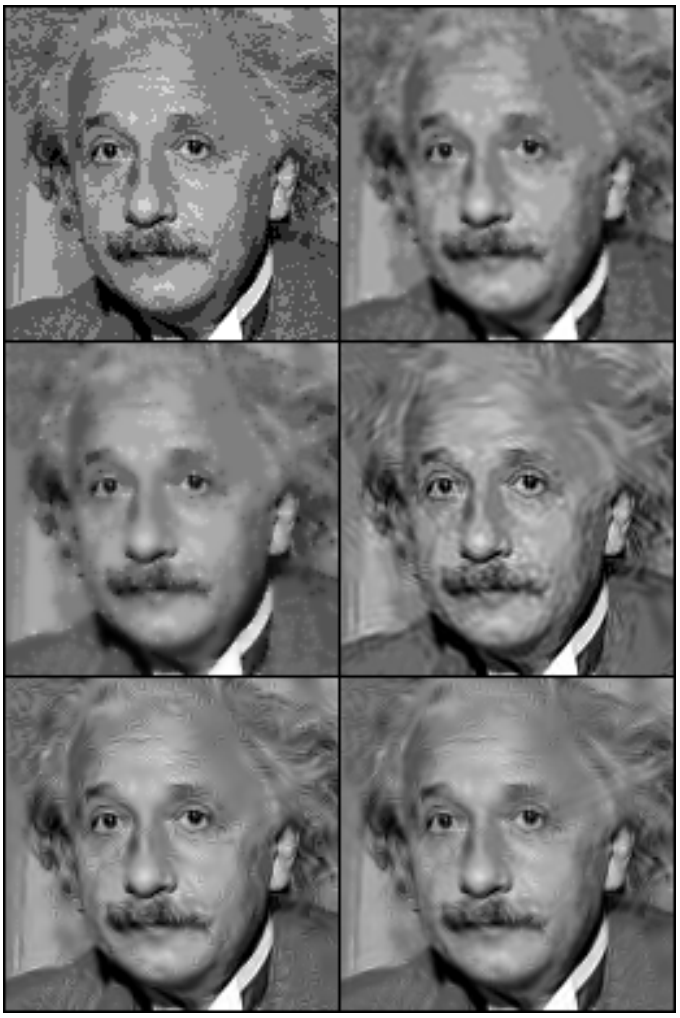

Figura 13.3: **Top-left**, Einstein *quantised with* 3 *bits and cropped to* 128×128, *(PSNR:* 27,98 *dB).* **Top-right**, *RCIR result (*30,39 *dB).* **Centre-left**, *CD result (*30,44 *dB).* **Centre-right**, *DT+OP result using DT-CWT with* 8 *scales (*30,72 *dB).* **Bottom-left**, ℓ_0*-AP result using DT-CWT with* 8 *scales (*31,21 *dB).* **Bottom-right**, ℓ_0*-AP result using jointly* 8*-scale DT-CWT and* 6*-scale Curvelets, with equal scale factor,* $\sqrt{\frac{1}{2}}$ *(*31,93 *dB).*

dB for *Peppers*.

Note, in Table 13.1, that for a high number of quantisation bits, there is a relative decrease in PSNR when using ℓ_0-AP. However, in these cases this method also manages to remove low-contrast artifacts, improving the visual appearance when enhancing the contrast of the image. Figure 13.5 shows an example. Left panel is a 32×32 detail of a smooth area in a photographic 8-bits image, with a contrast amplification factor around 40 times. Right

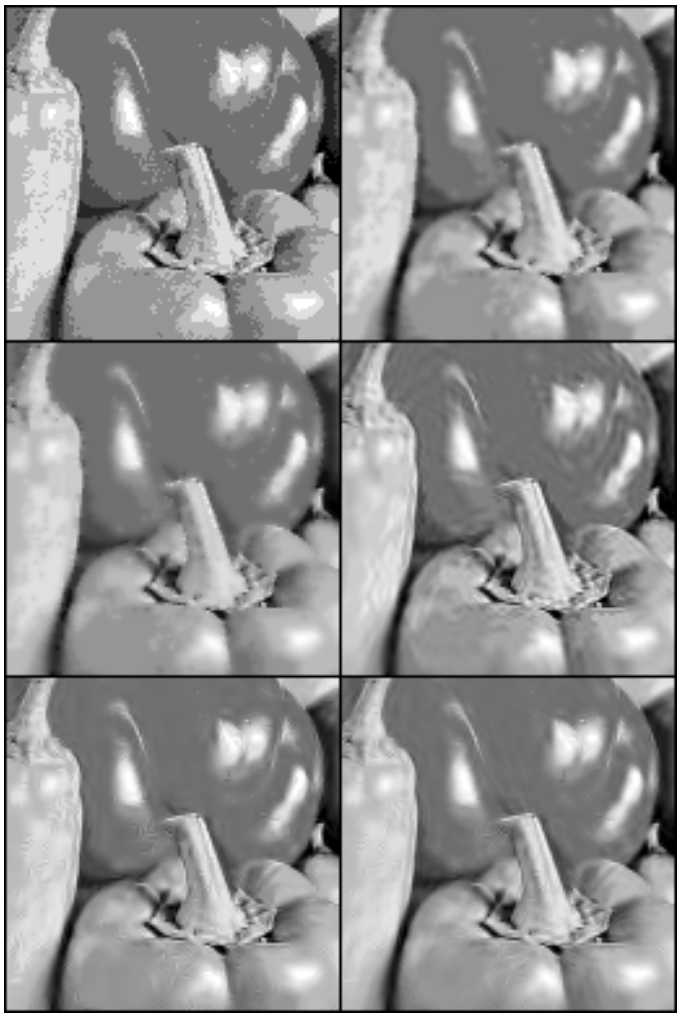

Figura 13.4: **Top-left**, *Peppers quantised with* 3 *bits and cropped to* 128×128, *(PSNR:* 28,81 *dB).* **Top-right**, *RCIR result (*29,65 *dB).* **Centre-left**, *CD result (*29,85 *dB).* **Centre-right**, *DT+OP result using DT-CWT with* 8 *scales (*30,38 *dB).* **Bottom-left**, ℓ_0*-AP result using* 8*-scale DT-CWT (*31,07 *dB).* **Bottom-right**, ℓ_0*-AP result using jointly* 8*-scale DT-CWT and* 6*-scale Curvelets, with equal scale factor,* $\sqrt{\frac{1}{2}}$ *(*31,46 *dB).*

panel is the same crop in the result after processing with ℓ_0-AP. Note the more natural appeareance.

Regarding computation time, because of their simplicity, RCIR and CD are clearly faster. They are iterative methods only requiring a convolution and a projection onto the consistency set at each iteration. The methods based on promoting sparseness are dominated by one analysis and one synthesis operation per iteration, and they also have to look for the value of

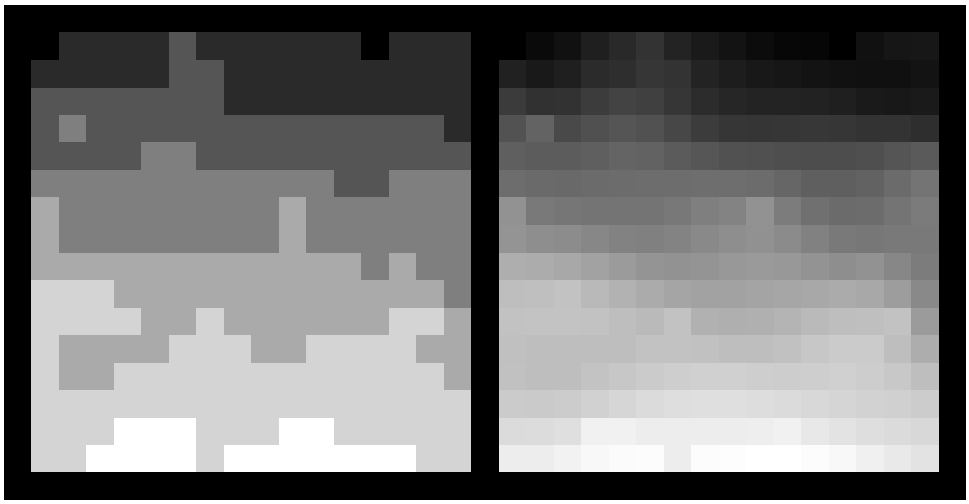

Figura 13.5: **Left**, *detail of the sky of a photographic* 8*-bits image with contrast amplified approximately* 40 *times.* **Right**, *same detail after processing with* ℓ_0*-AP.*

the threshold preserving a certain number of coefficients. Thus, both RCIR and CD perform around 10 iterations for each estimation, taking barely half a second in average, for a 256×256 image. DT+OP takes between 15 seconds and 5 minutes using DT-CWT with 8 scales, depending on the number of quantisation bits of the observed image. Finally, ℓ_0-AP takes 80 seconds on average using DT-CWT with 8 scales. Times are for our MATLAB® implementatoin on an Intel®, $Core^{TM}2$ Duo with 1,66 GHz and 2 GB RAM.

13.1.5. Conclusions

We have analysed the performance of ℓ_0-AP applied to removing spatial quantisation artifacts, comparing it to ℓ_1-AP and to other recently published methods (RCIR, DT+OP CD). We can say that the search for the sparsest image within the consistency set made by ℓ_0-AP provides very satisfactory results. In general, we have seen that methods based on promoting sparseness outperform those based on linear filtering operations. However, ℓ_0-GM, which possesses a better energy compaction capacity, provides estimations too concentrated in low frequencies. The discussion about why this happens is left to a future work.

13.2. Interpolation of missing pixels

13.2.1. Introduction

Missing pixels in images is a common problem both in the capture and transmission of digital images. It is also usual wanting to remove some undesired details in an image (overprinted test, publicity, a disturbing cable in a beautiful landscape, etc.) or to restore images degraded by the pass of years.

In the last 30 years, or even more, many different techniques have been proposed to recover missing pixels (they are usually referred as *in-painting* techniques). On the other hand, texture-synthesis methods can also be used to fill-in missing regions. There are many papers using this latter type of strategies, and we can refer to [136, 137, 138, 36] among others. Unfortunately, the need to manually indicate the areas of the image from where the information needed for the interpolation should be taken, makes them inappropriate in practice. The most successful heuristic strategies combine the edge propagation (using partial differential equations, PDE) with local texture synthesis (e.g., [139, 140, 141, 142]).

There is a fast and very simple method providing comparable results to PDE-based methods [143]. It is based on iteratively combining a filtering linear operation and the non-linear constraint preserving the observed pixels.

Recently, some different strategies based on promoting sparseness have been developed. A good example of this is [22], proposing a formulation based on *Expectation-Maximisation* to approximate the sparsest solution consistent with the observation through optimal minimisation of the ℓ_1-norm.

In this section we compare the performance of our methods (ℓ_p-AP and ℓ_p-GM) adapted to pixel interpolation of missing regions of the image. We see, through application examples, that ℓ_0-GM provides the best results, among them, in MSE sense. We also compare to some of the referred techniques ([22] and [143]).

13.2.2. Consistency set

When the degradation is missing pixels in the image, the consistency set is composed by those images resulting in the same observation when missing the same pixels. Then, given a subset of fixed indices, I, from 1 to N, and given an observation $\mathbf{y}$ preserving pixels y_i from the original image

for all $i \in I$, we define the consistency set associated to $\mathbf{y}$, $R_I(\mathbf{y})$, as:

$$R_I(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \ x_i = y_i, \ \forall i \in I\}.$$

Given an image $\mathbf{x} \in \mathbb{R}^N$ and a $N \times N$ diagonal matrix $\mathbf{D}$, where each element d_{ii} is 1 if $i \in I$ and 0 otherwise, the orthogonal projection of a vector $\mathbf{x} \in \mathbb{R}^N$ onto $R_I(\mathbf{y})$ is $P_{R_I(\mathbf{y})}(\mathbf{x}) = \mathbf{D}\mathbf{y} + (\mathbf{I} - \mathbf{D})\mathbf{x}$, where $\mathbf{I}$ is the $N \times N$ identity matrix.

13.2.3. ℓ_0-AP: new strategy for searching the radius

We have experienced that ℓ_p-AP does not provide good interpolation results if we search for the sparsest image within the consistency set. In practice, the quality of the interpolation depends on finding a precise value of R allowing to minimise the error of our estimation. We call R_{opt} to this value. Then, what we are looking for is, given a sparseness level R_{opt}, the projection onto the consistency set of the image with coefficients belonging to the ℓ_p-ball of radius R_{opt}, and whose pixels in I are closest, in a MSE sense, to the observed pixels in $\mathbf{y}$. To find this optimal value we propose a solution based on maximising the Mean Square Value (MSV) of the interpolated pixels [18]. Intuitively, we see that, for small values of R, only the more salient features of the observed image will be represented, so the estimation will be too smooth and we will necessarily obtain low MSV in the interpolated areas. When we choose a very high R, the broken edges caused by missing pixels will be better represented by using a lot of vectors that approximating them at lower scales. This provokes a poor interpolation and, once again, a low MSV. Finally, if we use intermediate values for R, we expect to have enough functions to represent the main features of the image, but not enough to describe false edges. Because of that, the missing holes will be filled with the appropriate dictionary functions, what will cause a higher MSV in the interpolated areas, and this is a better interpolation.

Bold line in Figure 13.6, corresponding to the left vertical axis, shows the normalised MSE in the estimated pixels for each value of R, where R is normalised by R_{opt}, which corresponds to the minimum (at one) of this curve. Dashed line shows the normalised MSV, corresponding to the right vertical axis. We call R_{max} to the value of R where this curve reaches its maximum. Dotted lines indicate the typical deviation of this curve for each value of R. Dashed-dotted line is the real MSV value of the lost pixels in the original image, which is an upper bound for the MSV of the estimation. All these values are averaged in our test set, using a randomly generated mask where approximately 40 % of pixels are lost. For each test 250 ℓ_0-AP iterations were executed. The described method proposes to estimate R_{opt}

from the observed value R_{max}. Then, for this percentage of missing pixels, we have that $\hat{R}_{opt} = \frac{1}{0,7} R_{max}$.

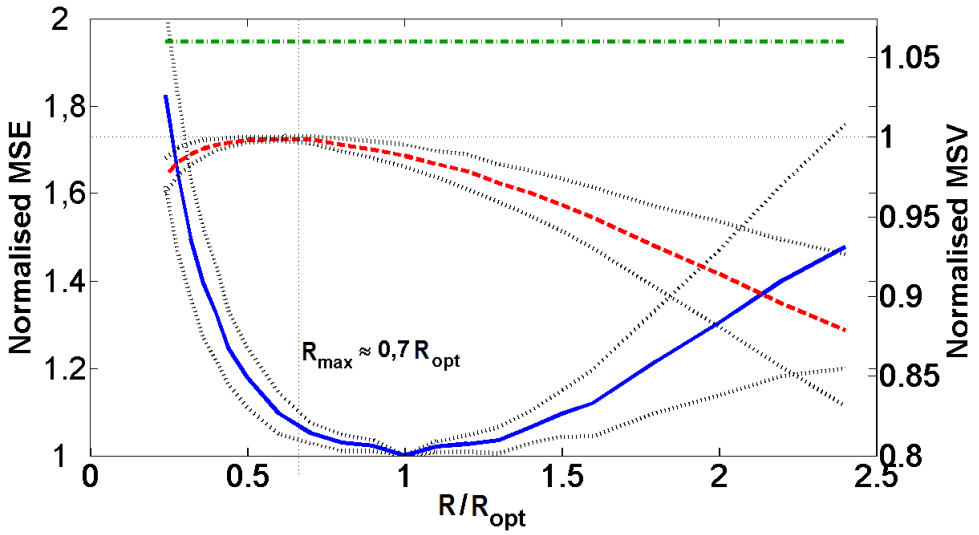

Figura 13.6: **Bold line - left axis**, *Mean Square Error of the estimated pixels normalised to the minimum value of this curve, in ordinates, and to the value for which this minimum occurs, in abscissas.* **Dashed line - right axis**, *Normalised Mean Square Value of the estimated pixels.* **Dotted line**, *typical deviation for each value in the horizontal axis.* **Dashed - dotted line**, *Mean Square Value of the original pixels in the missing positions. All curves are averaged in our test set, using a random mask where approximately* 40 % *of the pixels are lost.*

13.2.4. Implementation

Fast-inpainting. This method [143] simply consists of iteratively applying a linear filtering using the $2D$ convolving mask:

$$\begin{pmatrix} 0{,}073235 & 0{,}176765 & 0{,}073235 \\ 0{,}176765 & 0 & 0{,}176765 \\ 0{,}073235 & 0{,}176765 & 0{,}073235 \end{pmatrix} \tag{13.2}$$

and a projection onto the consistency set to preserve the observed pixels. In our implementation, the iterations end when the mean square difference of the estimation at some iteration with respect to the one at previous iteration is less than 10^{-3}.

EM-inpainting. This method [22] follows a convex relaxation approach. We have used the implementation available in MCALab 8.02 for MATLAB®, which can be downloaded from [144]. We have used the values of the parameters described in [22]. The representation used is 6-scale

Curvelets combined with Local DCT (LDCT) using a block size of 32×32. In [144] one can also find the analysis and synthesis functions for both representations.

ℓ_p-AP and ℓ_p-GM. To apply ℓ_p-AP to in-painting. We first estimate the optimal radius, for which several interpolations are required, and then we execute the method once again. We have used 100 iterations for each interpolation. Regarding ℓ_p-GM we have experienced that the performance is not significantly improved when β is greater than 0,8. In addition, in this case there are no significant differences when using α values greater than α_0. We stop the iterations when the threshold is below 0,1.

As ℓ_1-GM and *EM-inpainting* follow a very similar strategy (promoting sparseness through minimizing the ℓ_1-norm), we only present the results of that implementation giving us the best performance for each case.

13.2.5. Results and discussion

13.2.5.1. Missing random pixels (*filling-in*)

Here we consider that the pixels are independently lost with a given probability. Restoration of this type of degradations is usually known as *filling-in*.

We systematically compare ℓ_0-GM (using SA) to ℓ_1-GM (using also SA) and *Fast-inpainting*. Tables 13.2 and 13.3 show the averaged performance of each method with the images in our test set and for a wide range of the percentage of missing pixels. For ℓ_0-GM, we use two different representations: 6-scale Curvelets alone and combined with LDCT using 32×32 blocks. For ℓ_1-GM, we only use the combined representation (providing the best results). For both ℓ_0-GM and ℓ_1-GM, we have used a scale factor of $\sqrt{0{,}5}$ for both dictionaries. Note that by minimising the ℓ_0-norm we obtain the best results, except for very high percentages of missing pixels. We see again that using combined incoherent dictionaries improves the quality of the estimation.

Figure 13.7 shows the images corresponding to the compared methods for randomly missing $\approx 80\,\%$ of image pixels for *Barbara*. We observe that the method based on minimising the ℓ_1-norm (ℓ_1-GM) is not able to interpolate properly the missing pixels. On the other hand, the linear interpolation is totally unable to recover the texture. Surprisingly, ℓ_0-GM achieves very good results, given the high percentage of missing pixels, both in smooth and texture areas. We can also see that using combined representations removes the artifacts inherent to the representation (isolated selected atoms).

Time per iteration for ℓ_1-GM and ℓ_0-GM are, obviously, very similar ($\approx$ 3 min.). *Fast-inpainting* takes only around 0,5 seconds per image, so it

Method	PSNR (dB)				
$\approx$ % missing pixels	10	20	30	40	50
Observation	24,05	20,90	19,13	17,90	16,82
Fast-inpainting	37,72	34,26	32,23	30,67	29,08
ℓ_1-GM	42,21	38,14	35,48	33,29	30,84
ℓ_0-GM (Curv.)	*43,00*	*39,15*	*36,67*	*34,72*	*32,56*
ℓ_0-GM (Curv.+LDCT)	**43,48**	**39,65**	**37,09**	**35,14**	**32,96**

Cuadro 13.2: *PSNR (through MSE average) when restoring the image in our test set after randomly missing different percentages of pixels. The PSNR of the observation has been calculated by using the global mean for lost positions. Both ℓ_0-GM and ℓ_1-GM use 6-scale Curvelets and LDCT with block-size 32×32, and the same scale factor for both of them ($\sqrt{0{,}5}$). ℓ_0-GM is also presented using only Curvelets. Bold numbers indicate the best method for each percentage, and italic the second best.*

Method	PSNR (dB)			
$\approx$ % missing pixels	60	70	80	90
Observation	16,06	15,42	14,78	14,30
Fast-inpainting	27,78	26,48	*24,85*	**22,88**
ℓ_1-GM	28,75	26,38	22,84	18,54
ℓ_0-GM (Curv.)	*30,70*	*28,45*	24,79	*19,86*
ℓ_0-GM (Curv.+LDCT)	**31,03**	**28,77**	**25,14**	19,68

Cuadro 13.3: *Continuation of Table 13.2 for greater missing pixels percentages.*

is an alternative when there are not enough resources to apply ℓ_0-GM.

13.2.5.2. Missing pixel areas (in-painting)

Recovery of missing areas in an image is more difficult than interpolating randomly missing pixels. Here we compare the application of ℓ_0-GM to this problem against the methods described previously. Both the test images and the pixels masks used can be found in [144]. We have also downloaded from this page the results of *EM-inpainting*, enforcing the value of observed pixels in order to compare in the same conditions as the other methods[3].

Figure 13.8 shows a particularly interesting example because missing pixels are localised both in smooth and textured areas. Top-left panel is the observation, with missing pixels filled in using the mean of the

[3]Note that enforcing the values of the observed pixels necessarily increases the PSNR of the estimation, whereas it may make some artifacts more visible.

Figura 13.7: *Visual interpolation example of randomly missing pixels.* **Top-left**, Barbara *image, cropped to* 128×128. **Top-right**, *missing* $\approx 80\,\%$ *of the pixels and filling them with the global mean (PSNR:* 14,75 *dB).* **Centre-left**, *interpolation made by* ℓ_1*-GM (*23,26 *dB).* **Centre-right**, *result from* Fast-inpainting *(*24,84 *dB).* **Bottom-left**, ℓ_0*-GM result using Curvelets with* 6 *scales (*25,19 *dB)* **Bottom-right**, *interpolation made by* ℓ_0*-GM combining* 6*-scale Curvelets and LDCT with block size* 32×32*, and equal scale factors,* $\sqrt{0{,}5}$ *(*25,65 *dB).*

observed pixels. Top-right panel corresponds to *Fast-inpainting* (32,71 dB), and bottom-left to *EM-inpainting* (34,14 dB) using Curvelets and LDCT with 32×32 block size. Last panel is ℓ_0-GM result using Curvelets with 6 scales (34,92 dB). Note that, in contrast with the randomly missing pixels case, now the result based on minimising the ℓ_1-norm is better than the one based on iterative linear filtering. Once again, ℓ_0-GM provides the

Figura 13.8: **Top-left**, Barbara *image where value of missing pixels is the global mean of the observed ones (PSNR:* 24,19 *dB).* **Top-right**, Fast-inpainting *result (*32,71 *dB).* **Bottom - left**, EM-inpainting *result using* 6*-scale Curvelets and LDCT with block size* 32×32 *(*34,14 *dB).* **Bottom-right**, ℓ_0*-GM result using* 6*-scale Curvelets (*34,92 *dB).*

best performance. Figure 13.9 zooms in the second quadrant of the *EM-inpainting* (left) and ℓ_0-GM (right) results. Partial recovery of the lost eye is particularly interesting in ℓ_0-GM, but we also note a much better interpolation of the nose and the mouth.

Figure 13.10 is a practical example of restoration of old degraded photos. The image obtained in [144] was deformed, so we have stretched it with a 1,4 scale factor in the vertical direction. We have also removed last row as a requisite of the analysis and synthesis functions of LDCT, where the number of rows and columns should be multiple of half the block size used). The

Figura 13.9: **Left**, *detail of the result of* EM-inpainting *shown in Figure 13.8 (PSNR:* 34,38 *dB).* **Right**, *same for the method* ℓ_0*-GM (*35,13 *dB).*

compared methods[4] appear in the same order as in Figure 13.8, but ℓ_0-GM is now using 6-scale Curvelets combined with LDCT. Once again, we have enforced the observed values in the *EM-inpainting* result to have similar conditions for the comparison. We see again that ℓ_0-GM is qualitatively better than the other two methods. In contrast with the other methods, the lower thick horizontal line is barely visible in ℓ_0-GM. Also, faces of the girls, particularly the oldest and the youngest, are much better interpolated with ℓ_0-GM.

13.2.6. Conclusions

In this section we have applied the methods derived in this Thesis to the interpolation of missing pixels in the image. We have introduced a new heuristic to find the best radium for ℓ_p-AP, based on maximising the MSV of the estimated pixels. This solution, in general, is less sparse than that obtained with the classical strategy (looking for the sparsest image inside the consistency set).

Our experiments show, however, that using ℓ_0-GM with AS provides much better results. This is consistent with the model of promoting sparse solutions. We have compared to a very efficient method (*Fast-inpainting*) [143] based on combining linear and non-linear operations in the image domain. We have also compared to methods based on promoting

[4]The *EM-inpainting* result available in [144] has a different size to that of the observation and pixels mask, so we have replicated 3 times the first column.

Figura 13.10: **Top-left**, *damaged photographic image.* **Top-right**, Fast-inpainting *result.* **Bottom-left**, EM-inpainting *result using Curvelets and LDCT.* **Bottom-right**, *our result using ℓ_0-GM with* 6*-scale Curvelets and LDCT using* 32×32 *blocks and both scale factors to* $\sqrt{0{,}5}$.

sparseness through the minimisation of the ℓ_1-norm (*EM-inpainting* [22] and ℓ_1-GM). In all the experiments, ℓ_0-GM has clearly outperformed its competitors.

To conclude, we have observed, again, that methods based on finding suboptimal solutions to the minimisation of the ℓ_0-norm behave substantially better than those minimising optimally the ℓ_1-norm, for usual images and representations. Our results also outperform other heuristics. We should mention the good behaviour of ℓ_0-GM with high randomly missing pixels percentages and with complicated richly textured areas. Using Curvelets clearly contributes to the success of edge/lines interpolation, because its atoms are very appropriate for elongated features. Moreover, we have seen that, usually, combination of Curvelets with LDCT further improves the results.

13.3. Spatial-chromatic interpolation in digital camera mosaics

13.3.1. Introduction

Most conventional digital cameras are based on the Colour Filter Array technology introduced by [145]. This means that they have a sensor capturing only one colour at each pixel. To reconstruct the complete colour image it is needed, in consequence, to interpolate the non-captured colour components at each position. This process is known as *de-mosaicing*.

There have been proposed very diverse techniques to solve this problem (see [146] for a review). In order to integrate the de-mosaicing as part of the digital image capturing process, to have a fast algorithm is very important, especially when considering the rapidly increasing resolution of CCD sensors. Bilinear interpolation, for example, is very fast, but it processes independently each colour channel, thus ignoring the correlation between them. This results in poor results [147]. On the other hand, high performance iterative methods are too slow, and thus inadequate for real time image capture [148, 149, 150]. However, their good results allow having high-quality images if it is possible to post-process them after the capture. Finally, some existing methods are based on linear filtering taking into account the inter-channel correlation, then trying to reach a good balance between computation and image quality [151, 152, 147, 153].

In this section we explore the performance of ℓ_0-GM applied to de-mosaicing. Given its iterative nature, based on analysis and synthesis operations with redundant dictionaries, we cannot expect to be competitive with other methods in terms of speed. However, the high quality of ℓ_0-GM results makes an appealing alternative when we can post-process the image after capturing it. Firstly, we compare the results of ℓ_0-GM to other methods proposed in this Thesis, and then to three state-of-the-art methods.

13.3.2. Consistency set

The degradation inherent to the Bayer mosaic is, similarly to the in-painting case, in missing "pixels.ºf the three-folded image formed by the three chromatic channels of the image. Therefore, the set of images consistent with the observation is analogous to that used in previous section, that is, it is formed by all those images preserving the observed colour components. Then, given a set of indices, I, taken from the interval $\{1, ..., 3N\}$, and given an observation $\mathbf{y} \in \mathbb{R}^{3N}$, preserving all pixels y_i of the original image where $i \in I$, we define the consistency set associated to

$\mathbf{y}$, $R_d(\mathbf{y})$ as:

$$R_d(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \ x_i = y_i, \ \forall i \in I\}.$$

Given an RGB image $\mathbf{x} \in \mathbb{R}^{3N}$ and a diagonal $3N \times 3N$ matrix $\mathbf{D}$ where each element d_{ii} is 1 if $i \in I$ and 0 otherwise, the orthogonal projection of a vector $\mathbf{v}$ onto $R_d(\mathbf{y})$ is simply $P_{R_d(\mathbf{y})}(\mathbf{v}) = \mathbf{D}\mathbf{y} + (\mathbf{I} - \mathbf{D})\mathbf{v}$, where $\mathbf{I}$ is the $3N \times 3N$ identity matrix.

13.3.3. Additional constraint increasing the spatial-chromatic correlation

As we have said, it exists a strong correlation between the amplitude distributions of the pixels of the three chromatic channels of a RGB image. If we apply our methods independently to each channel the results will be unsatisfactory for not considering this correlation. Then, we introduce a modification in the sparseness promoting methods, which maintains the methodology and the convergence properties but preserves better the correlation between channels.

To start with, we change the colour space of our images from RGB to YUV. In this new space the correlation between channels is reduced, because it is composed of one luminance and two chrominance channels. As transforming from one colour space to the other and vice versa is a linear operation, this does not affect to the geometrical and convergence properties of the methods, as the composed transformation including the change of colour representation and the redundant transform it is still a tight frame, assuming the latter was already a tight frame.

Chromatic components U and V represent differences between colours. They are somehow indicating how correlated they are. We want to keep a high correlation, to avoid classical high frequency colour aliasing artifacts. Thus, the elements of such a representation should be low in amplitude, especially in the high frequencies. Then, we want to promote the smoothness of the U and V channels. This is consistent to many colour image representations, which use less information for the chrominance channels than for the luminance. An example is the PAL colour TV system.

The proposed modification is to introduce one more step at each iteration of the algorithm. This new step consists of setting to zero all high-frequency sub-bands of U and V wavelet representation. This can be interpreted as an added knowledge to the *a priori* model, so we simultaneously promote sparse images and those having high correlation between chromatic channels in the high spatial frequencies. Note that setting to zero some frequencies is not against with promoting sparseness, because the number of significant

coefficients is decreased. In addition, this constraint is an orthogonal projection onto a convex set, so the convergence properties of the method are basically unaffected.

A similar idea is described in [152], where a low-pass filter is applied to the frequencies of R and B channels. This manages to reduce the colour artifacts of the bilinear interpolation. However, it also smoothes a lot the image, while our method, as it preserves the high frequency details of the luminance, it keeps sharp edges (as shown in subsection 13.3.5).

13.3.4. Implementation

Existing methods. We have compared to methods having competitive performance with the current state-of-the-art. First method is based on promoting the channel correlation using alternating projections [148]. Second method is an iterative algorithm dealing with representation of colour differences and using an adaptive stopping criterion in this space, with the objective of keeping high correlation between colour channels and removing zipper artifacts [150]. Finally, we also compare with a heuristic method, which is effective to keep channel correlation, although it is computationally inefficient [149]. All these methods are implemented in a MATLAB® package available in the Web page of Professor Xin Li [154].

ℓ_p-GM. In all the ℓ_p-GM experiments we used $\alpha = \alpha_0$ and $\beta = 0{,}8$, which provide a good trade-off between computation and quality. We established the stopping threshold for the iterations in 0,01. We used 5 scales for both DT-CWT and Curvelets.

13.3.5. Results and discussion

13.3.5.1. ℓ_0-GM vs. ℓ_1-GM

The results obtained for these application using ℓ_p-AP are worse (and much slower) than those using bilinear interpolation. So we have not included these methods in the comparison.

In our experiments we have also found that ℓ_0-GM provides better results than ℓ_1-GM. Figure 13.11 is an example, where the methods have been applied to the Bayer mosaic with pattern 'GB' built with image 15 of *Eastman Kodak* database [155] (*Lighthouse*). Top-left panel shows a 64×64 crop of the original image with a very high-frequency pattern and, thus, particularly difficult to interpolate. The result of top-right panel corresponds to ℓ_1-GM using DT-CWT. PSNR values per channel are: 37,24 dB for R, 39,87 for G, and 37,17 for B. We can still appreciate some colour artifacts. Bottom-left panel is the result of ℓ_0-GM using Curvelets (39,29, 42,27 and

38,20 dB). Finally, bottom-right panel is ℓ_0-GM using DT-CWT (39,59, 41,99 and 39,07 dB). Note that ℓ_0-GM is clearly better than ℓ_1-GM, in terms of removing colour artifacts. Using ℓ_0-GM, PSNR is similar for both DT-CWT and Curvelets, but, on the one hand, this difference is hardly appreciable in these dB levels; and, on the other hand, colour correlation seems better using DT-CWT. In addition, the implementation we used for Curvelets is 4 times slower than the one for DT-CWT, which translates in Curvelets taking around 100 minutes per each full image in [155], whereas only 25 minutes using DT-CWT.

13.3.5.2. ℓ_0-GM vs. other methods

We show in Table 13.4 the averaged MSE for each RGB channel obtained by a set of methods and for the images 18, 31, 32, 33, 12, 34, 39, 15, 40, 16, 17 and 19 of [155] (same selection as in [150]). We also averaged the error in ∇E^*_{ab} S-CIELab metric [156] and the computation time. We use MSE instead of PSNR because is a more significant datum in these examples, provided the little difference between the methods and the high level of signal-to-noise ratio they achieve. In addition to this, in Tables 13.5 and 13.6 we have detailed these results for every image. Note that, although ℓ_0-GM is slower than the rest, its performance is comparable, being the best or second best in several cases.

However, we want to emphasize that our method is particularly good in high-frequency areas where the de-mosaicing is very complicated. In Figure 13.12 we can see the crop of the results of the methods compared in a specially difficult area of the image 15 of [155] (128×128). We can appreciate the significant reduction of colour artifacts in our method. We have also reduced the zipper artifacts with respect to [149] and [148]. These artifacts, are due to imposing the observed values, according to the mosaic structure, where the holes are not well-interpolated [150].

Method	MSE			∇E^*_{ab}	Time
	R	G	B	*S-CIELab*	(s.)
Lu & Tam [149]	8,67	5,59	11,91	**0,78**	932,22
Gunturk et al. [148]	7,92	*3,61*	10,60	0,84	*9,40*
Li [150]	**7,66**	**3,46**	**8,61**	*0,83*	**2,31**
ℓ_0-GM	*7,68*	4,31	*9,53*	0,93	1238,50

Cuadro 13.4: *MSE and S-CIELab error averaged in the* 12 *images described in the text, including computation times. Bold numbers indicate the best result for each column, and italic the second best.*

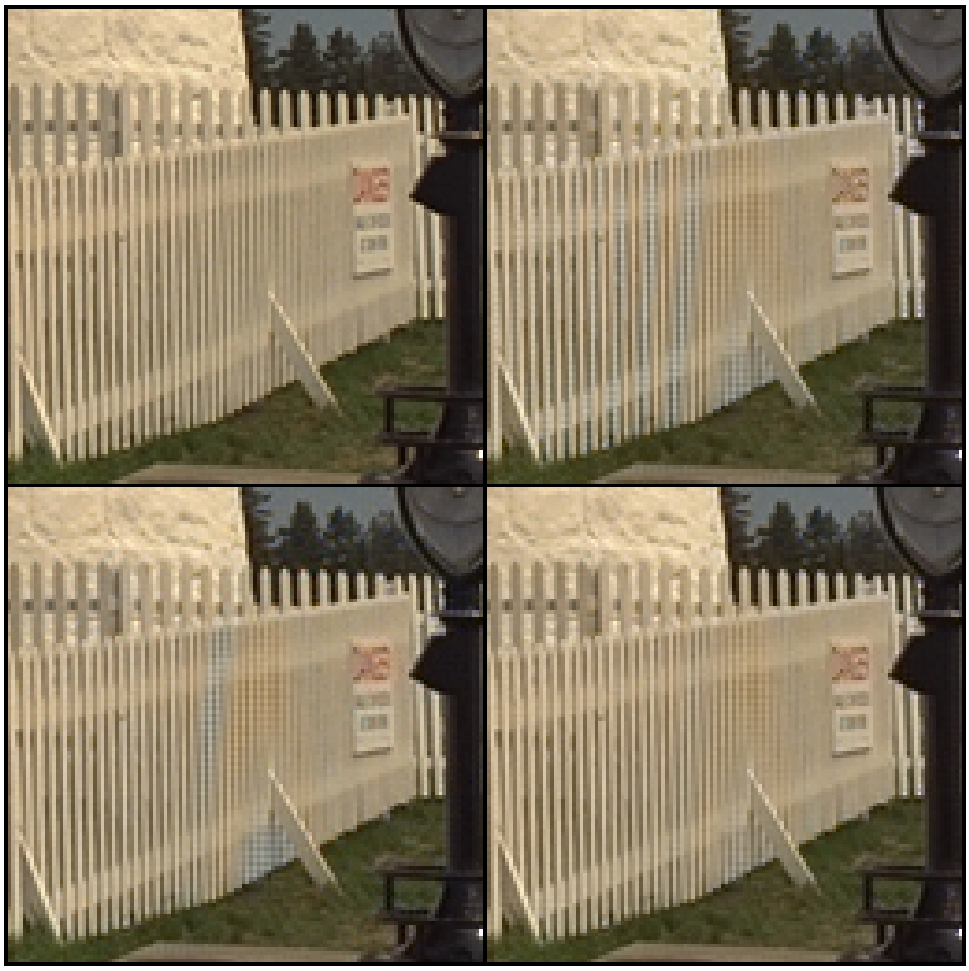

Figura 13.11: *Visual example of the comparison between ℓ_1-GM and ℓ_0-GM applied to demosaicing.* **Top-left**, 64×64 *detail of image* 15 *of* Eastman Kodak *database.* **Top-right**, *result of de-mosaicing a Bayer mosaic with patter 'GB' using ℓ_1-GM with DT-CWT. PSNR for channels R, G and B is* 37,24 *dB,* 39,87 *dB and* 37,17 *dB.* **Bottom-left**, *ℓ_0-GM result using Curvelets (*39,29*,* 42,27 *and* 38,20 *dB).* **Top-right**, *ℓ_0-GM result using DT-CWT (*39,59*,* 41,99 *and* 39,07 *dB).*

13.3.6. Conclusions

We have proposed the application of ℓ_0-GM to the problem of interpolating the lost chromatic components after capturing a natural image using a Bayer mosaic. We have applied our methods to find sparse approximations to the representation of the image in the YUV colour space. To promote chromatic regularity, we have introduced an extra projection minimising the ℓ_2-norm of the high-frequency components of the U and V channels by setting to zero the high-frequency sub-bands of a redundant

Image	Method	MSE			∇E^*_{ab}	Time
		R	G	B	*S-CIELab*	(s.)
	[149]	3,26	*1,51*	**3,16**	**0,56**	981,77
18	[148]	**2,69**	2,01	4,88	0,62	9,71
	[150]	4,25	1,99	3,49	0,63	1,78
	ℓ_0-GM	*3,05*	**1,44**	*3,24*	*0,58*	1303,50
	[149]	9,73	6,43	11,23	**0,88**	980,13
31	[148]	*7,98*	*3,46*	13,29	0,97	10,09
	[150]	8,15	**3,26**	*9,86*	**0,88**	2,07
	ℓ_0-GM	**7,21**	3,82	**8,25**	*0,93*	1368,15
	[149]	**3,06**	**1,75**	**4,12**	**0,51**	969,55
32	[148]	*3,43*	*2,21*	6,59	0,63	8,63
	[150]	3,99	2,49	5,51	0,65	2,39
	ℓ_0-GM	4,29	2,23	*4,81*	*0,57*	1316,93
	[149]	20,50	12,17	19,81	**1,29**	970,12
33	[148]	18,28	*7,42*	23,06	1,40	8,84
	[150]	*16,74*	**6,28**	**14,48**	*1,35*	3,09
	ℓ_0-GM	**15,71**	8,44	*17,88*	1,48	1384,49
	[149]	*3,30*	2,05	*4,06*	**0,53**	971,74
12	[148]	**3,29**	*1,79*	5,43	*0,60*	10,21
	[150]	3,57	**1,77**	**3,95**	0,58	2,30
	ℓ_0-GM	3,40	2,02	4,37	*0,60*	1428,25
	[149]	*8,27*	5,14	*7,27*	**0,73**	973,10
34	[148]	**6,84**	*3,78*	7,61	0,76	9,05
	[150]	8,38	**3,29**	**6,90**	*0,75*	2,06
	ℓ_0-GM	9,65	4,39	9,05	0,96	1403,20

Cuadro 13.5: *MSE, S-CIELab and computation time for the* 12 *images describe in the text and the four methods compared. Bold numbers indicate the best result for each column and method, and italic the second best.*

linear representation of these channels at each iteration. This does not change the basic geometrical interpretation and convergence properties of the algorithm.

We have seen again that promoting sparseness through a direct local minimisation of the ℓ_0-norm outperforms the optimal minimisation of ℓ_1-norm. We have also seen that ℓ_0-GM is competitive with other methods, although it is very slow. Moreover, we have seen that our method behaves particularly well, clearly better than the others, in difficult high-frequency

Image	Method	MSE			∇E^*_{ab}	Time
		R	G	B	*S-CIELab*	(s.)
	[149]	4,91	3,18	5,50	*0,83*	976,80
39	[148]	4,53	**1,67**	6,23	0,92	8,73
	[150]	*4,34*	*1,72*	*4,92*	**0,82**	2,32
	ℓ_0-GM	**3,30**	1,78	**3,94**	**0,82**	1385,81
	[149]	6,80	4,53	8,02	**0,74**	975,13
15	[148]	*6,67*	*2,67*	*7,71*	**0,74**	9,66
	[150]	**5,88**	**2,51**	**6,86**	**0,74**	2,24
	ℓ_0-GM	7,08	4,12	7,93	*0,88*	1144,42
	[149]	6,83	5,12	34,98	*0,62*	890,25
40	[148]	10,57	**2,17**	*7,71*	**0,61**	9,77
	[150]	**5,35**	*2,80*	7,80	*0,62*	2,40
	ℓ_0-GM	*5,55*	3,04	**7,44**	0,69	1488,72
	[149]	8,97	6,49	10,04	*0,85*	832,32
16	[148]	*7,43*	**3,26**	**9,34**	**0,84**	8,29
	[150]	8,08	*3,28*	*9,45*	0,86	1,99
	ℓ_0-GM	**7,22**	4,52	10,13	1,10	1495,98
	[149]	*7,72*	4,71	**8,85**	**0,85**	833,02
19	[148]	**7,06**	*4,66*	10,95	*0,92*	9,77
	[150]	8,65	**4,52**	*9,52*	0,94	2,10
	ℓ_0-GM	9,82	6,01	11,97	1,07	1436,15
	[149]	20,64	13,96	25,91	**0,99**	832,75
19	[148]	16,33	*8,19*	*24,35*	*1,05*	10,02
	[150]	**14,58**	**7,57**	**20,59**	1,09	3,04
	ℓ_0-GM	*15,80*	9,94	25,38	1,46	1452,55

Cuadro 13.6: *Continuation of Table 13.5 for the rest of the* 12 *images used.*

areas, thus reducing the artifacts significantly. Therefore, we can conclude that it is a promising alternative when the image can be post-processed after capturing them.

The results of ℓ_0-GM still have many zipper artifacts. These appear because we are forcing the observed pixels to not quite well interpolated areas.

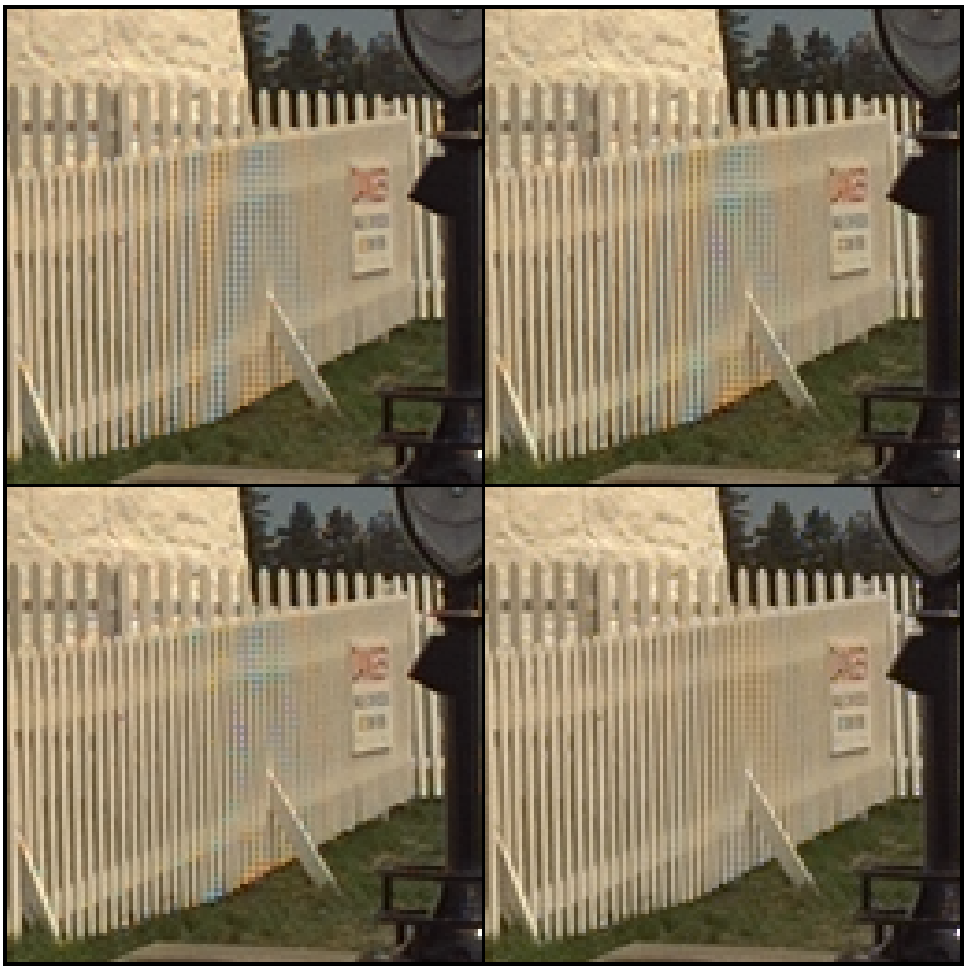

Figura 13.12: *Visual comparison between de-mosaicing methods.* **Top-left**, *result of method in [149] with the image* 15 *of* Eastman Kodak *database. We have cropped the image to* 64×64 *to improve the visibility of artifacts. PSNR of R,G and B channels is, respectively,* 39,81, 41,57 *and* 39,09 *dB.* **Top-right**, *[148] result (*39,89, 43,87 *and* 39,26 *dB).* **Bottom-left**, *[150] result (*40,39, 44,14 *and* 39,73 *dB).* **Bottom-right**, ℓ_0*-GM result (*39,59, 41,99, 39,07 *dB).*

13.4. Detail increase

13.4.1. Introduction

Images often suffer from a resolution lost. This could happen, for example, when capturing using photodetectors integrating the incident light; or also when transmitting images through a limited channel.

Detail increase or super-resolution of images consists of the process of obtaining an image or sequence of images with higher-resolution from a set

of lower-resolution observations [157]. There are many works approaching this problem when there are multiple observations (e.g., video). This is called dynamic super-resolution (e.g., [158, 159, 157]). Here we have focused on the case of having a single observation, called static or single-frame super-resolution case. It is also known as , detail increse, image scaling, interpolation, zooming-in or enlargement.

There are very simple linear methods (such as bilinear, bicubic, etc.), which treat homogeneously every pixel, interpolating their value from a linear combination of the neighbours. Linearity seriously limit the final quality. On the one hand there is an excessive blurring, diffusing the edges and avoiding the preservation of details. On the other hand, there are often aliasing artifacts.

These problems have motivated the study of more powerful super-resolution algorithms, based on non-linear techniques. All of them have in common taking advantage of the strong correlation between neighbouring pixels. Some of them use heuristics adapted to the problem (e.g., [160, 161]). Others are based on learning the relationship between observed and original images, as in [162]. However, these methods lack of a mathematical model founding their good performance. There are other methods based on setting *a priori* models of the statistics of natural images. For example, [163] establishes a model based on high-kurtosis distributions. Similarly, [164] develops a successful method based on promoting sparseness via minimisation of the ℓ_1-norm. See [165] for recent (2006) and interesting review on non-linear static super-resolution methods.

In this chapter we apply ℓ_p-GM to the static super-resolution problem. We see that both versions, $p = 0$ and $p = 1$, behave similarly both in terms of PSNR and visually, although ℓ_0-GM obtains sparser solutions. We have also compared, for reference, to the nearest neighbour and bilinear interpolation. Our objective in this section is not to propose an alternative to current static super-resolution methods but to analyse the potential of the models and methods used in this Thesis regarding this kind of applications.

13.4.2. Consistency set

We name L the number of pixels averaged to obtain every observed pixel. We note the observed image $\mathbf{y} \in R^{N/L}$. We define a family of indices sets, J_i, with $i = \{1, ..., N/L\}$, and corresponding to the non-overlapping $\sqrt{L} \times \sqrt{L}$ block formed by all $j \in \{1, ..., N\}$ whose corresponding pixels in the original image, $\mathbf{x}_0 \in \mathbb{R}^N$, have been averaged to provide the observed value y_i. We also define $\mathbf{x}_0^{J_i}$ as the block formed by pixels of $\mathbf{x}_0 \in \mathbb{R}^N$ in the positions indicated by J_i. The consistency set, $R_a(\mathbf{y})$, is then formed by

those images $\mathbf{x} \in \mathbb{R}^N$ whose associated blocks, $\mathbf{x}^{J_i}$, preserve the observed average. Then we have that:

$$R_a(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : \ < \mathbf{x}^{J_i} >= y_i, \ \forall i \in \{1, ..., N/L\}\},$$

where $< \mathbf{a} >$ denotes the average value of the coefficients of vector $\mathbf{a}$. Geometrically, it is easy to check that the projection onto the affine subspace of vectors preserving a given averaged consists of subtracting a constant vector in amplitud, which is the difference between the mean of the vector to be projected and the target mean. Thus, given $\mathbf{x} \in \mathbb{R}^N$, we have that $P_{R_a(\mathbf{y})}(\mathbf{x}) = \mathbf{z}$, where, for all $i \in \{1, ..., N/L\}$ and $j \in \{1, ..., N\}$:

$$z_j^{J_i} = x_j^{J_i} - (< \mathbf{x}^{J_i} > - y_i).$$

13.4.3. Implementation

ℓ_p-GM. We have experienced that, for application to static super-resolution, best results of ℓ_p-GM, for both $p = 0$ and $p = 1$, are obtained when using DT-CWT with only 3 scales. The parameters used are $\alpha = \alpha_0$ and $\beta = 0{,}8$. Iterations end when the threshold is below 0,01.

13.4.4. Results and discussion

Table 13.7 compares methods ℓ_0-GM, ℓ_1-GM, nearest neighbours and bilinear filtering. It reflects the averaged PSNR of the estimation in the images of our test set. To begin with, it is interesting to note the good behaviour of the simplest possible method, nearest neighbours. This is because we are imposing a local mean, what is, in average, the best possible linear strategy for this degradation. On the other hand, we see that ℓ_p-GM, in both cases $p = 0$ and $p = 1$, behaves quite well. It is curious to observe that, in contrast to the rest of studied degradations, both cases provide very similar results in PSNR terms. This does not mean that the results are strictly similar. The ℓ_0-GM result provides a significantly sparser distribution of coefficients ($\approx 2{,}15 \cdot 10^5$ for ℓ_0-GM and $\approx 2{,}45 \cdot 10^5$ for ℓ_1-GM, on average). With these results at hand, we conclude that the relative performance of ℓ_0-GM has decreased with respect to previous applications, probably because the function performance vs. achieved sparseness has a maximum (this hypothesis is consistent with the bad results provided by ℓ_0-GM in de-quantizing). In fact, if we use β values closer to 1, the sparseness is further increased but the estimation error is also increased.

Figure 13.13 shows an example, using (*House*) of visual results of the methods. Both ℓ_1-GM and ℓ_0-GM have a sharper visual aspect, and better

behaviour in the edges than the linear strategies compared. The aliasing is a significantly reduced (note, for example, the edges of the roof). Finally, we see that, in this case, there is no significant difference, despite the disparity of sparseness, between ℓ_1-GM and ℓ_0-GM.

	PSNR (dB)				
Method	*Barbara*	*Boat*	*House*	*Lena*	*Peppers*
Nearest-neighbour	27,09	26,54	30,52	27,68	25,74
Bilinear interpolation	24,16	24,25	25,84	25,41	23,76
ℓ_1-GM	**27,49**	**28,84**	*33,50*	*30,66*	*28,06*
ℓ_0-GM	*27,14*	*28,80*	**33,53**	**30,75**	**28,07**

Cuadro 13.7: *PSNR (using averaged MSE) obtained in the super-resolution of the methods to recover original size of the images of our test set, when they are averaged in non-overlapping* 2×2 *blocks. Bold numbers indicate the best result for each image, and italic the second best.*

13.4.5. Conclusions

In this section we have explored the application of ℓ_0-GM and ℓ_1-GM to static super-resolution. Both of them clearly outperform linear methods compared (nearest neighbours and bilinear interpolation). Nevertheless, we have seen that the sparser solution, provided by ℓ_0-GM, is not better than that of ℓ_1-GM. Moreover, for both methods the increase in the performance is inverted when going beyond the sparseness level shown in the experiments. These are preliminary results that need to be improved in the future, through a better understanding of the underlying sparseness promoting mechanisms for this type of applications.

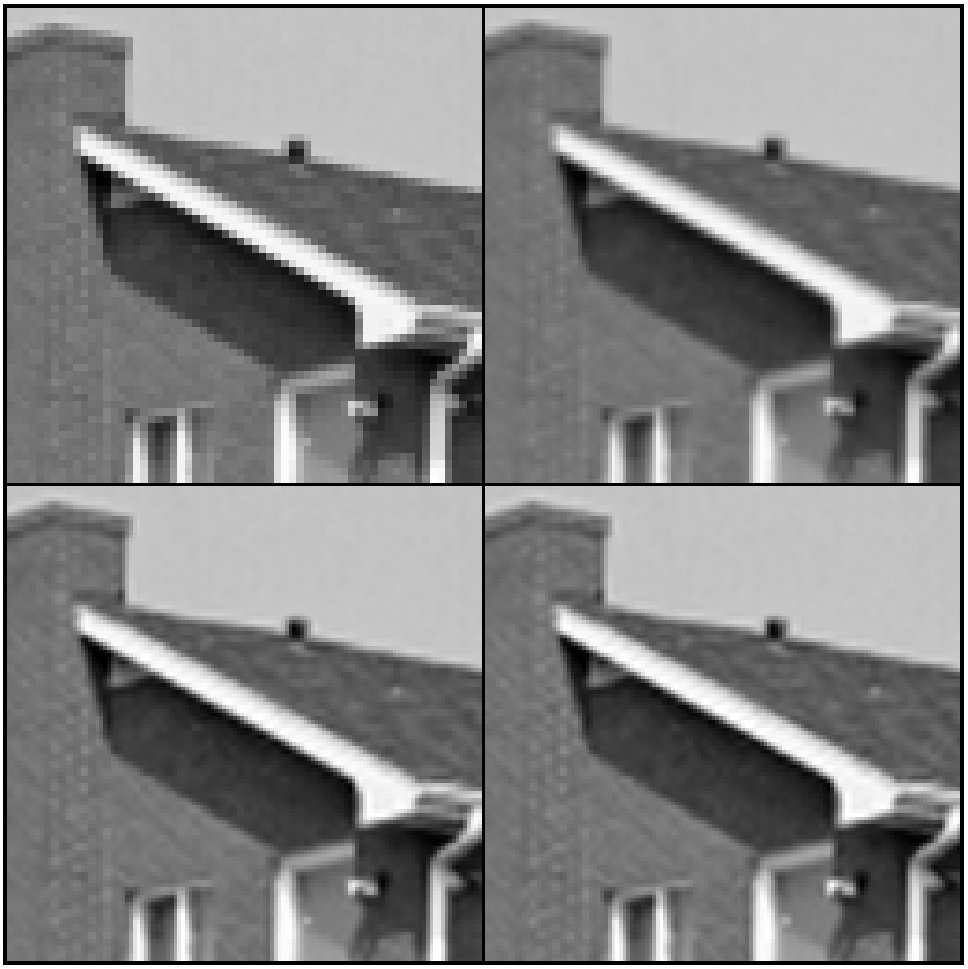

Figura 13.13: **Top-left**, *nearest neighbour interpolation (replication) of the resizing of the sub-sampling of the* 2×2 *averaged blocks* House *image (PSNR:* 30,52 *dB). We have cropped to* 128×128 *to improve visibility.* **Top-right**, *result of bilinear interpolation (*29,65 *dB).* **Bottom-left**, *result of* ℓ_1*-GM (*33,50 *dB).* **Bottom-right**, *result of* ℓ_0*-GM (*33,53 *dB).*

Capítulo 14

Conclusions and future work

14.1. Conclusions

The main conclusion that we obtain from this Thesis is that, although the global optimisation posed by the sparse approximation problem is NP-complex, it is not only possible but relatively simple to find equivalent formulations that allow, using tight frames, the application of well-known optimisation techniques to solve, at least locally, the problem. Although the proposed methods have been recently proposed and used as heuristics, until this work, up to our knowledge, nobody had established an appropriate theoretical framework to obtain them as a solution to a classical optimisation problems. This is against the common belief, quite extended through the scientific community in this field, that it is only possible to theoretically derive these kind of methods by using convex approximations to the cost function.

The main objective of this work has not been to obtain the precise theoretical conditions under which the methods find the global optimum. Instead, we have preferred to study, in an exhaustive way, their practical application in real image processing conditions, both in terms of energy compaction and application to restoration problems.

We have derived two methods to solve the sparse approximation problem. First one is formulated from the equivalent problem of minimising the MSE to the represented image from a vector of synthesis coefficients inside a ℓ_p-ball of given radius. It results in a method based on alternated orthogonal projections, using two sets: 1) the set of vectors that reconstruct perfectly the image; and 2) the ℓ_p-ball of given radium. We have called this method ℓ_p-AP. We have focused on the cases $p = 0$, where sub-optimal solutions are achieved, and $p = 1$, where we find the global minimum to the convex relaxation problem. In the experiments, we have

shown that ℓ_0-AP outperforms ℓ_1-AP under the applied conditions (natural images and useful sparseness levels in typical redundant dictionaries). It also outperforms other iterative techniques based on fixed threshold and also greedy strategies. However, we have seen that the choice of elementary functions is better in ℓ_1-AP than in ℓ_0-AP. In addition, the need of ℓ_p-AP for choosing an appropriate value for a sparseness parameter decreases its applicability.

To overcome these problems, we have derived another method. Firstly, we have reformulated our optimisation problem by finding a continuous and constrained function which is equivalent to the cost function of the sparse approximation problem. Then, we have derived a generalised version of IHT using gradient descent on this new function. We have proved that the convergence point of this method is a local minimum to the sparse approximation problem. Next, we have proposed ℓ_0-GM method, by re-writting the new function as an infinitely sharp cost function convolved with a smoothing kernel, which has allowed us to use a deterministic annealing approach to justify the use of decreasing threshold. We have also derived a similar version of the method using the ℓ_1-norm instead, ℓ_1-GM, whose use is recommended when our estimation is required to have low sparseness levels.

We have experienced that ℓ_0-GM outperforms all compared methods in providing sparse approximation solutions, including ℓ_0-AP and the optimisation of the support given by ℓ_1-AP. Then, we can conclude that methods based on minimising the ℓ_0-norm are better than those based on ℓ_1 in the conditions of this study.

It is important to note that the number of coefficients required by ℓ_0-GM to reconstruct perfectly the image tends to the theoretical asymptote as the annealing gets slower. This means a quasi-optimal asymptotic performance in the fidelity-sparseness curve. In addition, solutions with low sparseness approximate quite well the optimal solution for other values when thresholding *a posteriori*. This is very important in practice, because, in this way, we avoid searching for the optimal sparseness level in a first stage, thus simplifying the final implementation of the method and also increasing the efficiency of adjusting the sparseness level afterwards.

Regarding the application to restoration problem, we have seen that is generally easy to adapt these methods to restore images affected by strictly reproducible degradations. We have studied two different kinds of sparseness that can be used as *a priori* model for this problems. On the one hand, the synthesis-based sparseness (SS) assumes that natural images can be expressed as a linear combination of a few elementary functions of a given dictionary. This approximation, though completely valid, does not

have a solid empirical justification. We have proposed instead the use of *a priori* models based on the sparseness of the analysis coefficients (SA), which are based on the observation that the energy of the linear response of redundant wavelet-like dictionaries to natural images is concentrated in a small proportion of coefficients. This allows having a fully justified empirical basis. Adapting our methods to this kind of sparseness does not require to change their conceptual framework.

We have proposed to use ℓ_0-AP, with SS, for removing spatial quantisation artifacts and ℓ_0-GM, with SA, to several interpolation-based problems, as the recovery of missing pixels, the construction of colour images from mosaics and the increase of the detail of the images. We have seen that results based on minimisation of the ℓ_1-norm are worse in most of the studied applications. In addition to this, we have seen that our methods have a similar or superior performance with respect to other existing ones. To the best of our knowledge, nobody has explicitly applied before sparse-promoting techniques to de-quantizing.

Despite good performance, these results made us also realise of some weaknesses of the model. For example, we have seen that ℓ_0-GM provides worse results than ℓ_0-AP when removing spatial quantisation artifacts, being sparser. Moreover, ℓ_0-GM has problems to interpolate regular grids, because it tends to represent the artifacts. We have used heuristics to force the method to get out of these non-favourable local minima. However, we have experienced that this problem is not only due to the method, but also to the too a somehow simplistic sparseness model used here.

14.2. Future work

We believe that these Thesis opens several interesting venues to explore in the future. Firstly, the marginal statistics of the analysis coefficients can be better approximated by using intermediate norms ($0 < p < 1$) [80, 78]. In this sense, we believe that it is possible, though not trivial, to derive an analogous method to ℓ_0-GM and ℓ_1-GM based on them. For restoration purposes, the use of intermediate norms is more justified as *a priori* model for the analysis coefficients (see, e.g., [31]).

Secondly, we would like to study the use of *a priori* knowledge for synthesis coefficients in a more justified basis. On the one hand, using intermediate norms could be justified as a better compromise between the good behaviour of ℓ_0 and the ability of ℓ_1 to avoid local minima. On the other hand, we are working on a fully justified Expectation-Maximization-like method based on maximising the likelihood of a model for the synthesis coefficients.

With a more practical orientation, but also with important theoretical impact, we want to explore the problems of the method when dealing with regular grid interpolation. We believe that these problems are caused by the extreme simplicity of the model.

Finally, it would be interesting to study the application of the proposed methods to classical image restoration problems, as additive noise and blurring. This could be approached through a statistical formulation of the sparse approximation problem by searching the *Maximum A Posteriori* (MAP) solution to the restoration problem (as seen, for example, in [78]).

Apéndices

Apéndice A

Conjunto de imágenes de prueba

La mayoría de los resultados de esta Tesis se han realizado sobre un conjunto de imágenes estándar de prueba de tamaño 256×256 compuesto por *House*, *Boat*, *Barbara*, *Peppers* y *Lena*. *Boat* y *Barbara* han sido recortadas desde su tamaño original comenzando por la fila 200 y columna 100 en *Boat*; y por la fila 150, columna 50, en *Barbara*. En la Figura A.1 mostramos este conjunto.

Figura A.1: *Conjunto de imágenes de prueba utilizadas en esta Tesis.*

Apéndice B

Test images set

Most of the experiments in this Thesis has been performed over a standard test set of images of size 256×256, composed by *House*, *Boat*, *Barbara*, *Peppers* and *Lena*. *Boat* and *Barbara* has been cropped from their original size starting by row 200 and column 100 in *Boat*; and by row 150 and columns 50 in *Barbara*. In Figure B.1 we show this set.

Figura B.1: *Test set of images used in this Thesis.*

Apéndice C

Minimización del error cuadrático medio de la reconstrucción dado un subconjunto de coeficientes activos.

Demostramos aquí que la Ecuación (3.11) resuelve la pseudoinversa involucrada en el problema de minimización del error de reconstrucción de una imagen como combinación lineal de un conjunto dado de funciones elementales de un diccionario redundante. Dada una imagen $\mathbf{x} \in \mathbb{R}^N$, un subconjunto I de R índices extraídos del conjunto $\{1, ..., M\}$, y una matriz $\mathbf{\Phi}_I$ de tamaño $N \times R$ formada por las columnas ϕ_i de $\mathbf{\Phi}$ tales que $i \in I$, queremos resolver en $\mathbf{a}_I$:

$$\hat{\mathbf{a}}_I = \arg\min_{\mathbf{a}_I} \|\mathbf{\Phi}_I \mathbf{a}_I - \mathbf{x}\|_2, \tag{C.1}$$

que puede expresarse como:

$$\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^\sharp \mathbf{x},$$

donde $\mathbf{\Phi}_I^\sharp$ es la pseudoinversa de $\mathbf{\Phi}_I$. Estudiamos dos posibilidades: 1) $rango(\mathbf{\Phi}_I) = R \leq N$, y 2) $R > rango(\mathbf{\Phi}_I) = N$.

C.1. Primer caso: Subconjunto incompleto

Cuando $rango(\mathbf{\Phi}_I) = R \leq N$, entonces:

$$\hat{\mathbf{a}}_I = [\mathbf{\Phi}_I^T \mathbf{\Phi}_I]^{-1} \mathbf{\Phi}_I^T \mathbf{x}.$$

La inversa involucrada es potencialmente enorme. Afortunadamente, podemos usar la expansión de Taylor de la inversa de una matriz [166] para tener:

$$\hat{\mathbf{a}}_I = \sum_{k=0}^{\infty} (\mathbf{I} - \mathbf{\Phi}_I^T \mathbf{\Phi}_I)^k \mathbf{\Phi}_I^T \mathbf{x}.$$

Como condición de convergencia necesaria para la expansión de Taylor, comprobamos que, para marcos usados normalmente en representación de imágenes, los autovalores de $(\mathbf{I} - \mathbf{\Phi}_I^T \mathbf{\Phi}_I)$ no son mayores que 1 en valor absoluto. De aquí derivamos el siguiente método iterativo para calcular $\hat{\mathbf{a}}_I$:

$$\mathbf{a}_I^{(k+1)} = \mathbf{a}_I^{(k)} + \mathbf{\Phi}_I^T (\mathbf{x} - \mathbf{\Phi}_I \mathbf{a}_I^{(k)}). \tag{C.2}$$

Ahora definimos $\mathbf{S}_I$ como la matriz de tamaño $R \times N$ que selecciona los R coeficientes indicados por el conjunto I. Entonces, $\mathbf{S}_I^T$ es el operador que expande un vector de tamaño $R \times 1$ en un vector de tamaño $N \times 1$ reinsertando cada coeficiente en su posición original y poniendo a cero el resto. Entonces, $\mathbf{\Phi}_I = \mathbf{\Phi}\mathbf{S}_I^T$ y $\mathbf{\Phi}_I^T = \mathbf{S}_I \mathbf{\Phi}^T$, y sustituyendo en la Ecuación (C.2), llegamos a:

$$\mathbf{a}_I^{(k+1)} = \mathbf{a}_I^{(k)} + \mathbf{S}_I \mathbf{\Phi}^T (\mathbf{x} - \mathbf{\Phi}\mathbf{S}_I^T \mathbf{a}_I^{(k)}).$$

Multiplicando ambos lados por $\mathbf{S}_I^T$ (que es una matriz de expansión, por lo que no destruye información), tenemos:

$$\mathbf{S}_I^T \mathbf{a}_I^{(k+1)} = \mathbf{S}_I^T \mathbf{a}_I^{(k)} + \mathbf{S}_I^T \mathbf{S}_I \mathbf{\Phi}^T (\mathbf{x} - \mathbf{\Phi}\mathbf{S}_I^T \mathbf{a}_I^{(k)}).$$

Y como $\mathbf{a}_I = \mathbf{S}_I \mathbf{a}$, para algún $\mathbf{a} \in \mathbb{R}^M$ podemos escribir:

$$\mathbf{S}_I^T \mathbf{S}_I \mathbf{a}^{(k+1)} = \mathbf{S}_I^T \mathbf{S}_I \mathbf{a}^{(k)} + \mathbf{S}_I^T \mathbf{S}_I \mathbf{\Phi}^T (\mathbf{x} - \mathbf{\Phi}\mathbf{S}_I^T \mathbf{S}_I \mathbf{a}^{(k)}).$$

Sea $\mathbf{D}_I$ una matriz diagonal de tamaño $M \times M$, donde $d_{ii} = 1$ si $i \in I$ y 0 en caso contrario. Notando que $\mathbf{S}_I^T \mathbf{S}_I = \mathbf{D}_I$ y usando el hecho de que $\mathbf{D}_I$ es idempotente, vemos que:

$$\mathbf{D}_I \mathbf{a}^{(k+1)} = \mathbf{D}_I [\mathbf{D}_I \mathbf{a}^{(k)} + \mathbf{\Phi}^T (\mathbf{x} - \mathbf{\Phi}\mathbf{D}_I \mathbf{a}^{(k)})].$$

Como el término de la derecha sólo depende de $\mathbf{D}_I \mathbf{a}^{(k)}$, y, por su construcción (véase Ecuación (3.11)) $\mathbf{a}^{(k)} = \mathbf{D}_I \mathbf{a}^{(k)}$, para todo $k \geq 0$, entonces estas iteraciones son completamente equivalentes a las de la Ecuación (3.11), siendo $\mathbf{a}^{(k)}$ el resultado intermedio de esas iteraciones.

C.2. Segundo caso: Subconjunto completo

Cuando $R > rango(\mathbf{\Phi}_I) = N$, la Ecuación (C.1) tiene infinitas soluciones con reconstrucción perfecta de $\mathbf{x}$. La pseudoinversa da, entre ellas, aquella de mínima norma euclídea:

$$\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^T[\mathbf{\Phi}_I\mathbf{\Phi}_I^T]^{-1}\mathbf{x}.$$

Podemos escribir $\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^T\hat{\mathbf{z}}_I$, donde $\hat{\mathbf{z}}_I = [\mathbf{\Phi}_I\mathbf{\Phi}_I^T]^{-1}\mathbf{x}$. Entonces:

$$\hat{\mathbf{z}}_I = \sum_{k=0}^{\infty}[\mathbf{I} - \mathbf{\Phi}_I\mathbf{\Phi}_I^T]^k\mathbf{x},$$

que puede calcularse a través de las siguientes iteraciones:

$$\mathbf{z}_I^{(k+1)} = \mathbf{z}_I^{(k)} - \mathbf{\Phi}_I\mathbf{\Phi}_I^T\mathbf{z}_I^{(k)} + \mathbf{x}.$$

Multiplicando por $\mathbf{\Phi}_I^T$:

$$\mathbf{\Phi}_I^T\mathbf{z}_I^{(k+1)} = \mathbf{\Phi}_I^T\mathbf{z}_I^{(k)} - \mathbf{\Phi}_I^T\mathbf{\Phi}_I\mathbf{\Phi}_I^T\mathbf{z}_I^{(k)} + \mathbf{\Phi}_I^T\mathbf{x},$$

y sustituyendo $\mathbf{\Phi}_I^T\mathbf{z}_I^{(k)}$ por $\mathbf{a}_I^{(k)}$ obtenemos la Ecuación (C.2) y, así, se alcanza la solución usando el mismo método iterativo que en el caso anterior.

Apéndice D

Minimisation of the quadratic error of the reconstruction with a given support.

We prove next that Equation (10.11) solves the pseudo-inverse involved in the problem of minimising the reconstruction error of an image as linear combination of a given set of elementary functions from a redundant dictionary. Given an image $\mathbf{x} \in \mathbb{R}^N$, a subset I of R indices extracted from the set $\{1, ..., M\}$, and a $N \times R$ matrix Φ_I formed by columns ϕ_i from $\mathbf{\Phi}$, we want to solve for $\mathbf{a}_I$:

$$\hat{\mathbf{a}}_I = \arg \min_{\mathbf{a}_I} \|\mathbf{\Phi}_I \mathbf{a}_I - \mathbf{x}\|_2, \tag{D.1}$$

which can be expressed as:

$$\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^\sharp \mathbf{x},$$

where $\mathbf{\Phi}_I^\sharp$ is the pseudo-inverse of $\mathbf{\Phi}_I$. We study two possibilities: 1) $range(\mathbf{\Phi}_I) = R \leq N$, and 2) $R > range(\mathbf{\Phi}_I) = N$.

D.1. First case: incomplete subset

When $range(\mathbf{\Phi}_I) = R \leq N$, then:

$$\hat{\mathbf{a}}_I = [\mathbf{\Phi}_I^T \mathbf{\Phi}_I]^{-1} \mathbf{\Phi}_I^T \mathbf{x}.$$

The involved matrix inversion is potentially huge. Fortunately, we can use the Taylor expansion of the inverse of a matrix [166], and write:

$$\hat{\mathbf{a}}_I = \sum_{k=0}^{\infty} (\mathbf{I} - \mathbf{\Phi}_I^T \mathbf{\Phi}_I)^k \mathbf{\Phi}_I^T \mathbf{x}.$$

As necessary convergence condition for the Taylor expansion, we check that, for usually used frames, the eigenvalues of $(\mathbf{I} - \mathbf{\Phi}_I^T \mathbf{\Phi}_I)$ are not above 1 in absolute value. From here we derive the following iterative method to calculate $\hat{\mathbf{a}}_I$:

$$\mathbf{a}_I^{(k+1)} = \mathbf{a}_I^{(k)} + \mathbf{\Phi}_I^T(\mathbf{x} - \mathbf{\Phi}_I \mathbf{a}_I^{(k)}). \tag{D.2}$$

Now we define $\mathbf{S}_I$ as the $R \times N$ matrix selecting the R coefficients indicated by set I. Then, $\mathbf{S}_I^T$ is the operator expanding a $R \times 1$ vector into a $N \times 1$ one by placing each coefficient in its original position and setting the rest to zero. Then, $\mathbf{\Phi}_I = \mathbf{\Phi}\mathbf{S}_I^T$ and $\mathbf{\Phi}_I^T = \mathbf{S}_I\mathbf{\Phi}^T$, and substituting in Equation (D.2) we get:

$$\mathbf{a}_I^{(k+1)} = \mathbf{a}_I^{(k)} + \mathbf{S}_I\mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{S}_I^T\mathbf{a}_I^{(k)}).$$

Multiplying both sides by $\mathbf{S}_I^T$ (which is an expansion matrix, so it does not destroy any information), it yields:

$$\mathbf{S}_I^T\mathbf{a}_I^{(k+1)} = \mathbf{S}_I^T\mathbf{a}_I^{(k)} + \mathbf{S}_I^T\mathbf{S}_I\mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{S}_I^T\mathbf{a}_I^{(k)}).$$

And as $\mathbf{a}_I = \mathbf{S}_I\mathbf{a}$, for some $\mathbf{a} \in \mathbb{R}^M$ we can write:

$$\mathbf{S}_I^T\mathbf{S}_I\mathbf{a}^{(k+1)} = \mathbf{S}_I^T\mathbf{S}_I\mathbf{a}^{(k)} + \mathbf{S}_I^T\mathbf{S}_I\mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{S}_I^T\mathbf{S}_I\mathbf{a}^{(k)}).$$

Let $\mathbf{D}_I$ be a $M \times M$ diagonal matrix, where $d_{ii} = 1$ if $i \in I$ and 0 otherwise. Noting that $\mathbf{S}_I^T\mathbf{S}_I = \mathbf{D}_I$ and using the fact that $\mathbf{D}_I$ is idempotent, we see that:

$$\mathbf{D}_I\mathbf{a}^{(k+1)} = \mathbf{D}_I[\mathbf{D}_I\mathbf{a}^{(k)} + \mathbf{\Phi}^T(\mathbf{x} - \mathbf{\Phi}\mathbf{D}_I\mathbf{a}^{(k)})].$$

As the right term only depends on $\mathbf{D}_I\mathbf{a}^{(k)}$, and, by construction (following Equation (10.11)), $\mathbf{a}^{(k)} = \mathbf{D}_I\mathbf{a}^{(k)}$, for all $k \geq 0$, then these iterations are completely equivalent to those of Equation (10.11), being $\mathbf{a}^{(k)}$ the intermediate result of those iterations.

D.2. Second case: complete subset

When $R > range(\mathbf{\Phi}_I) = N$, then Equation (D.1) has infinite solutions with perfect reconstruction of $\mathbf{x}$. The pseudo-inverse provides, among them, the one with minimum Euclidean norm:

$$\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^T[\mathbf{\Phi}_I\mathbf{\Phi}_I^T]^{-1}\mathbf{x}.$$

We can write $\hat{\mathbf{a}}_I = \mathbf{\Phi}_I^T \hat{\mathbf{z}}_I$, where $\hat{\mathbf{z}}_I = [\mathbf{\Phi}_I \mathbf{\Phi}_I^T]^{-1}\mathbf{x}$. Then:

$$\hat{\mathbf{z}}_I = \sum_{k=0}^{\infty} [\mathbf{I} - \mathbf{\Phi}_I \mathbf{\Phi}_I^T]^k \mathbf{x},$$

that can be calculated using the following iterations:

$$\mathbf{z}_I^{(k+1)} = \mathbf{z}_I^{(k)} - \mathbf{\Phi}_I \mathbf{\Phi}_I^T \mathbf{z}_I^{(k)} + \mathbf{x}.$$

Multiplying by $\mathbf{\Phi}_I^T$:

$$\mathbf{\Phi}_I^T \mathbf{z}_I^{(k+1)} = \mathbf{\Phi}_I^T \mathbf{z}_I^{(k)} - \mathbf{\Phi}_I^T \mathbf{\Phi}_I \mathbf{\Phi}_I^T \mathbf{z}_I^{(k)} + \mathbf{\Phi}_I^T \mathbf{x},$$

and substituting $\mathbf{\Phi}_I^T \mathbf{z}_I^{(k)}$ by $\mathbf{a}_I^{(k)}$ we obtain the Equation (D.2), so the solution is reached using the same iterative method than in the previous case.

Apéndice E

Fusión de dos marcos de Parseval en uno sólo

Sea $\mathbf{\Phi}_A$ una matriz $N \times M$ con $M > N$ y $\mathbf{\Phi}_B$ una matriz $N \times L$ con $L > N$. Supongamos que ambas matrices son marcos de Parseval. Entonces, si formamos una nueva matriz de tamaño $N \times (M + L)$ como la unión de las columnas de $\mathbf{\Phi}_A$ y $\mathbf{\Phi}_B$, resulta que esta nueva matriz no es a su vez un marco de Parseval, porque la transformación lineal duplica la norma del vector original. Por ello, debemos modular cada una de las dos matrices primitivas con un factor de escala. Para conseguir preservar la energía en la transformación conjunta, la suma de los cuadrados de estos factores tiene que sumar 1. La diferencia entre estos dos factores mide la importancia relativa de cada una de las matrices en el nuevo marco.

Formalmente, definimos la matriz $\mathbf{\Phi}$ de tamaño $N \times (M + L)$ como aquella formada por la unión de las columnas de $\mathbf{\Phi}_A$ y $\mathbf{\Phi}_B$, multiplicadas respectivamente por dos factores de escala, $\sqrt{\gamma_A}$ y $\sqrt{\gamma_B}$, donde $\gamma_A + \gamma_B = 1$.

Apéndice F

A Parseval frame formed concatenating two Parseval frames

Let $\mathbf{\Phi}_A$ be a $N \times M$ matrix with $M > N$ and $\mathbf{\Phi}_B$ a $N \times L$ matrix with $L > N$. Assume that both matrices are Parseval frames. Then, if we form a new $N \times (M + L)$ matrix as the union of the columns of $\mathbf{\Phi}_A$ and $\mathbf{\Phi}_B$, this new matrix is no longer a Parseval frame, because the linear transformation doubles the energy of the original vector. Then, we should modulate each primitive matrix with a scale factor. In order to preserve the energy of the joint transformation, the sum of the square power of these factors should add 1. These two factors act as weight of the relative importance of each matrix in the new frame.

Formally, we define the $N \times (M + L)$ matrix $\mathbf{\Phi}$ as formed by the union of the columns of $\mathbf{\Phi}_A$ and $\mathbf{\Phi}_B$, respectively multiplied by two scale factors, $\sqrt{\gamma_A}$ and $\sqrt{\gamma_B}$, such that $\gamma_A + \gamma_B = 1$.

Apéndice G

Listado de publicaciones

Este es un listado de las publicaciones que se han derivado del trabajo realizado en esta tesis:

- L. Mancera, J.A. Guerrero-Colón, J. Portilla. Sparse Approximation via Orthogonal Projections: Beyond Greed and Convexity. *IEEE Transactions on Image Processing* (enviado)
- J. Portilla, L. Mancera. L0-based sparse approximation: Two alternative methods and some applications. *SPIE Optics & Photonics*, edited by SPIE, San Diego (CA), August 2007.
- L. Mancera, J. Portilla. L0-norm-based Sparse Representation through Alternate Projections. *13th International Conference on Image Processing (ICIP'06)*, Atlanta, GE (USA), October 2006.
- L. Mancera, J. Portilla. Image De-Quantizing via Enforcing Sparseness in Overcomplete Representations. *Lecture Notes in Computer Sciences*, vol. **3708**, pp. 411-418. También en *7th International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS 2005)*, Antwerp (Belgium), September 2005. (**Presente en 2005 JCR Science Edition**)

Apéndice H

Publications

This is a list of publications derived from this work:

- L. Mancera, J.A. Guerrero-Colón, J. Portilla. Sparse Approximation via Orthogonal Projections: Beyond Greed and Convexity. *IEEE Transactions on Image Processing* (submitted)
- J. Portilla, L. Mancera. L0-based sparse approximation: Two alternative methods and some applications. *SPIE Optics & Photonics*, edited by SPIE, San Diego (CA), August 2007.
- L. Mancera, J. Portilla. L0-norm-based Sparse Representation through Alternate Projections. *13th International Conference on Image Processing (ICIP'06)*, Atlanta, GE (USA), October 2006.
- L. Mancera, J. Portilla. Image De-Quantizing via Enforcing Sparseness in Overcomplete Representations. *Lecture Notes in Computer Sciences*, vol. **3708**, pp. 411-418. Also in *7th International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS 2005)*, Antwerp (Belgium), September 2005. (**Present in 2005 JCR Science Edition**)

Bibliografía

[1] E.P. Simoncelli and B. Olshausen, "Natural image statistics and neural representation," *Annual Review of Neuroscience*, vol. 24, pp. 1193–1216, May 2001.

[2] D.L. Ruderman, "The statistics of natural images," *Network: Computational Neural Systems*, vol. 5, no. 4, pp. 517–548, November 1994.

[3] S.G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, July 1989.

[4] H. Barlow, "Redundancy reduction revisited," *Network: Computational Neural Systems*, vol. 12, no. 3, pp. 241–253, May 2001.

[5] B.A. Olshausen and D.J. Field, "Sparse coding of sensory inputs," *Current Opinion in Neurobiology*, vol. 14, pp. 481–487, July 2004.

[6] B.A. Olshausen and D.J. Field, "Emergence of simple-cell receptive fields properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, June 1996.

[7] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, and D.J. Heeger, "Shiftable multi-scale transforms," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 587–607, March 1992.

[8] D.L. Donoho, "De-noising by soft thresholding," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[9] E.J. Candes, M. Rudelson, T. Tao, and R. Vershynin, "Error correction in linear programming," in *Proceedings 46th Sympusium on Foundations of Computer Science*, Pittsburgh, PA, 22-25 October 2005, IEEE Signal Processing Society.

[10] B.A. Olshausen and D.J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by v1?," *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[11] R.R. Coifman and D.L. Donoho, "Translation invariant de-noising," *Lecture Notes in Statistics*, vol. 103, pp. 125–150, May 1995.

[12] J. Portilla and L. Mancera, "L0-based sparse approximation: Two alternative methods and some applications," in *SPIE Optics and Photonics*, San Diego, CA, 26-30 August 2007, SPIE.

[13] S.S. Chen, "Basis Pursuit," *Ph.D. Thesis, Stanford University*, 1995.

[14] S. Mallat and Z. Zhang, "Matching Pursuit in time-frequency dictionary," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, December 1993.

[15] T.H. Reeves and N.G. Kingsbury, "Overcomplete image coding using iterative projection-based noise shaping," in *Proceedings of the 9th IEEE International Conference on Image Processing*, Rochester, NY, 23-25 September 2002, vol. 3, pp. 597–600, IEEE Signal Processing Society.

[16] M.A.T. Figueiredo and R.D. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 906–916, August 2003.

[17] N. Kingsbury and T. Reeves, "Redundant representation with complex wavelets: How to achieve sparsity," in *Proceedings of the 10th IEEE International Conference on Image Processing*, Barcelona, Spain, 14-18 September 2003, vol. 1, pp. 45–48, IEEE Signal Processing Society.

[18] L. Mancera and J. Portilla, "L0-norm-based representation through alternate projections," in *Proceedings 13th IEEE International Conference on Image Processing*, Atlanta, GE, p. 2089, IEEE Signal Processing Society.

[19] I. Daubechies, G. Teschke, and L. Vese, "Iteratively solving linear inverse problems under general convex constraints," *Inverse Problems and Imaging*, vol. 1, no. 1, pp. 29–46, 2007.

[20] J. Portilla, V. Strela, M. Wainwright, and E.P. Simoncelli, "Image denoising using a Scale Mixture of Gaussians in the wavelet domain,"

IEEE Transactions on Image Processing, vol. 12, no. 11, pp. 1338–1351, November 2003.

[21] J.L. Starck, "Morphological Component Analysis," in *Proceedings of the SPIE*, San Diego, CA, August 2005, vol. 5914.

[22] M.J. Fadili and J.L. Starck, "EM algorithm for sparse representation-based image inpainting," in *Proceedings 12th IEEE International Conference on Image Processing*, Genoa, Italy, 11-14 September 2005, IEEE Signal Processing Society.

[23] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, December 2006.

[24] Y. C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive function approximation with application to wavelet decomposition," in *Proceedings 27th Asilomar Conference in Signals, Systems and Computers*, 1-3 November 1993.

[25] D.L. Donoho, Y. Tsaig, I. Drori, and J.L. Starck, "Sparse solution of undetermined linear equations by Stagewise Orthogonal Matching Pursuit," *Technical Report*, April 2006.

[26] L. Mancera and J. Portilla, "Image dequantizing via enforcing sparseness in overcomplete representations," in *7th International Conference on Advanced Concepts in Intelligent Vision Systems*, Antwerp, Belgium, 20-23 September 2005, vol. LNCS-3708, pp. 411–418, Springer Verlag.

[27] N.N. Abdelmalek, "An efficient method for the discrete linear L1 approximation problem," *Mathematical Computation*, vol. 29, no. 131, pp. 844–8502, July 1975.

[28] I.F. Gorodnitsky and B.D. Rao, "Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, March 1997.

[29] T. Blumensath, M. Yaghoobi, and M.E. Davies, "Iterative hard thresholding and L0 regularisation," in *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, 15-20 April 2007.

[30] I. Daubechies, M. De Friese, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Maths*, vol. 57, pp. 1413–1457, 2004.

[31] M.A.T. Figueiredo and R.D. Nowak, "A Bound Optimization Approach to wavelet-based image deconvolution," in *Proceedings 12th IEEE International Conference on Image Processing*, Genoa, Italy, 11-14 September 2005, vol. 2, pp. 782–785, IEEE Signal Processing Society.

[32] D.L. Donoho, "For most large undetermined systems of linear equations the minimal L1-norm solution is also the sparsest solution," *Technical Report*, 2004.

[33] J.A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, October 2004.

[34] D.L. Donoho, M. Elad, and V.N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, January 2006.

[35] A.C. Gilbert, M.J. Strauss, J.A. Tropp, and R. Vershynin, "Algorithmic linear dimension reduction in the L1 norm for sparse vectors," *arXiv:math/0608079*, August 2006.

[36] O.G. Guleryuz, "On missing data prediction using sparse signal models: A comparison of atomic decomposition with iterated denoising," in *Proceedings of the SPIE*, San Diego, CA, August 2005, vol. 5914.

[37] D.C. Youla, "Generalized image restoration by the method of alternating orthogonal projections," *IEEE Transactions on Circuits and Systems*, vol. CAS-25, no. 9, pp. 694–702, September 1978.

[38] P. Combettes, "The foundation of set theoretic estimation," *Proceedings of the IEEE*, vol. 81, no. 2, pp. 182–208, February 1993.

[39] P. Combettes and V. Wajs, "Signal recovery by proximal forward-backward splitting," *SIAM Journal on Multiscale Modeling and Simulation*, vol. 4, pp. 1168–1200, 2005.

[40] I. Daubechies, M. Fornasier, and I. Loris, "Accelerated projected gradient method for linear inverse problems with sparsity constraints," *Submitted to arXiv:0706.4297v1*, June 2007.

[41] K.K. Herrity, A.C. Gilbert, and J.A. Tropp, "Sparse approximation via iterative thresholding," in *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, 14-19 May 2006.

[42] R.T. Rockafellar, "Monotone operators and the proximal point algorithm," *SIAM Journal of Control and Optimization*, vol. 14, pp. 877–898, 1976.

[43] R.R. Coifman and M.V. Wickerhauser, "Entropy-based algorithms for best-basis selection," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 713–718, March 1992.

[44] I. Daubechies, *Ten Lectures on Wavelets*, Cambridge University Press, 1992.

[45] N. Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *Journal of Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 234–253, May 2001.

[46] S.S. Chen, D.L. Donoho, and M.A. Saunders, "Atomic decomposition by Basis Pursuit," *SIAM Journal on Signal Processing*, vol. 20, no. 1, pp. 33–61, 1999.

[47] A. Cohen, I. Daubechies, O.G. Guleryuz, and M.T. Orchard, "Of the importance of combining wavelet-based non-linear approximation with coding strategies," *IEEE Transactions on Information Theory*, vol. 48, no. 7, pp. 1895–1921, July 2001.

[48] D.L. Donoho J.L. Starck, M. Elad, "Redundant multiscale transforms and their application for Morphological Component Analysis," *Journal of Advances in Imaging and Electron Physics*, vol. 123, pp. 287–384, 2004.

[49] D.L. Donoho and M. Elad, "On the stability of Basis Pursuit in the presence of noise," *Signal Processing*, vol. 86, pp. 511–532, 2006.

[50] I. Daubechies, "Time-frequency localization operator: a geometric phase space approach," *IEEE Transactions on Information Theory*, vol. 34, no. 4, pp. 605–612, July 1988.

[51] G. Davis, S.G. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constructive Approximations*, vol. 13, pp. 57–98, 1997.

[52] S. Weisberg, *Applied Linear Regression*, Wiley, New York, 1980.

[53] C. Daniel and F.S. Wood, *Fitting Equations to Data: Computer Analysis of Multifactor Data*, Wiley, New York, 1980.

[54] T. Hastie, R. Tibshirani, and J.H. Friedman, *Elements of Statistical Learning*, Springer-Verlag, New York, 2001.

[55] R.A. DeVore and V.N. Temlyakov, "Some remarks on greedy algorithms," *Advanced Computational Mathematics*, vol. 12, pp. 213–227, 1996.

[56] V.N. Temlyakov, "Greedy algorithms and m-term approximation," *Journal of Approximation Theory*, vol. 98, pp. 117–145, 1999.

[57] V.N. Temlyakov, "Weak greedy algorithms," *Advances in Computational Mathematics*, vol. 5, pp. 173–187, 2000.

[58] R. Gribonval and P. Vandergheynst, "On the exponential convergence of Matching Pursuits in quasi-incoherent dictionaries," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 255–261, January 2006.

[59] V.N. Temlyakov, "Nonlinear methods of approximation," *Foundations of Computational Mathematics*, vol. 3, pp. 33–107, 2003.

[60] M. Andrle, L. Rebollo-Neyra, and E. Sagianos, "Backward-Optimized Orthogonal Matching Pursuit approach," *IEEE Signal Processing Letters*, vol. 11, no. 9, pp. 705–708, September 2004.

[61] G.Z. Zarabulut, L. Moura, D. Panario, and A. Yongacoglu, "Flexible Tree-Search based Orthogonal Matching Pursuit algorithm," in *30th IEEE International Conference on Acoustic, Speech, and Signal Processing*, Philadelphia, PA, 18-23 March 2005, pp. 673–676, IEEE Signal Processing Society.

[62] C. La and M.N. Do, "Tree-Based Orthogonal Matching Pursuit algorithm for signal reconstruction," in *Proceedings 13th IEEE International Conference on Image Processing*, Atlanta, GE, 8-11 October 2006, IEEE Signal Processing Society.

[63] P. Jost, P. Vandergheynst, and P. Frossard, "Tree-based pursuit: Algorithm and properties," *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4685.

[64] S.F. Cotter and B.D. Rao, "Application of tree-based searches to Matching Pursuit," in *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, Salt Lake City, UT, 7-11 May 2001, vol. 6, pp. 3933–3936.

[65] B. Efron, T. Hastie, I. Jonhstone, and R. Tibshirani, "Least Angle Regression," *Annual Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[66] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 53–69, January 2007.

[67] C. Distante, A. Leone, L. My, M. Rizzello, and P. Siciliano, "A video compression algorithm based on Matching Pursuit integrated into a wireless embedded sensor node compliant with ieee 1451.1 standard architecture," in *Proceedings of the 2nd IASTED International Conference*, Cambridge, MA, 8-10 November 2004.

[68] A. Rahmoune, P. Vandergheynst, and P. Frossard, "MP3D: Highly scalable video coding scheme based on Matching Pursuit," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada, 17-21 May 2004, vol. 3, pp. 133–136.

[69] J.L. Lin, W.L. Hwang, and S.C. Pei, "Video compression based on Orthonormal Matching Pursuits," in *Proceedings International Symposyum on Circuits and Systems*, pp. 4–8.

[70] R. Figueras i Ventura, P. Vandergheynst, P. Frossard, and A. Cavallaro, "Color image scalable coding with Matching Pursuit," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada, 17-21 May 2004, vol. 3, pp. 53–56.

[71] R. Gribonval, "Sparse decomposition of stereo signals with Matching Pursuit and application to blind separation of more than two sources from a stereo mixture," in *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, Orlando, FL, 13-17 May 2002.

[72] D.L. Donoho and I.M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 851, no. 3, pp. 425–455, 1994.

[73] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society B.*, vol. 58, no. 1, pp. 267–288, 1996.

[74] J.A. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1030–1051, March 2006.

[75] I.W. Selesnick, R.V. Slyke, and O.G. Guleryuz, "Pixel recovery via L1 minimization in the wavelet domain," in *Proceedings of the 11th IEEE International Conference on Image Processing*, Singapore, 24-27 October 2004, IEEE Signal Processing Society.

[76] M.F. Duarte, M.B. Wakin, and R.G. Baraniuk, "Fast reconstruction of piecewise smooth signals from random projections," in *Online Proceedings of Workshop on Signal Proc. with Adaptive Sparse Structured Representations (SPARS)*, 2005.

[77] D.L. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[78] M.A.T. Figueiredo, R.D. Nowak, and S.J. Wright, "Gradient projection for sparse reconstruction: Application to Compressed Sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing*, vol. 1, no. 4, 2007.

[79] R.D. Nowak and M.A.T. Figueiredo, "Overcomplete image coding using iterative projection-based noise shaping," in *Proceedings of the 35th Asilomar Conference on Signals, Systems and Computers*, Monterrey, CA, 2001.

[80] E.P. Simoncelli and E.H. Adelson, "Noise removal via Bayesian wavelet coring," in *Proceedings of the 3rd IEEE International Conference on Image Processing*, Lausanne, Switzerland, 16-19 September 1996, vol. 1, pp. 379–382, IEEE Signal Processing Society.

[81] A. Perez and R.C. Gonzalez, "An iterative thresholding algorithm for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 6, pp. 742–751, November 1987.

[82] M. Elad, "Why simple shrinkage is still relevant for redundant representations?," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5559–5469, December 2006.

[83] A. Chambolle, R.A. DeVore, N.Y. Lee, and B.J. Lucier, "Nonlinear wavelet image processing: Variational problems, compression and noise removal through wavelet shrinkage," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 319–335, March 1998.

[84] P. Moulin and J. Liu, "Analysis of multiresolution image de-noising schemes using generalized-Gaussian and complexity priors," *IEEE Transactions on Information Theory, Special Issue on Multiscale Analysis*, vol. 45, no. 3, pp. 909–919, April 1999.

[85] T. Adeyemi and M.E. Davies, "Sparse representation of images using overcomplete complex wavelets," in *IEEE SP 13th Workshop on Statistical Signal Processing*, Bourdeaux, France, pp. 805–809.

[86] M. Elad, B. Matalon, J. Shtok, and M. Zibulevsky, "A wide-angle view at iterated shrinkage algorithm," in *SPIE Optics and Photonics*, San Diego, CA, 26-30 August 2007, SPIE.

[87] M. Elad, "Shrinkage for redundant representations," in *Workshop of Signal Processing with Adaptive Sparse Structured Representations (Spars05)*, 2005.

[88] L.A. Karlovitz, "Construction of nearest points in the Lp, p even and L1 norms," *Journal of Approximation Theorey*, vol. 3, pp. 123–127, 1970.

[89] R.M. Figueras i Ventura and E.P. Simoncelli, "Statistically driven sparse image approximation," in *Proceedings 14th IEEE International Conference on Image Processing*, San Antonio, TX, 16-19 September 2007, IEEE Signal Processing Society.

[90] H.Y. Gao and A.G. Bruce, "Waveshrink with firm shrinkage," *Statistica Sinica*, vol. 7, pp. 855–874, 1997.

[91] M. Fornasier and H. Rauhut, "Iterative thresholding algorithms," *Submitted*, April 2007.

[92] M. Fornasier, "Accelerated iterative thresholding algorithms," *Technical Report*, 2007.

[93] O.G. Guleryuz, "Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising: Part I - theory," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 539–554, March 2006.

[94] O.G. Guleryuz, "Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising: Part II - applications," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 555–571, March 2006.

[95] M.J. Fadili and J.L. Starck, "Sparse representation-based image deconvolution by iterative thresholding," in *Astronomical Data Analysis (ADA'06)*, Marseille, France, September 2006.

[96] J. Bobin, J.L. Starck, J. Fadili, Y. Moudden, and D.L. Donoho, "Morphological Component Analysis: an adaptive thresholding strategy," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2675–2681, November 2007.

[97] S. Fischer and G. Cristobal, "Minimum entropy transform using gabor wavelets for image compression," in *Proceedings of the 11th International Conference on Image Analysis and Processing (ICIAP'01)*, Palermo, Italy, 26-28 September 2001.

[98] G.H. Mohimani, M. Babaie-Zadeh, and Christian Jutten, "Fast sparse representation based on smoothed L0 norm," in *7th International ICA Conference (ICA 2007)*, London, UK, 9-12 September 2007.

[99] M.A.T. Figueiredo, J.B. Dias, J.P. Oliveira, and R.D. Nowak, "On Total Variation de-noising: A new Majorization-Minimization algorithm and an experimental comparison with wavelet de-noising," in *Proceedings 13th IEEE International Conference on Image Processing*, Atlanta, GE, 8-11 October 2006, IEEE Signal Processing Society.

[100] M.A.T. Figueiredo, J.M. Bioucas-Dias, and R.D. Nowak, "Majorization-Minimization algorithms for wavelet based image restoration," *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2980–2991, 2007.

[101] I. Drori, "Fast L1 minimization by iterative thresholding for multidimensional nmr spectroscopy," *Submitted to EURASIP Journal on Advances in Signal Processing*, August 2007.

[102] C. Rozell, D. Johnson, R. Baraniuk, and B. Olshausen, "Locally competitive algorithms for sparse approximation," in *Proceedings 14th IEEE International Conference on Image Processing*, San Antonio, TX, 16-19 September 2007, IEEE Signal Processing Society.

[103] J. Haupt and R.D. Nowak, "Compressive sampling vs. conventional imaging," in *Proceedings 13th IEEE International Conference on Image Processing*, Atlanta, GE, 8-11 October 2006, pp. 1269–1272, IEEE Signal Processing Society.

[104] D.L. Donoho and P.B. Stark, "Uncertainty principles and signal recovery," *SIAM Journal of Applied Mathematics*, vol. 49, no. 3, pp. 906–931, 1989.

[105] D.L. Donoho and X. Huo, "Uncertainity principles and ideal atomic decomposition," *Technical Report*, June 1999.

[106] E.J. Candes and J. Romberg, "Practical signal recovery from random projections," in *Proceedings of the SPIE XI Conference on Wavelet Applications in Signal and Image Processing*, 2004, vol. 5914.

[107] E. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, December 2005.

[108] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, February 2006.

[109] E. Candes, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1208–1223, August 2006.

[110] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, December 2006.

[111] M. Elad and A.M. Bruckstein, "A generalized uncertainity principle and sparse representation in pair of bases," *IEEE Transactions on Information Theory*, vol. 48, no. 9, pp. 2558–2567, September 2002.

[112] R. Gribonval and M.Ñielsen, "Sparse representations in unions of bases," *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3320–3325, December 2003.

[113] D.L. Donoho and M. Elad, "Optimally sparse representation from overcomplete dictionaries vial L1-norm minimization," *Proceedings of the Natural Academy of Sciences, USA*, vol. 100, no. 5, pp. 2197–3002, 2003.

[114] J.J. Fuchs, "On sparse representation on arbitrary redundant bases," *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1341–1344, June 2004.

[115] S. Kunis and H. Rauhut, "Random sampling of sparse trigonometric polynomials II - Orthogonal Matching Pursuit versus Basis Pursuit," *Submitted to arXiv:math/0604429v2*, February 2007.

[116] A. Bruckstein, D.L. Donoho, and M. Elad, "From sparse solutions of system of equations to sparse modeling of signal and images," *To appear in SIAM Review*, 2007.

[117] J.A. Tropp, I.S. Dhillon, R.W. Heath, and T. Strohmer, "Designing structured tight frames via an alternating projection method," *IEEE Transactions on Information Theory*, vol. 51, no. 1, pp. 188–209, January 2005.

[118] E.J. Candes and D.L. Donoho, "Curvelets - a surprisingly effective nonadaptive representation for objects with edges," in *Curves and Surfaces IV*, L.L. Schumaker et al., Ed. Vanderbilt University Press, Nashville, TN, 1999.

[119] J.A. Guerrero-Colón, L. Mancera, and J. Portilla, "Image restoration using space-variant Gaussian Scale Mixtures in overcomplete pyramids," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 27.

[120] N. Kingsbury, "Web page," *http://www-sigproc.eng.cam.ac.uk/ñgk/.*

[121] E. Candes, L. Demanet, D.L. Donoho, and L. Ying, "Curvelab," *http://www.curvelab.org/.*

[122] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–681, May 1983.

[123] A. Blake and A. Zisserman, "Graduated non-convexity," in *Visual Reconstruction*, MA MIT Press, Cambridge, Ed. 1987.

[124] F. Rooms, W. Philips, and J. Portilla, "Parametric psf estimation via sparseness maximization in the wavelet domain," in *Wavelet Applications in Industrial Processing II*, F. Truchetet and O. Laligant, Eds. November 2004, vol. 5607, pp. 26–33, Proceedings of the SPIE.

[125] Z. Wang, G. Wu, H.R. Sheikh, E.P. Simoncelli, E.H. Yang, and A.C. Bovik, "Quality-aware images," *IEEE Transactions on Image Processing*, vol. 15, no. 6, pp. 1680–1689, June 2005.

[126] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, no. 3, pp. 947–978, June 2007.

[127] Z. Xiong, M.T. Orchard, and Y. Zhang, "A deblocking algorithm for jpeg compressed images using overcomplete wavelet representations," *IEEE Transactions on Circuit and Systems for Video Technology*, vol. 7, no. 2, pp. 443–437, April 1997.

[128] V.K. Goyal, M. Vetterli, and N.T. Thao, "Quantized overcomplete expansions in rn: Analysis, synthesis and algorithms," vol. 44, no. 1, pp. 16–31, January 1998.

[129] H. Paek, R. Kim, and S. Lee, "On the pocs-based postprocessing technique to reduce the blocking artifacts in transform coded images," *IEEE Transactions on Circuit and Systems for Video Technology*, vol. 8, no. 3, pp. 358–367, June 1998.

[130] J. Mateos, A.K. Katsaggelos, and R. Molina, "A Bayesian approach to estimate and transmit regularization parameters for reducing blocking artifacts," *IEEE Transactions on Image Processing*, vol. 9, no. 7, pp. 1200–1215, July 2000.

[131] X. Li, "Improved wavelet decoding via set theoretic estimation," *IEEE Transactions on Circuit and Systems for Video Technology*, vol. 15, no. 1, pp. 108–112, January 2005.

[132] L. Moisan J.M. Morel A. Desolneux, S. Ladjal, "Dequantizing image orientation," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1129–1140, October 2002.

[133] Y.H. Chan and Y.H. Fung, "A regularized constrained iterative restoration algorithm for restoring color-quantized images," *Signal Processing*, vol. 85, no. 7, pp. 1375–1387, July 2005.

[134] D. Keysers, C.H. Lambert, and T.M. Breuel, "Color image dequantization by constrained diffusion," in *Proceedings of the SPIE*, January 2006, vol. 6058.

[135] E.P. Simoncelli, "The Steerable Pyramid: A flexible architecture for multi-scale derivative computation," in *Proceedings of the 2nd IEEE International Conference on Image Processing*, Washington DC, 23-26 October 1995, vol. 3, pp. 444–447, IEEE Signal Processing Society.

[136] A. Hirani and T. Totsuka, "Combining frequency and spatial domain information for fast interactive image noise removal," in *Proceedings of the ACM SIGGRAPH 1996*, 1996.

[137] J.S. De Bonet, "Multiresolution sampling procedure for analysis and synthesis of texture images," in *Proceedings of the ACM SIGGRAPH 1997*, 1997.

[138] J. Portilla and E.P. Simoncelli, "Texture model based on joint statistics of complex wavelet coefficients," *International Journal of Computer Vision*, vol. 40, no. 1, pp. 49–71, 2000.

[139] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proceedings of the SIGGRAPH 2000*, New Orleans, USA, July 2000.

[140] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 882–889, August 2003.

[141] H. Chen and I. Hagiwara, "Image reconstruction based on combination of wavelet decomposition, inpainting and texture synthesis," in *International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG'06)*, Plzen, Czech Republic, January 30 - February 3 2006.

[142] J.H. Kim, S.H. Lee, and N.I. Cho, "Bayesian image interpolation based on the learning and estimation of higher bandwavelet coefficients," in *Proceedings 13th IEEE International Conference on Image Processing*, Atlanta, GE, 8-11 October 2006, pp. 1269–1272, IEEE Signal Processing Society.

[143] M.M. Oliveira, B. Bowen, R. McKenna, and Y.S. Chang, "Fast digital image inpainting," in *Proceedings of the International Conference on Visualisation, Imaging and Image Processing (VIIP 2001)*, Marbella, Spain, 3-5 September 2001.

[144] J. Fadili, "Web page," *http://www.greyc.ensicaen.fr/ jfadili/.*

[145] B.E. Bayer, "Color imaging array," *U.S. Patent 3971065*, 1976.

[146] B.K. Gunturk, J. Glotzbach, Y. Altunbasak, R.W. Schafer, and R.M. Meresereau, "De-mosaicking: Color filter array interpolation in single chip digital cameras," *IEEE Signal Processing Magazine, Special Issue on Color Image Processing*, 2005.

[147] J. Portilla, D. Otaduy, and C. Dorronsoro, "Low-complexity linear demosaicing using joint spatial-chromatic image statistics," in *Proceedings of the 12th Interntational Conference on Image Processing*, Genoa, Italy, 11-14 September 2005.

[148] B.K. Gunturk and R.M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 997–1013, 2002.

[149] W. Lu and Y.P. Tan, "Color filter array demosaicing: new methods and performance measures," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1194–1210, 2003.

[150] X. Li, "Demosaicing by successive approximation," *IEEE Transactions on Image Processing*, vol. 14, no. 3, pp. 370–379, March 2005.

[151] S.C. Pei and I.K. Tam, "Effective color interpolation in ccd color filter arrays using signal correlation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 6, pp. 503–512, 2003.

[152] D. Alleysson, S. Susstrunk, and J. Herault, "Linear demosaicing inspired by the human visual system," *IEEE Transactions on Image Processing*, vol. 14, no. 4, pp. 1–12, April 2005.

[153] N.X. Lian and Y.P. Tan, "An efficient and effective color filter array de-mosaicking method," in *Proceedings 14th IEEE International Conference on Image Processing*, San Antonio, TX, 16-19 September 2007, IEEE Signal Processing Society.

[154] X. Li, "Web page," *http://www.csee.wvu.edu/ xinl/.*

[155] 40 scanned images, "Eastman kodak(c) photographic color image database," 1993.

[156] M.D. Fairchild, *Color Appearance Models*, Addison-Wesley, 1997.

[157] D. Barreto, L.D. Alvarez, R. Molina, A.K. Katsaggelos, and G.M. Callico, "Region-based super-resolution for compression," *Multidimensional Systems and Signal Processing, special issue on papers presented at the I International Conference in Super Resolution (Hong Kong, 2006)*, vol. 18, no. 2.

[158] C.A. Segall, A.K. Katsaggelos, R. Molina, and J. Mateos, "Bayesian resolution enhancement of compressed video," *IEEE Transactions on Image Processing*, vol. 13, no. 7, pp. 898–911, July 2004.

[159] S. Farsiu, M. Elad, and P. Milanfar, "Video-to-video dynamic super-resolution for grayscale and color sequences," *EURASIP Journal on Applied Signal Proccessing, Special Issue on Superresolution Imaging*, pp. 1–15, 2006, Article ID 61859.

[160] C.B. Atkins, C.A. Bouman, and J.P. Allebach, "Optimal image scaling using pixel classification," in *Proceedings of the 6th International Conference on Image Processing*, Thessaloniki, Greece, 7-10 October 2001, pp. 864–867.

[161] S. Battiato, G. Gallo, and F. Stanco, "Smart interpolation by anisotropic diffusion," in *Proceedings of 12th International Conference on Image Analysis and Processing*, Barcelona, Spain, 2003.

[162] D.D. Muresan and T.W. Parks, "Adaptively quadratic (aqua) image interpolation," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 690–698, May 2004.

[163] M.F. Tappen, B.C. Russell, and W.T. Freeman, "Exploiting the sparse derivative prior for super-resolution and image demosaicing," in *3rd International Workshop on Statistical and Computational Theories of Vision*, 2003.

[164] S. Farsiu, M. Elad, and P. Milanfar, "Multi-frame demosaicing and super-resolution of color images," *IEEE Transactions on Image Processing*, vol. 15, no. 1, pp. 141–159, January 2006.

[165] J.D. van Ouwerkerk, "Image super-resolution survey," *Image and Vision Computing*, vol. 24, no. 10, pp. 1039–1052, October 2006.

[166] G.H. Golub and C.F. Van Loan, *Matrix Computations*, John Hopkins University Press.