



UNIVERSIDAD DE GRANADA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA  
Y DE TELECOMUNICACIÓN

MÉTODOS BAYESIANOS APLICADOS A  
RESTAURACIÓN DE IMÁGENES EN  
PIRÁMIDES SOBRE-COMPLETAS Y A  
IDENTIFICACIÓN DE CÁMARAS  
DIGITALES.

Tesis Doctoral

Jose Antonio Guerrero Colón

Ingeniero en Informática

2008





DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN E  
INTELIGENCIA ARTIFICIAL  
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE  
TELECOMUNICACIÓN

MÉTODOS BAYESIANOS APLICADOS A  
RESTAURACIÓN DE IMÁGENES EN  
PIRÁMIDES SOBRE-COMPLETAS Y A  
IDENTIFICACIÓN DE CÁMARAS  
DIGITALES.

Tesis Doctoral

**Autor:** Jose Antonio Guerrero Colón  
*Ingeniero en Informática*

**Director:** Dr. Francisco Javier de la Portilla Muelas  
*Doctor Ingeniero de Telecomunicación*

2008



Título:  
MÉTODOS BAYESIANOS APLICADOS A RESTAURACIÓN DE  
IMÁGENES EN PIRÁMIDES SOBRE-COMPLETAS Y A  
IDENTIFICACIÓN DE CÁMARAS DIGITALES.

Autor:  
Jose Antonio Guerrero Colón

**Tribunal:**

Presidente : Rafael Molina Soriano

Vocales : Xosé R. Fernández Vidal

Mário A.T. Figueiredo

Eero P. Simoncelli

Secretario : Javier Mateos Delgado

Suplentes : Joaquín Fernández Valdívía

Nicolás Guil Mata

Acuerdan otorgar la calificación de:

Granada, 14 de Marzo de 2008



*A mis padres*





# Agradecimientos

En primer lugar quiero agradecer a Javier Portilla la confianza, dedicación y grandes dosis de paciencia que ha empleado en mi formación como investigador. Además de enriquecer el contenido de esta Tesis, me ha transmitido valores que van más allá del ámbito científico. También quiero expresar mi profundo agradecimiento a Carlos Dorronsoro Díaz cuyo interés, cercanía y amabilidad han sido pieza clave en el desarrollo de esta Tesis.

A Joaquín Fernández Valdivia le agradezco, además de la formación recibida, su predisposición a ayudarme en todo momento. En los mismos términos, agradezco a Rafael Molina Soriano el atento trato que he recibido por su parte, como investigador, profesor y persona. A Carlos Mantas Ruiz, por ofrecer siempre su amable atención en esta, a veces larga, travesía. También agradezco al Profesor José Muñoz Pérez, de la Universidad de Málaga, la amabilidad, atención y claridad que me ha brindado. Por supuesto, a todos los miembros del grupo al que pertenezco, Procesamiento de la Información Visual, y al resto de personas del Departamento de Ciencias de la Computación e Inteligencia Artificial.

De fuera de nuestras fronteras quiero agradecer a los Profesores Andy Harvey, Mário Figueiredo y Eero Simoncelli, tanto por darme la oportunidad de colaborar con ellos, como por el magnífico trato recibido. A los Profesores José Manuel Bioucas Dias y Aleksandra Pizurica por los comentarios e informes que han realizado sobre este trabajo. También me gustaría recordar a todos y cada uno de los compañeros que dejé en los centros que visité y que me hicieron sentir como en casa, a pesar de estar a cientos de kilómetros.

A todos mis amigos y compañeros que, de una forma u otra, han contribuido al desarrollo de esta Tesis, en especial a mi compañero de fatigas, el ya Doctor, Luis Mancera Pascual.

Para terminar quiero dedicar este trabajo a mis seres más queridos. A Lola, por caminar a mi lado durante todos estos años y proporcionarme el aliento que a veces me faltaba, gracias por estar ahí, quédate siempre. A toda mi familia, y muy especialmente a mis Padres, por el incondicional cariño y apoyo que me han dado desde que llegué a este mundo. A todos vosotros: este trabajo os pertenece.

*A la financiación de este trabajo han contribuido la Universidad de Granada, a través de dos contratos con el Centro de Investigación de la Armada (CIDA) e Iliá Sistemas y el Ministerio de Educación y Ciencia, a través del proyecto TEC2006/13845/TCM y el contrato FPU AP2005-2969.*



# Índice general

Índice de Figuras	xv
Índice de Tablas	xxi
<b>Part I: English Summary</b>	<b>1</b>
S.1. Introduction . . . . .	5
S.1.1. Goals and contributions of the Thesis . . . . .	8
S.2. Publications . . . . .	11
S.2.1. Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids . . . . .	13
S.2.2. Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids . . . . .	19
S.2.3. Image restoration using adaptive Gaussian scale mixtures in overcomplete pyramids . . . . .	37
S.2.4. Image denoising using mixtures of Gaussian scale mixtures .	55
S.2.5. Deblurring-by-Denoising using spatially adaptive Gaussian scale mixtures in overcomplete pyramids . . . . .	61
S.2.6. Low-cost wavefront coding using coma and a denoising- based deconvolution . . . . .	67
S.2.7. Realistic Bayesian camera identification based on demosaic- ing traces . . . . .	81
S.3. Conclusions and future work . . . . .	87
S.3.1. Conclusions . . . . .	87
S.3.2. Future work . . . . .	89

<b>Parte II: Versión en español</b>	<b>1</b>
<b>1. Introducción</b>	<b>3</b>
1.1. Objetivos y contribuciones de la Tesis . . . . .	7
<b>2. Fundamentos</b>	<b>11</b>
2.1. Introducción . . . . .	11
2.2. Campos aleatorios homogéneos . . . . .	11
2.3. Paradigma Bayesiano . . . . .	12
2.3.1. Estimación Bayesiana del filtro de Wiener . . . . .	14
2.3.2. Clasificador Bayesiano Gaussiano . . . . .	16
2.4. Representación de las imágenes naturales . . . . .	17
2.4.1. Las limitaciones de la transformada de Fourier . . . . .	17
2.4.2. Transformada de Fourier enventanada y sus limitaciones . .	17
2.4.3. Análisis multi-escala . . . . .	18
2.4.4. Análisis multi-resolución de imágenes digitales . . . . .	20
2.4.5. Idoneidad del análisis multi-resolución en el procesamiento de imágenes . . . . .	23
2.5. Introducción a las mezclas en escala de Gaussianas (GSM) . . . . .	25
2.5.1. Estadística de las imágenes en el dominio ondicular . . . . .	25
2.5.2. Modelado de las propiedades estadísticas no Gaussianas . .	28
2.6. Restauración de imágenes usando GSM . . . . .	29
2.6.1. Eliminación de ruido . . . . .	30
2.6.2. Eliminación de ruido y emborronamiento . . . . .	32
<b>3. Modelado con mezclas en escala de Gaussianas espacialmente   adaptativas</b>	<b>33</b>
3.1. Introducción . . . . .	33
3.2. Motivación . . . . .	36
3.3. GSM espacialmente adaptativas: SVGSM . . . . .	37
3.4. Eliminación de ruido usando SVGSM . . . . .	38
3.4.1. Estimación de parámetros . . . . .	38
3.4.2. Estimación de coeficientes . . . . .	39
3.5. Implementación . . . . .	39
3.5.1. Tratamiento de bordes y organización por bloques . . . . .	39
3.5.2. Experimentos . . . . .	41
3.5.3. Coste computacional . . . . .	41
3.6. Resultados y discusión . . . . .	42
3.6.1. Características del modelo . . . . .	42
3.6.2. Comparación con el estado del arte . . . . .	46
3.7. Conclusiones . . . . .	48

<b>4. Modelado con múltiples mezclas en escala de Gaussianas</b>	<b>49</b>
4.1. Introducción . . . . .	49
4.2. Motivación . . . . .	50
4.3. Mezcla de mezclas en escala de Gaussianas: MGSM . . . . .	52
4.3.1. Estimación de parámetros . . . . .	53
4.4. Eliminación de ruido usando MGSM . . . . .	55
4.4.1. Modelo de observación . . . . .	56
4.4.2. Estimación BLS de coeficientes . . . . .	56
4.4.3. Estimación de parámetros . . . . .	57
4.5. Implementación . . . . .	58
4.6. Resultados y discusión . . . . .	58
4.6.1. Características del modelo . . . . .	58
4.6.2. Eliminación de ruido . . . . .	64
4.7. Conclusiones . . . . .	67
<b>5. Deconvolución basada en métodos de eliminación de ruido</b>	<b>71</b>
5.1. Introducción . . . . .	71
5.2. Motivación . . . . .	72
5.3. Deconvolución basada en SVGSM . . . . .	75
5.3.1. Estimación de parámetros del prefiltro. . . . .	76
5.4. Implementación . . . . .	78
5.5. Resultados y discusión . . . . .	79
5.6. Conclusiones . . . . .	82
<b>6. Identificación Bayesiana de cámaras a partir de sus imágenes</b>	<b>83</b>
6.1. Introducción . . . . .	83
6.2. Formación de imágenes en cámaras digitales . . . . .	84
6.3. Extracción de características . . . . .	86
6.4. Clasificador Bayesiano para la identificación de cámaras . . . . .	88
6.4.1. Modelo de cámaras . . . . .	88
6.4.2. Regla de decisión . . . . .	90
6.5. Implementación . . . . .	91
6.6. Resultados y discusión . . . . .	91
6.6.1. Fuente de los datos . . . . .	91
6.6.2. Entrenamiento de los clasificadores . . . . .	92
6.6.3. Prueba de los clasificadores . . . . .	93
6.7. Conclusiones . . . . .	95
<b>7. Conclusiones y trabajo futuro</b>	<b>97</b>
7.1. Conclusiones . . . . .	97
7.2. Trabajo futuro . . . . .	99

---

<b>Apéndices</b>	<b>101</b>
<b>A. Estimación de parámetros MGSM</b>	<b>103</b>
A.1. Modelo de señal: estimación de parámetros . . . . .	103
A.1.1. Probabilidad $P_k$ de cada componente de la mezcla: . . . . .	103
A.1.2. Densidades $p_k(z)$ de los factores de escala: . . . . .	104
A.1.3. Matrices de covarianza $\mathbf{C}_k$ : . . . . .	104
A.2. Modelo de observación: demostración de estimación consistente . .	105
<b>B. Listado de publicaciones</b>	<b>107</b>
<b>Bibliografía</b>	<b>109</b>
<b>Índice alfabético</b>	<b>116</b>

# Índice de figuras

S.1.	Common degradations in restoration problems. On the left, <i>Einstein</i> image affected by additive noise. On the right, the same image degraded by blur and additive noise. . . . .	6
S.2.	Comparison of a natural image ( <i>Einstein</i> ) with a randomly generated one (white Gaussian noise). . . . .	7
S.3.	Contents scheme. . . . .	9
S.4.	Relations among publications and Chapters. . . . .	10
1.1.	Degradaciones típicas en problemas de restauración. A la izquierda la imagen de <i>Einstein</i> afectada por ruido aditivo. A la derecha la misma imagen afectada por emborronamiento y ruido. . . . .	4
1.2.	Comparación entre una imagen natural ( <i>Einstein</i> ) y una imagen generada aleatoriamente (ruido blanco Gaussiano). . . . .	5
1.3.	Esquema de contenidos. . . . .	8
1.4.	Relaciones entre publicaciones y Capítulos. . . . .	9
2.1.	Planos tiempo-frecuencia para diferentes dominios. . . . .	19
2.2.	Esquema genérico multi-resolución orientado: Análisis y Síntesis. La imagen de entrada es filtrada con las funciones base de análisis, obteniendo las subbandas asociadas a cada escala. Típicamente, se suele aplicar un esquema iterativo, que consiste en introducir el cuadro sombreado en el punto sólido del diagrama (que incluye filtrado paso bajo y submuestreo). La reconstrucción se obtiene de forma similar, pero aplicando los filtros de síntesis y sumando los resultados. Los componentes punteados son opcionales. . . . .	20
2.3.	Transformación wavelet (Haar) de la imagen de <i>Einstein</i> $256 \times 256$ . . . . .	22
2.4.	Transformación en 6 orientaciones y 4 escalas de la imagen de <i>Einstein</i> $256 \times 256$ , usando <i>Steerable pyramid</i> . . . . .	23
2.5.	División frecuencial de las representaciones utilizadas en esta Tesis. (a) <i>Full Steerable Pyramid</i> (FSP). (b) <i>Translation Invariant Haar Pyramid</i> (TIHP). Las regiones marcadas con una misma etiqueta representan el contenido frecuencial de una determinada subbanda. . . . .	25



2.6.	Subbanda vertical de alta frecuencia de la imagen de <i>Einstein</i> y su histograma. En (a) se observa que los valores bajos (en amplitud) son muy numerosos en comparación con los altos. En (b) se muestra el histograma, con la mayoría de los coeficientes concentrados en valores de baja amplitud, resultando un histograma muy no Gaussiano. . . . .	26
2.7.	Coefficientes en amplitud de la transformación wavelet (Haar) de la imagen de <i>Einstein</i> $256 \times 256$ . . . . .	27
2.8.	Función densidad de probabilidad condicional empírica de un coeficiente dado un vecino, en escala, orientación o espacio. La incertidumbre acerca de la amplitud del coeficiente vecino aumenta a medida que el coeficiente de referencia tiene mayor amplitud. Blanco indica alta probabilidad y negro probabilidad 0. Las columnas han sido normalizadas independientemente en amplitud (en lugar de en su integral) para facilitar la visualización. Figura tomada de [10]. . . . .	27
2.9.	Coefficientes estadísticos de una subbanda vertical de la imagen <i>Boats</i> y de una simulación usando GSM. (a) y (b): Histogramas marginales en escala logarítmica. (c) y (d): Histogramas condicionales de dos coeficientes vecinos (espaciales). Blanco representa alta probabilidad, negro probabilidad 0. Figura tomada de [10]. . . . .	29
3.1.	Variabilidad espacial de la autocovarianza dentro de una subbanda vertical (TIHP) de <i>Barbara</i> . Las muestras de autocovarianza corresponden a una vecindad GSM $3 \times 3$ . Cada una de ellas se calcula con las muestras de la región marcada en la imagen. . . . .	36
3.2.	Tratamiento de bordes y distribución de los bloques en un esquema multi-resolución. . . . .	40
3.3.	Comparación visual de resultados para las imágenes de <i>House</i> y <i>Barbara</i> . Se muestran regiones de $80 \times 80$ píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). El tamaño de bloque $B$ utilizado es de $32 \times 32$ para <i>House</i> y $16 \times 16$ para <i>Barbara</i> . De izquierda a derecha y de arriba a abajo se muestra ( <i>House/Barbara</i> PSNR en dB): original; ruidosa utilizando $\sigma_w = 20$ (22.11/22.11); método GSM [10] usando FSP (32.30/30.27); nuestro método con FSP (32.34/30.94); método GSM [10] usando TIHP (32.54/29.08); nuestro método con TIHP (32.71/30.19). . . . .	44

<p>3.4. Gráficas PSNR vs. desviación típica del ruido para las imágenes de <i>House</i> y <i>Barbara</i> (ajustadas entre 0 y 1) utilizando diferentes tamaños de bloque <math>B</math>. Línea discontinua azul para <math>B = 16 \times 16</math>, continua roja para <math>B = 32 \times 32</math> y línea verde punteada para <math>B = 128 \times 128</math>. Nótese como el tamaño de bloque óptimo se incrementa con el nivel de ruido. . . . .</p> <p>3.5. Comparación visual de resultados para las imágenes de <i>House</i> y <i>Barbara</i>. Se muestran regiones de <math>80 \times 80</math> píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo se muestra (<i>House/Barbara</i> PSNR en dB): observaciones ruidosas utilizando <math>\sigma_w = 10</math> (28.13/28.13); resultados de FoGSM [56] (35.63/35.01); resultados de SA-DCT [53] (35.98/33.48); resultados de 3DBM [52] (36.71/34.98); resultados de KSVD [54] (35.94/34.42); nuestros resultados usando <math>B = 16 \times 16</math> y las representaciones TIHP/FSP respectivamente (35.56/34.60). . . . .</p> <p>3.6. Incremento PSNR vs. PSNR de entrada, en dB, tomando como referencia el método presentado, SVGSM. Utilizamos TIHP para <i>House</i> y FSP con <i>Barbara</i>. Nuestros resultados, para <math>\sigma_w = \{5, 10, 15, 20, 25, 50\}</math>, se encuentran en la Tabla 3.2. Comparamos con cuatro métodos del estado del arte: línea punteada con círculos (azul) para 3DBM [52]; línea continua con cruces (lila) para SA-DCT [53]; línea discontinua con cuadrados (amarilla) para KSVD [54]; línea continua con círculos (turquesa) para GSM [10]; línea discontinua con rombos (gris) para FoGSM [56]; línea continua (roja) para nuestra aportación. . . . .</p> <p>4.1. Variabilidad de la covarianza en una subbanda vertical de TIHP. La imagen utilizada está formada por cuatro barras de dos texturas diferentes. Asumiendo una vecindad GSM <math>3 \times 3</math>, cada covarianza es estimada utilizando las muestras de la región asociada (cuadros de color). Se muestran las estimaciones utilizadas por el modelo GSM original (toda la subbanda, enmarcada con doble línea negra), modelo SVGSM (cuadros enmarcados con líneas punteadas negras y continuas verdes) y nuestra propuesta utilizando MGSM, usando 4 GSMs (cuadros enmarcados con línea azul discontinua). . . . .</p>	<p>46</p> <p>47</p> <p>48</p> <p>51</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------

- 4.2. Comportamiento del modelo al aumentar el número de componentes de la mezcla ( $K$ ) usando vecindades  $7 \times 7$ . En la esquina superior izquierda, la imagen original. Debajo, una subbanda de alta resolución de la representación *Steerable pyramid* [6], sin orientaciones. El cuadro  $2 \times 2$  restante, de izquierda a derecha y de arriba a abajo, representan la clasificación de acuerdo a la probabilidad a posteriori  $P(k|\mathbf{x}_m)$  que obtiene el modelo cuando se utilizan  $K = 2, 4, 6$  y  $8$  componentes GSM en el ajuste del modelo. . . . . 60
- 4.3. Comportamiento del modelo al aumentar el número de componentes de la mezcla ( $K$ ) usando vecindades  $7 \times 7$ . En la esquina superior izquierda, la imagen original. Debajo, una subbanda de alta resolución de la representación *Steerable pyramid* [6], sin orientaciones. El cuadro  $2 \times 2$  restante, de izquierda a derecha y de arriba a abajo, representan la clasificación de acuerdo a la probabilidad a posteriori  $P(k|\mathbf{x}_m)$  que obtiene el modelo cuando se utilizan  $K = 2, 4, 7$  y  $9$  componentes GSM en el ajuste del modelo. . . . . 61
- 4.4. Comportamiento del modelo usando la imagen de *House*. En la parte izquierda se muestra una subbanda de alta resolución (*Steerable pyramid*) no orientada. Debajo, se presenta la clasificación que realiza el modelo MGSM cuando se usan  $K = 8$  GSMs en la mezcla. Las vecindades con baja respuesta en energía han sido eliminadas (región en negro) para una mejor visualización. A la derecha de estas imágenes se muestran, por filas, los 6 primeros autovectores de cada componente, escalados por la raíz cuadrada de su autovalor correspondiente. . . . . 62
- 4.5. Dispersión de energía promedio en la vecindad  $7 \times 7$  utilizada en el ejemplo de la Figura 4.4. Se presenta la suma de los autovectores al cuadrado ponderada por sus correspondientes autovalores. El código de colores sigue el mismo criterio que en la citada figura. Los valores oscuros representan baja energía y los claros alta. . . . 63
- 4.6. Resultados visuales para los algoritmos de eliminación de ruido basados en GSM. A la izquierda la imagen de *House* y a la derecha la imagen de *Barbara*. Para una mejor visualización, se muestran regiones de  $128 \times 128$  píxeles comenzando en el píxel  $(24, 64)$  y  $(288, 64)$  respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo: Observación con ruido simulado (*House* con  $\sigma_w = 10$ , *Barbara* con  $\sigma_w = 25$ ); Resultado de GSM original [10]; Resultado de SVGSM [16]; Resultado de MGSM. Los valores PSNR pueden consultarse en la Tabla 4.1 . . . . . 64

4.7. Comparativa de resultados visuales, sobre las imágenes de *House* y *Barbara*, entre métodos del estado del arte y los resultados obtenidos con MGSM. Para una mejor visualización de los detalles se muestran regiones de  $80 \times 80$  píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo se muestra (*House/Barbara* PSNR en dB): observaciones ruidosas utilizando  $\sigma_w = 10$  (28.13/28.13); resultados de FoGSM [56] (35.63/35.01); resultado de [53] (35.98/33.48); resultados de [52] (36.71/34.98); resultados de [54] (35.94/34.42); nuestros resultados con MGSM (36.06/34.45). 68

4.8. Incremento PSNR vs. PSNR de entrada, en dB, tomando como referencia el método presentado, MGSM. Se muestran los resultados para las imágenes de *House*, *Barbara*, *Peppers* y *Boats*, todas ellas procesadas usando TIHP. Comparamos con cuatro métodos del estado del arte y nuestros métodos previos basados en GSM: línea punteada con círculos (azul) para 3DBM [52]; línea continua con cruces (lila) para SA-DCT [53]; línea discontinua con cuadrados (amarilla) para KSVD [54]; línea continua con círculos (turquesa) para GSM [10]; línea continua (roja) para SVGSM [16]; línea discontinua con rombos (gris) para FoGSM [56]; línea continua (verde) para el método MGSM. . . . . 69

4.9. Comparativa de rendimiento promedio para las imágenes de *House*, *Barbara*, *Peppers* y *Boats* tomando como referencia el método GSM. Las curvas muestran el incremento PSNR vs. PSNR de entrada para cuatro métodos del estado del arte y nuestros métodos previos basados en GSM: línea punteada con círculos (azul) para 3DBM [52]; línea continua con cruces (lila) para SA-DCT [53]; línea discontinua con cuadrados (amarilla) para KSVD [54]; línea continua (turquesa) para GSM [10]; línea continua con círculos (roja) para SVGSM [16]; línea discontinua con rombos (gris) para FoGSM [56]; línea continua (verde) para el método MGSM. . . . . 70

5.1. Esquema de dos pasos. . . . . 71

5.2.	Ilustración de las restricciones a la forma de los filtros en frecuencia cuando se utiliza una aproximación local (vecindad pequeña de tamaño fijo) junto con un esquema multi-resolución. Se muestra la misma señal limitada espacialmente en tres escalas diferentes. El esquema permite representar y compensar, siempre que no haya cruces por cero en frecuencia, filtros con variaciones lentas en las frecuencias altas y con variaciones rápidas en las frecuencias bajas (ver texto para más detalles). En la parte inferior de la Figura se muestran dos filtros paso bajo típicos (Gaussiano y uniforme, ambos $1D$ ) sobre sus correspondientes filtros inversos regularizados. El filtro Gaussiano puede ser compensado satisfactoriamente con esta aproximación, mientras el uniforme no. . . . .	74
5.3.	Razón de optimalidad del parámetro $\alpha$ . Ver texto para más detalles.	77
5.4.	Comparación visual de resultados para las imágenes de <i>House</i> y <i>Barbara</i> . Se muestran regiones de $80 \times 80$ píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo ( <i>House/Barbara</i> PSNR en dB): original; imagen degradada (24.11/23.33) usando PSF2 y $\sigma_w = \sqrt{0.308}$ para <i>House</i> y PSF1 y $\sigma_w = \sqrt{2}$ para <i>Barbara</i> ; resultados de [69] (34.76/29.05); resultados de [61] (34.76/27.09); resultados de GSM estándar [19] usando FSP (23.68/29.02); nuestra aportación [16] usando TIHP con <i>House</i> y FSP con <i>Barbara</i> (33.11/30.17). . . . .	81
5.5.	Resultado del método de deconvolución SVGSM aplicado a una degradación real. . . . .	82
6.1.	Arriba: Esquema de la formación de la imagen. Abajo: Mosaico de Bayer. . . . .	85
6.2.	Mosaicos locales para un patrón de Bayer clásico. Figura tomada de [21] . . . . .	86
6.3.	Comparativa de porcentaje de aciertos para un modelo Gaussiano y otro de mezcla de Gaussianas (incluyendo desviación media). . .	95
6.4.	Comparativa para reglas de decisión por voto y modelo producto. Arriba para un modelo de cámara Gaussiano, abajo para un modelo de mezcla de Gaussianas. En este último caso las barras rojas indican la desviación típica respecto de la media. . . . .	96

# Índice de Tablas

3.1. Tiempo de computación promedio en segundos (10 repeticiones) comparando diferentes tamaños de imágenes, bloques y representaciones. . . . .	41
3.2. Resultados de eliminación de ruido medido como relación señal ruido pico, en dB. La primera columna muestra la desviación típica del ruido y el valor PSNR inicial. Las siguientes cuatro columnas corresponden a los resultados del método GSM [10] (etiquetado como GSM) y a nuestra aportación (etiquetada como SVGSM) utilizando las representaciones TIHP y FSP (se resaltan en negrita los mejores resultados). En la última columna se muestra el tamaño de bloque $B$ utilizado para SVGSM, siendo $F$ el tamaño de la imagen completa. . . . .	43
4.1. Rendimiento del método de eliminación de ruido en términos de relación señal ruido pico, en dB (PSNR). La primera columna muestra la desviación estándar del nivel de ruido, así como la PSNR asociada. En cada fila se muestran resultados para diferentes métodos BLS basados en GSM, todos ellos para la representación TIHP: GSM [10], espacialmente variante (SVGSM) y el nuevo método presentado (MGSM). . . . .	65
4.2. Rendimiento del método de eliminación de ruido en términos de relación señal ruido pico, en dB (PSNR). La primera columna muestra la desviación estándar del nivel de ruido, así como la PSNR asociada. En cada fila se muestran resultados para diferentes métodos que representan el estado del arte, así como los resultados obtenidos con MGSM. . . . .	67
5.1. Resultados promedio del conjunto de entrenamiento (ver Sección 5.3.1) en términos de incremento de relación señal ruido (ISNR) en dB. . . . .	79

---

5.2. Comparación de nuestra aportación [16] con otros métodos de la literatura, en términos de incremento de relación señal ruido (ISNR) en dB. La primera fila muestra el filtro de convolución utilizado en la degradación (ver texto para más detalles) y la segunda muestra las desviaciones estándar del ruido. Los mejores resultados en cada caso se resaltan en negrita. . . . .	80
6.1. Porcentaje de aciertos para distintos tamaños del conjunto de entrenamiento. La primera columna muestra los resultados de modelar las clases como Gaussianas. La segunda columna muestra, para un modelo de mezcla de Gaussianas, el porcentaje de acierto promedio junto con la desviación típica para un total de 100 repeticiones. . . . .	94

# Part I: English Summary





BAYESIAN METHODS APPLIED TO  
IMAGE RESTORATION  
IN OVERCOMPLETE PYRAMIDS  
AND TO  
DIGITAL CAMERA IDENTIFICATION

Thesis dissertation



# Abstract

Bayesian paradigm is a useful framework for the development of many image processing applications. When dealing with restoration methods, this framework implies the use of statistical models capturing our *a priori* knowledge about natural images. The study of the image statistical properties has motivated the development of many representations suitable for capturing such properties. In particular, overcomplete oriented pyramids are among the ones providing better results in restoration applications. This fact explains one of the current tendencies: to develop statistical models in overcomplete domains. These models are crucial for the applications performance. An example is a recent local image model based on Gaussian scale mixtures (GSM). Taking the latter as a starting point, we describe in this Thesis two new image models that are applied to image denoising. Despite these models share a common background, they present very different approaches: a local vs a non-local characterization. Another important restoration application is image deconvolution under additive Gaussian noise. This problem, compared to image denoising, is a more difficult, ill-posed inverse problem, even if, as in this work, the degradation sources are assumed known. In this Thesis we develop a solution based on a two-step restoration scheme: (a) linear global blur compensation, and (b) non-linear local denoising. In such a way, the problem is decoupled into a global filtering and a non-white noise removal, reducing its complexity. In addition, this strategy allow us to apply and take advantage of very powerful denoising methods (as the ones presented here).

Bayesian tools are also widely applied to classification problems. In this Thesis we present two simple statistical models for a recently proposed camera descriptor that are used to develop a digital camera identification algorithm within a Bayesian framework.



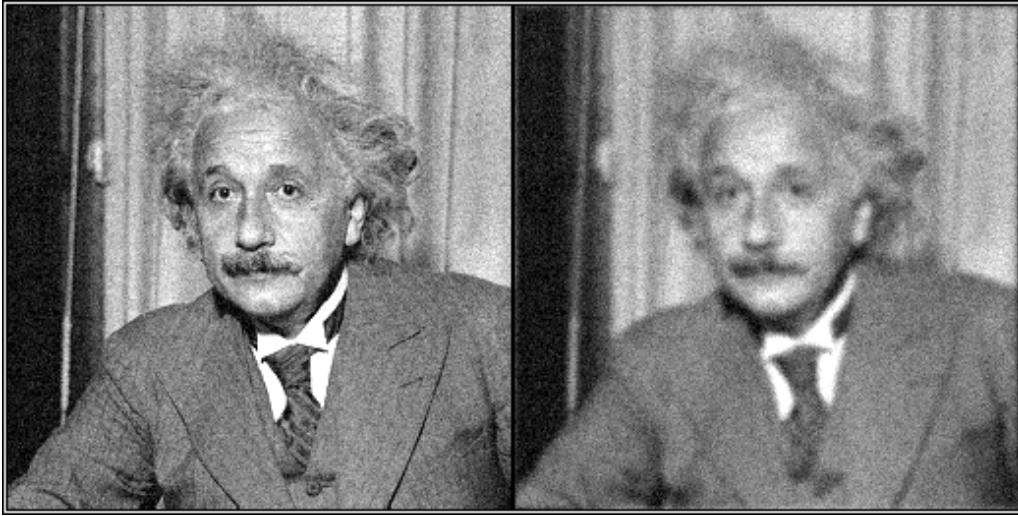
## S.1. Introduction

The new advances of the digital age have turned the images, even more than in the past, into one of the most important information vehicles. In a short period of time, the market of digital image acquisition devices has changed from a (relative) limited number of users, mainly professionals, to a worldwide market, comparable to that of mobile phones. The industry has flooded the market with these devices, improving their features and providing better and better quality at less cost. These facts have contributed to the increasing interest of researchers in the image processing field. The ubiquity of digital images has motivated the proliferation of applications in very different environments and with very different goals: codification, compression, surveillance, identification, restoration, fusion, classification, etc... to name only a few in the vast area of digital image processing [1]. The literature provides plenty of solutions to these problems. Despite many of them follow an ad-hoc approximation, the use of statistical models is becoming more and more widespread. Focussing on the involved information, this trend can be reasonably explained, as most of the applications share the same input information, i.e., the *natural images*, meaning by natural images those which capture scenes of the real world, in the form that we usually observe it. The common use of statistical models can be also explained considering how the knowledge about natural images is used. From this point of view, the explanation might be in the widespread use of the Bayesian paradigm, as it explicitly addresses the use of our prior knowledge. In addition, this paradigm can be used in the great majority of statistical inference problems. Such flexibility makes more appealing the use of statistical models, as they can be exploited in a wide variety of applications.

The Bayesian paradigm is widely used for image restoration. The main goal of restoration is to get an *estimation* of an ideal image from a degraded version of it. There are many areas (e.g., domestic, scientific, technical, industrial, defense, security, jury, etc...) where the processing of degraded images is useful, not only for improving their visual aspect, but also to allow a better interpretation of their content, both by humans and by computers. The word *degradation* may be considered in a wide sense, as there are many degradations sources, such as loss of information ("dead"pixels), blur, noise addition, blur under additive noise, etc... In Figure S.1 we show a natural image affected by the latter two degradations. The applications to compensate for them are also known as denoising and deconvolution (or deblurring) respectively. We devote the greater part of this Thesis to the development of these applications. The Bayesian paradigm uses, through Bayes rule, the information about the degradation as well as the *a priori* knowledge of natural images. In this work we assume<sup>1</sup> that the degradation process is perfectly known and correctly modelled. Under those conditions, the success

---

<sup>1</sup>This assumption is widely made in the literature.



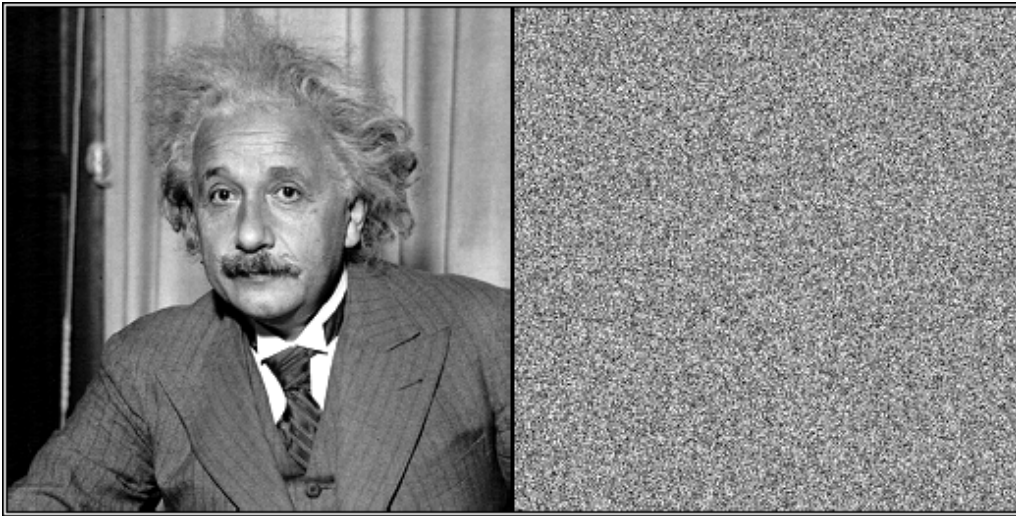
**Figure S.1.** Common degradations in restoration problems. On the left, *Einstein* image affected by additive noise. On the right, the same image degraded by blur and additive noise.

of Bayesian image restoration, when measured through simulations using known images as ideal references, mainly relies on the existence of a good statistical model for natural images.

To develop image models suitable to discern signal (image) from degradation (e.g. noise) one should be able to answer the following question: which features distinguish the natural images? Figure S.2 shows a natural image (left) beside a synthetic image of Gaussian noise (right). At a glance, a simple visual inspection allows us to extract significant differences, like the presence of smooth and structured areas in natural images, in contrast to the unstructured randomness showed by the artificial ones. Natural images have been widely analyzed in the literature, discovering many particular properties. Three of the most important ones are the following [2, 3]:

- High correlation among neighbor pixels.
- Presence of local oriented patterns, like edges, bars, corners, etc...
- The statistical properties are, approximately, scale invariant.

These features are closely related to biological visual models [4], motivated by the coincidence between the scenes acquired in the natural images and the stimuli of the real world to which the visual system has adapted through millions of years. The statistical properties of the images, together with the mentioned relations, have motivated a change in the representation, leading researchers to propose many transformations inspired in the human visual



**Figure S.2.** Comparison of a natural image (*Einstein*) with a randomly generated one (white Gaussian noise).

system (such as [5, 6]). Among them, critically sampled wavelets (orthogonal or bi-orthogonal) strongly reduce the correlation existing among neighbor coefficients, providing both scale and orientation selectivity. Such properties make these representations appropriate to efficiently represent natural images (e.g., JPEG2000 [7]). However, in terms of restoration performance (measured using the mean squared error, MSE) overcomplete oriented pyramids have proved to be more suitable representations, adding to the selectivity in scale/orientation two key advantages: translation invariance and more reliable estimations due to redundancy increase. Furthermore, Raphan *et al.* [8] have mathematically proved that, the expected image-domain MSE resulting from an overcomplete representation is less than or equal to that resulting from an original non-redundant representation.

Because of all previous reasons, it is nowadays quite common to develop image models in overcomplete domains rather than in the pixel domain. As before, we need to raise the same question, but with an important modification: which features distinguish the natural images, but now, in the overcomplete domain? Again, we get an answer from the study of the statistical properties of wavelet coefficients. One of the current references is the Gaussian Scale Mixture (GSM) model [9]. This model gives support to successful restoration applications, such as *et al.* [10]. The latter reference represents the starting point of this Thesis.

Another important research area is digital image forensics. The scientific and technical advances in acquisition devices have motivated a similar development of the image editing and processing software. Hence, it has become relatively simple, even for non experts, to manipulate the content of digital images with



very different purposes, such as removal/addition of contents, secret information introduction (steganography), watermarking and enhancement, just to name a few. In recent years, the interest in digital image forensics has significantly increased, both in the scientific community and in the security and defense area, giving raise to applications such as manipulation detection, coding/decoding of secret messages, authentication, etc...

Digital camera identification represents another interesting application within the forensics field. Here, the main goal is to identify the acquisition device used to capture a single observed digital image. The Bayesian paradigm provides a generic framework to measure the probability that a given observation belongs to a certain class. In many image classification problems, the input information is, again, a natural image. However, and in contrast to restoration applications, here the prior information is no longer an image model, but a discrete value representing the probability of each camera. Now, the performance of the application mainly relies on both, the discriminant features selection and the statistical modelling of the features given each class (equivalent to the degradation model in restoration), and not so much on the (still relevant) prior information. Following the parallelism with restoration applications, we should now answer this question: Which features of the digital images minimize the overlapping among the probability distributions of the camera classes? In this Thesis we use the traces of CFA demosaicing filters as features to describe digital cameras. We present two different statistical features models that are used to develop a Bayesian camera identification method.

### **S.1.1. Goals and contributions of the Thesis**

The main goals of this Thesis are summarized in its title: to use the Bayesian paradigm for developing image restoration methods (through estimation) and camera identification algorithms (through classification). In particular, this Thesis aims to cover the following objectives:

1. To propose new statistical models of natural images in overcomplete domains that overcome some limitations of our reference model (original local GSM [10]), increasing their strengths.
2. To use those new image models under a Bayesian estimation framework to develop restoration applications, like denoising and deblurring.
3. To propose statistical models of digital camera descriptors that, under a Bayesian classification scheme, result in a camera identification algorithm.

In Figure S.3 we show an schematic view of the general topics covered by this Thesis, putting into context our contributions (framed with double red lines) and the connections among all the treated subjects. Next we describe the Chapters

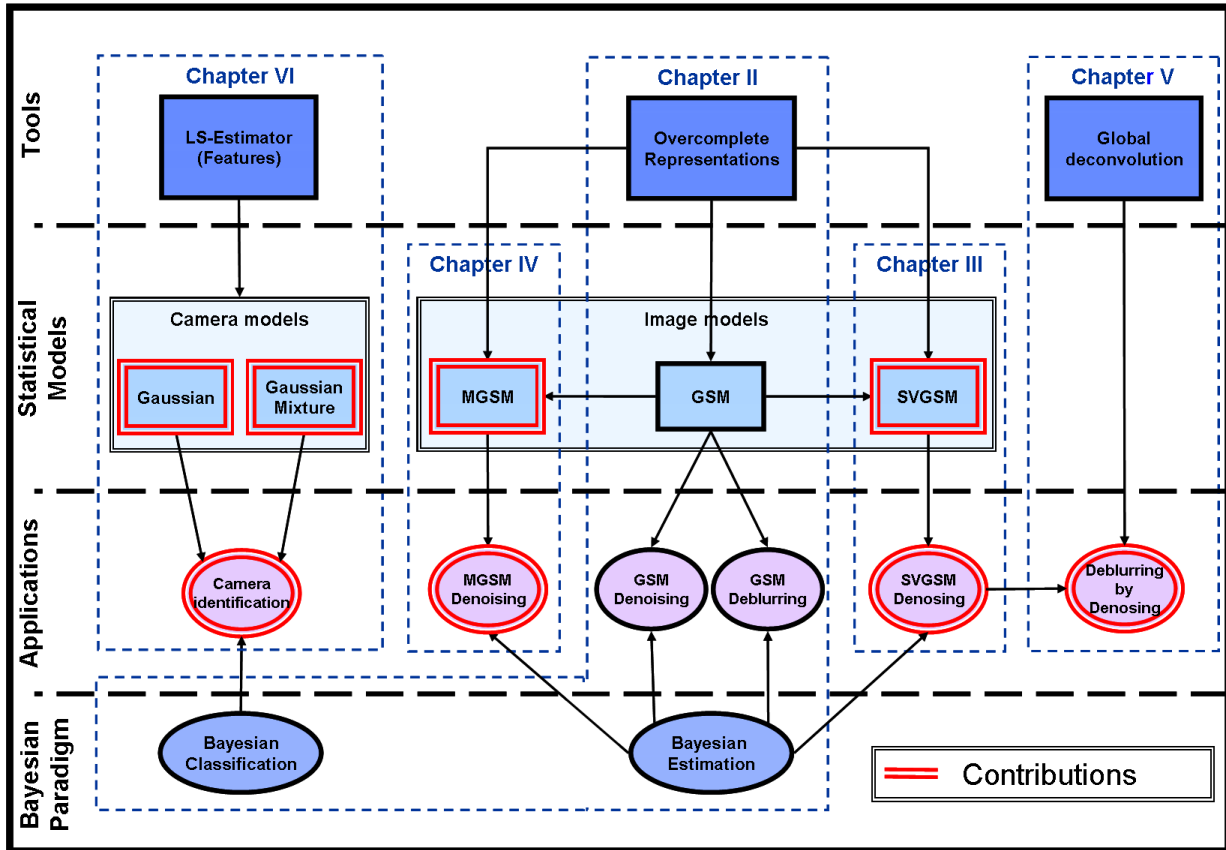
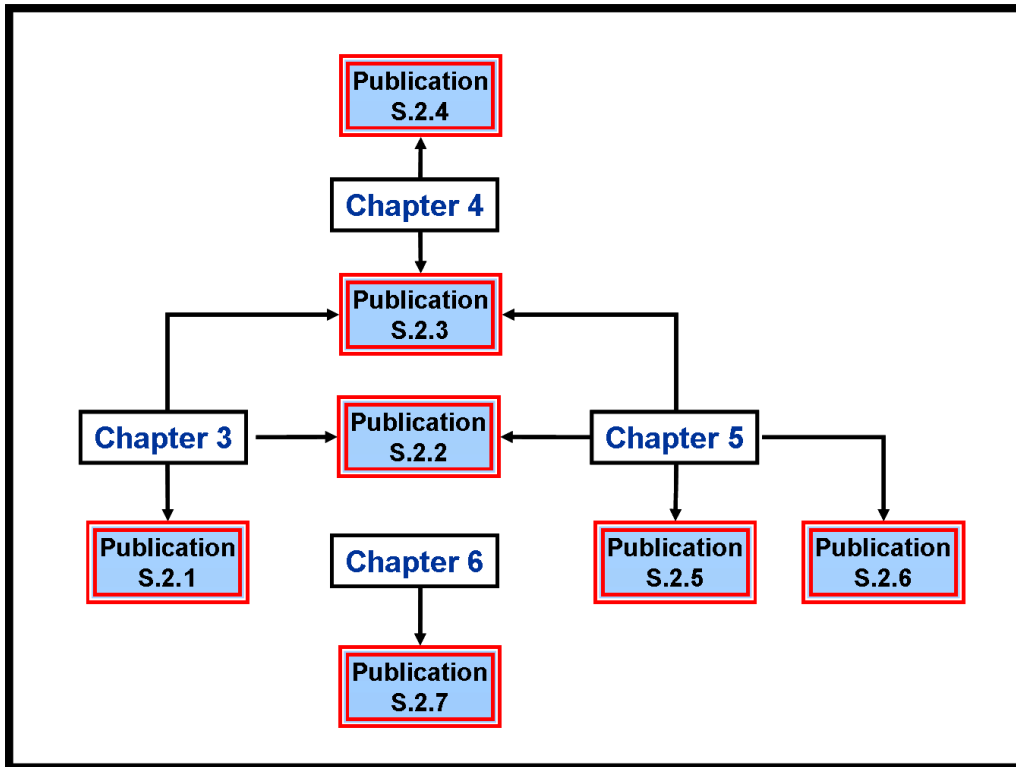


Figure S.3: Contents scheme.

content as well as the related publications included in the next Section of this Summary.

Chapter 2 describes the basic background needed for a better understanding of this Thesis. We include, among other things, a detailed description of the original GSM model. We recommend the reading of [10] to those that are not familiar with the latter model. In Chapters 3 and 4, the core of this Thesis, we describe two new statistical models that are used to develop two denoising algorithms. Despite both proposals take as reference the local GSM model, each one represents very different approaches. On the one hand, we describe a solution based on local statistics (Chapter 3) and on the other hand a solution based on non-local statistics (Chapter 4). The next Chapter presents a restoration algorithm (deconvolution under additive Gaussian noise) that propose a solution decoupled in two steps: a linear global deconvolution and a non-linear local denoising. In Chapter 6, which covers the last goal of this Thesis, we describe two simple statistical models that, coupled with the Bayesian classification paradigm, allow us to develop the desired



**Figure S.4:** Relations among publications and Chapters.

camera identification algorithm. Finally, Chapter 7 ends this Thesis, presenting the general conclusions and future work. The English version of that Chapter is available in the last Section of this Summary.

In Figure S.4 we present the relations among our publications and the contributions of this Thesis grouped by Chapters (from 3 to 6). As we can see, we have included a conference paper that summarizes the content of each Chapter, with the exception of Chapter 5 that counts with another one (attached in Section S.2.6) describing an interesting optical application of our deblurring algorithm. The content of Chapters 3 and 5 is fully covered by the journal publication attached in Section S.2.2. In addition, these contents are concisely treated in the publication of Section S.2.3, which also includes a simplified version of the new statistical model proposed in Chapter 4 (MGSM).

## S.2. Publications

In this section we attach the publications that give support to this Thesis. The full references are listed in Appendix B. Among them, we include four conference papers [11, 12, 13, 14], one patent<sup>2</sup> [15] and one international journal paper [16], all of them published. In addition, we include two recent conference papers submitted to the International Conference on Image Processing 2008 [17, 18] that are under review.

---

<sup>2</sup>We don't include this patent as it has not been yet translated into English.



## S.2.1. Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids

- J. A. Guerrero-Colón and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids", IEEE Int'l Conf. on Image Proc., vol. I, pp. 105-108, Genoa (Italy) September 2005.



# TWO-LEVEL ADAPTIVE DENOISING USING GAUSSIAN SCALE MIXTURES IN OVERCOMPLETE ORIENTED PYRAMIDS

Jose A. Guerrero-Colon<sup>\*</sup> and Javier Portilla<sup>†</sup>

Visual Information Processing Group  
Dept. of Computer Science and Artificial Intelligence  
Universidad de Granada  
{jaguerrero, javier}@decsai.ugr.es

*We describe an adaptive denoising method for images decomposed in overcomplete oriented pyramids. Our approach integrates two kinds of adaptation: 1) a 'coarse' adaptation, where a large window is used within each subband to estimate the local signal covariance; 2) a 'fine' adaptation, which uses small neighborhoods of coefficients modelled as the product of a Gaussian and a hidden multiplier, i.e., as Gaussian scale mixtures (GSM). The former provides adaptation to local spectral features, whereas the latter adapts to local energy fluctuations. We formulate our method as a Bayes Least Squares estimator using spatially variant GSMs. We also discuss the importance of image representation, compare the results using two different representations with complementary features, and study the effect of merging their results. We demonstrate through simulation that our method surpasses the state-of-the-art performance, in a  $L_2$ -norm sense.*

## 1. INTRODUCTION

Spatially adaptive models have been widely used for image denoising. In 1980, Lee proposed an adaptive Wiener method in the pixel domain [1]. With the introduction of *wavelets* (in the broad sense of the word), the local adaptation in the new domain became much more powerful. Some authors (e.g. [2, 3, 4]) set local thresholds for the coefficients depending on contextual measurements. Other authors continued developing the local Wiener idea in the new domain. Mihçak et. al. [5] included a prior on the local variance. A similar Bayesian approach was followed in [6]. Simoncelli et. al. [7] refined the Bayesian model using full signal and noise covariance, under a new Gaussian scale mixture (GSM) frame. All these methods followed an empirical Bayes strategy: first, estimate the local en-

ergy, and then, apply a Wiener filter using that estimation. Portilla et. al. [8], instead, introduced a single-step Bayes Least Squares estimator, further increasing the denoising performance. The referred adaptive Wiener methods either consider just the local variance, disregarding the covariance structure for each subband [5, 6], or consider the latter fixed, up to a scale factor [7, 8]. In both cases they perform a 'fine' adaptation, in the sense that local energy estimation is carried out using small neighborhoods.

Broad literature is available on local covariance estimation and its applications (e.g. [9]). In image denoising, this kind of estimation allows for a selective preservation of salient local spatial frequencies. However, as covariance estimation requires a much larger window than variance estimation, the result is a too 'coarse' adaptation. Thus, the extra complexity does not pay off in performance terms.

In this work we integrate both referred types of adaptation. We estimate the local covariance in large neighborhoods of the image subbands, and, once we are adapted to those features, we use small neighborhoods as in [8] to account for local variance fluctuations. This two-level adaptation results in a significant performance increase with respect to current state-of-the-art methods.

Besides adaptation, another critical issue is image representation. In this work we use two complementary representations: the full steerable pyramid [8], and an overcomplete version of the Haar wavelet [10]. We discuss the use of these representations, and, as in [11], show the potential benefits of combining their results.

## 2. IMAGE MODELLING

### 2.1. Image Representation

In recent years, the power of statistical image models has been increased by transforming the image from the pixel domain to new representations. Overcomplete oriented pyramids (such as the steerable pyramid, the curvelet or the complex wavelets), have demonstrated to better fit the statistical

<sup>\*</sup>JAGC is supported by 2050-559 grant contract, funded by CIDA, Ministerio de Defensa (Spain).

<sup>†</sup>JP is supported by TIC2003-01504 grant, funded by the Ministerio de Ciencia y Tecnología (Spain) and the "Ramón y Cajal" program.



properties of typical images rather than single-scale representations. Each of the referred representations fits best for processing certain image features, but none is superior in absolute terms. For this reason, some authors have proposed to combine them. The simplest way to combine two representations is to average their results [11].

In this work we have used two representations: the Full Steerable Pyramid (FSP) and the Trapezoidal Undecimated Pyramid (TUP). The FSP is an extension of Simoncelli’s original steerable pyramid, including orientation selectivity in the high-pass residual [8]. The major advantages of the FSP are that (1) it is flexible in its number of orientations, and (2) their filters are highly selective in frequency. Its drawbacks are the large spatial support of the filters and the high computational cost derived from its large redundancy (best denoising results are reported for 8 orientations [8], a redundancy factor close to 19). TUP is an overcomplete version of the Haar wavelet, obtained by decimating its  $a$  *trous* subbands from the third scale on, counting upwards. This gives rise to a *trapezoidal* oriented pyramid with a redundancy factor of 7. Unlike the Haar wavelet, and like FSP, TUP is invariant to translation, and it still has (almost) perfect reconstruction (e.g., PSNR of 53.6 dB for *Lena* reconstruction). Complementary to FSP, TUP’s kernels are very small, thus providing excellent spatial localization. TUP’s weaknesses are its poor performance for processing oblique orientations and the reduced spectral localization of its kernels. According to these features, FSP is more powerful for processing texture-rich images, and TUP for processing images made of flat patches with well-defined boundaries (especially if oriented horizontally and vertically). Due to their complementary character, averaging the results obtained with each representations will generally increase the overall performance.

## 2.2. Spatially Variant GSM

The GSM model in the wavelet domain [12] is able to represent local variance fluctuations, a basic feature of typical images that explain most of the non-linear statistical patterns reported in recent years. This model assumes a fixed signal covariance for all neighborhoods in the same subband. However, typical images have different spatial covariance at different locations. Note also that, although the filtering associated with each subband reduces the spatial variability of the covariance within a subband, it does not remove it completely. Thus, to further increase the adaptability of our model, we use a GSM model with a signal covariance that changes at different regions of the image.

Let’s consider an image decomposed into oriented subbands at multiple scales. We denote as  $x_c^j(n, m)$  the coefficient corresponding to the subband  $j$ , centered at spatial location  $(n, m)$ . We denote as  $\mathbf{x}^j(n, m)$  a *neighborhood* of coefficients clustered around this *reference coefficient*

(see neighborhood structure in section 4). We model these neighborhoods as Gaussian scale mixtures (GSMs). A random vector  $\mathbf{x}$  is a GSM [13] if it can be expressed as the product of two independent random variables:  $\mathbf{x} = \sqrt{z}\mathbf{u}$ , where  $z$  is a positive scalar and  $\mathbf{u}$  is a zero-mean Gaussian vector. We model the desired spatially variant behavior by defining  $\mathbf{C}_u^{j,(n,m)}$  as the (spatially variant) covariance of  $\mathbf{u}^j(n, m)$ . The density of  $\mathbf{x}$  is determined uniquely by  $p_z(z)$  and  $\mathbf{C}_u^{j,(n,m)}$  as follows:

$$\begin{aligned} p^{j,(n,m)}(\mathbf{x}) &= \int p^{j,(n,m)}(\mathbf{x}|z) p_z(z) dz \\ &= \int \frac{\exp\left(-\mathbf{x}^T (z\mathbf{C}_u^{j,(n,m)})^{-1} \mathbf{x}/2\right)}{(2\pi)^{N/2} |z\mathbf{C}_u^{j,(n,m)}|^{1/2}} p_z(z) dz, \end{aligned}$$

where  $N$  is the dimensionality of  $\mathbf{x}$  and  $\mathbf{u}$  (the size of the neighborhood). Note that  $\mathbf{x}^j(n, m)$  is conditionally Gaussian for a given  $z$ . Following [8], we have assumed the mixing density  $p_z(z)$  constant in  $\log z$  for all the subbands in the working range.

## 3. TWO-LEVEL ADAPTIVE DENOISING

We assume the image is corrupted by additive zero-mean independent Gaussian noise of known spectral density. Our image denoising method, as it is usual, performs the denoising in the *wavelet* domain and then invert the pyramid. Following the GSM model, we can express a neighborhood of noisy coefficients as <sup>1</sup>  $\mathbf{y} = \sqrt{z}\mathbf{u} + \mathbf{w}$  where  $\mathbf{w}$  is the zero mean Gaussian noise vector with covariance matrix  $\mathbf{C}_w$ . Note that  $\mathbf{C}_w$  (unlike the signal covariance  $\mathbf{C}_u$ ) is assumed constant for all neighborhoods within a subband (but different for each subband).

### 3.1. Parameter estimation

For a given  $z$ , the covariance of a vector of observations is  $\mathbf{C}_{\mathbf{y}|z} = z\mathbf{C}_u + \mathbf{C}_w$ . Because of the independence of  $z$ ,  $\mathbf{u}$  and  $\mathbf{w}$ , and taking expectations over  $z$ , it yields:  $\mathbf{C}_{\mathbf{y}} = \mathbb{E}\{\mathbf{C}_{\mathbf{y}|z}\} = \mathbb{E}\{z\}\mathbf{C}_u + \mathbf{C}_w$ . Setting  $\mathbb{E}\{z\} = 1$ , we obtain  $\mathbf{C}_{\mathbf{y}} = \mathbf{C}_u + \mathbf{C}_w$ .  $\mathbf{C}_w$  is computed at each subband by applying the pyramidal representation to a deterministic function with the same sample autocorrelation as the noise. To estimate the spatially variant signal covariance  $\mathbf{C}_u$  we assume that it changes smoothly over the image, and, thus, that it may be sampled by estimating it into a set of non-overlapping regions. We divide each subband into  $B \times B$  blocks, which are assumed uniform in covariance. For each of this blocks we estimate  $\mathbf{C}_u$  by subtracting  $\mathbf{C}_w$  from the sample covariance of its coefficients,  $\mathbf{C}_{\mathbf{y}}$ .

<sup>1</sup>For notational simplicity, we drop the superscript  $j$  and the indices  $(n, m)$  in the following.

### 3.2. Coefficient estimation

For each neighborhood  $\mathbf{y}$  of observed coefficients, we estimate  $x_c$ , the reference coefficient, at the center of the neighborhood. The Bayes least squares (BLS) estimate can be expressed in this case [8] as

$$\mathbb{E}\{x_c|\mathbf{y}\} = \int_0^\infty p(z|\mathbf{y}) [\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\}]_{x_c} dz. \quad (1)$$

Thus, the solution is written as the average of the Bayes least squares estimate of  $x_c$  when conditioned on  $z$ , weighted by the posterior density of  $z$ ,  $p(z|\mathbf{y})$ . The advantage of the GSM model is that the coefficient neighborhood vector  $\mathbf{x}$  is Gaussian when conditioned on  $z$ . This fact, coupled with the assumption of additive Gaussian noise means that the expected value inside the integral of (1) is simply a Wiener estimate:

$$\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\} = z\mathbf{C}_u(z\mathbf{C}_u + \mathbf{C}_w)^{-1}\mathbf{y}.$$

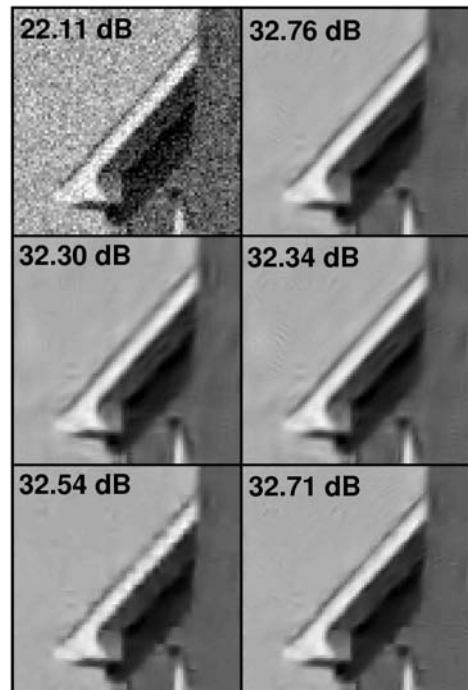
We can simplify the dependence of this expression on  $z$  by diagonalizing the matrix  $z\mathbf{C}_u + \mathbf{C}_w$  (see [8]). Note that, unlike in [8] now this diagonalization depends on the spatial location (as  $\mathbf{C}_u$  is a function of  $(n, m)$ ), and, thus, it needs to be computed once for each block of each subband.

## 4. IMPLEMENTATION

We have used for both representations 4 scales for  $256 \times 256$  images and 5 scales for  $512 \times 512$  images. The number of orientations is fixed for the TUP (3), and 8 for the FSP as in [8]. For both representation we have used a  $3 \times 3$  GSM neighborhood around each coefficient. In the case of FSP we also included a coefficient at the same location and orientation at the next coarser scale (a *parent*), as in [8]. We have tested different size blocks  $B$  for estimating the signal covariance ( $B = \{16, 32, 64, 128\}$ ). We have used the same block size  $B$  for all pyramid levels. Finally, we have been very careful to avoid boundary artifacts, a critical issue for block-based image processing. We have used non-overlapping blocks, with overlapping extensions, which become mirror reflections at the image boundaries.

## 5. RESULTS AND DISCUSSION

In Table 1 we show the results of the method applied to four standard test images and four noise levels. We have included a comparison to the original BLS-GSM method (a single level of adaptation, *1LA*), both with the FSP and the TUP. These results show the performance gain derived from the two-level adaptation (*2LA*) strategy. It is significant, though, how some images experience a big improvement (e.g., *Barbara*, an image having widely varying high frequency texture), whereas in other cases the improvement



**Fig. 1.** Visual comparison results on *House* image (cropped to  $80 \times 80$ ). From left to right and top to bottom: Noisy ( $\sigma_w = 20$ ); Our FSP and TUP results averaged; BLS-GSM [8] with FSP; Our method with FSP; BLS-GSM [8] with TUP; Our method with TUP.

is modest (e.g., *Lena*, which does not present local spectral peaks). They also show how different representations may fit better different images, depending on their texture and edges contents: *House* and *Peppers* are best denoised using the TUP, whereas for *Lena* and *Barbara* we obtain better results with the FSP. The second column counting from the right shows the effect of averaging the denoised images obtained using each representation. We observe that, except for *Barbara*, this provides an extra performance gain, which in some cases (*House*) becomes very significant. Finally, the last column shows the block size used for the reported results (excluding the averaged ones), the one providing the best results for each case. We observe that this parameter depends on the particular image, and that it tends to increase with the noise level.

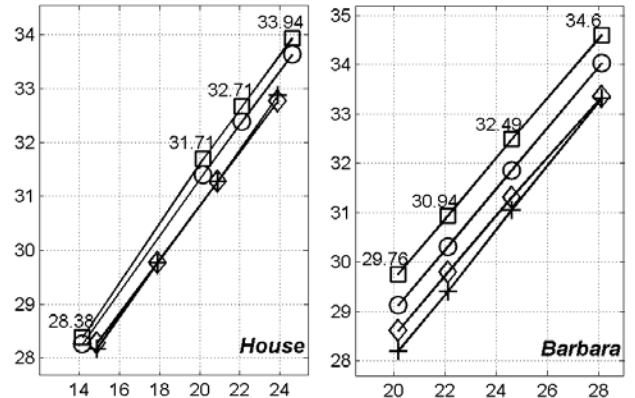
Fig. 1 shows some visual results (a  $80 \times 80$  pixel patch of *House*). This figure allows to visually interpret: 1) the effect of our two-level adaptation (right column) vs. one-level adaptation (BLS-GSM, left column); 2) the effect of using different representations (FSP in the middle row, and TUP in the bottom row); and 3) the effect of averaging both of them (up right corner). Note how many artifacts on the roof line obtained with the TUP representation disappear when using two-level adaptation. Finally, Figure 2 shows a PSNR comparison between our results (from Table 1, excluding the averaged ones) and the previously best reported results [8, 14, 3, 4, 6].

$\sigma / \text{PSNR}$	FSP		TUP		FSP +	Block Size
	1LA	2LA	1LA	2LA	TUP	
<i>Lena</i>						
10/ 28.13	35.60	<b>35.66</b>	35.23	35.38	<b>35.71</b>	64
15/ 24.61	33.90	<b>33.96</b>	33.50	33.65	<b>34.01</b>	64
20/ 22.11	32.67	<b>32.71</b>	32.25	32.40	<b>32.77</b>	64
25/ 20.17	31.69	<b>31.72</b>	31.26	31.40	<b>31.78</b>	64
<i>Barbara</i>						
10/ 28.13	34.02	<b>34.60</b>	33.13	34.13	34.58	16
15/ 24.61	31.83	<b>32.49</b>	30.76	31.85	32.40	16
20/ 22.11	30.27	<b>30.94</b>	29.08	30.19	30.81	16
25/ 20.17	29.07	<b>29.76</b>	27.81	28.81	29.52	32
<i>House</i>						
10/ 28.13	35.29	35.44	35.32	<b>35.56</b>	<b>35.72</b>	16
15/ 24.61	33.54	33.64	33.73	<b>33.94</b>	<b>34.01</b>	32
20/ 22.11	32.30	32.34	32.54	<b>32.71</b>	<b>32.76</b>	32
25/ 20.17	31.32	31.30	31.59	<b>31.71</b>	<b>31.76</b>	32
<i>Peppers</i>						
10/ 28.13	33.74	33.81	34.06	<b>34.24</b>	<b>34.28</b>	16
15/ 24.61	31.7	31.72	32.02	<b>32.18</b>	<b>32.19</b>	16
20/ 22.11	30.27	30.24	30.57	<b>30.67</b>	<b>30.69</b>	16
25/ 20.17	29.17	29.10	29.45	<b>29.50</b>	<b>29.54</b>	16

**Table 1.** Denoising performance expressed as Peak Signal-to-Noise Ratio,  $10 \log_{10}(255^2/\sigma_e^2)$  in dB, where  $\sigma_e^2$  is the mean square error. First column shows the noise standard deviation and the corresponding PSNR. Next four columns correspond to FSP and TUP representations in the one-level adaptive (1LA) and two-level adaptive (2LA) version (highlighted the best of those 4 results). Next column shows the PSNR obtained using the average of the two 2LA results (highlighted if they are better than non-averaged results). Last column shows the block size used for the 2LA method.

## 6. REFERENCES

- [1] J S Lee, "Digital image enhancement and noise filtering by use of local statistics," *IEEE Pat. Anal. Mach. Intell.*, vol. PAMI-2, pp. 165–168, March 1980.
- [2] S G Chang, B Yu, and M Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," in *Fifth IEEE Int'l Conf on Image Proc*, Chicago, Oct 1998, vol. 1, pp. 535–539.
- [3] A Pižurica, W Philips, I Lemahieu, and M Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Proc.*, vol. 11, no. 5, pp. 545–557, May 2002.
- [4] L Şendur and I W Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Trans. Signal Proc.*, vol. 50, no. 11, pp. 2744–2756, Nov 2002.
- [5] M K Mihçak, I Kozintsev, K Ramchandran, and P Moulin, "Low-complexity image denoising based on statistical modeling of wavelet coefficients," *IEEE Trans. on Signal Processing*, vol. 6, no. 12, pp. 300–303, Dec 1999.



**Fig. 2.** PSNR output vs. PSNR input, in dB. We use TUP for *House* and FSP for *Barbara*. Our results, for  $\sigma_w = \{10, 15, 20, 25\}$ , are in Table 1. We use  $B = 128$  for *House* with  $\sigma_w = 50$ . We compare to three state-of-the-art methods. *House*: [14] (crosses) and [3] (diamonds). *Barbara*: [6] (crosses) and [4] (diamonds). For both images: [8] (circles) and this method (squares).

- [6] X Li and M T Orchard, "Spatially adaptive image denoising under overcomplete expansion," in *IEEE Int'l Conf on Image Proc*, Vancouver, Sept 2000, vol. 3, pp. 300–303.
- [7] J Portilla, V Strela, M Wainwright, and E Simoncelli, "Adaptive Wiener denoising using a Gaussian scale mixture model in the wavelet domain," in *Proc 8th IEEE Int'l Conf on Image Proc*, Thessaloniki, Greece, Oct 7-10 2001, pp. 37–40.
- [8] J Portilla, V Strela, M Wainwright, and E P Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Proc.*, vol. 12, pp. 1338–1351, Nov 2003.
- [9] S. Mallat, G. Papanicolaou, and Z. Zhang, "Adaptive covariance estimation of locally stationary processes," *Ann. Stat.*, vol. 26, no. 1, pp. 1–47, 1998.
- [10] J. Portilla, "Full blind denoising through noise covariance estimation using gaussian scale mixtures in the wavelet domain," in *IEEE Int'l Conf on Image Proc*, Singapore, Oct 2004, pp. 1217–1220.
- [11] J L Starck, D L Donoho, and E Candes, "Very high quality image restoration," in *Proc. SPIE conf. Signal and Image Processing*, San Diego, August 2001, vol. 4478, pp. 9–19.
- [12] M J Wainwright and E P Simoncelli, "Scale mixtures of Gaussians and the statistics of natural images," in *Adv. Neural Information Processing Systems*, S. A. Solla, T. K. Leen, and K.-R. Müller, Eds., Cambridge, MA, May 2000, vol. 12, pp. 855–861, Dec 1999.
- [13] D Andrews and C Mallows, "Scale mixtures of normal distributions," *J. Royal Stat. Soc.*, vol. 36, pp. 99–102, 1974.
- [14] M Malfait and D Roose, "Wavelet-based image denoising using a Markov random field a priori model," *IEEE Trans. Image Proc.*, vol. 6, pp. 549–565, April 1997.

## S.2.2. Image restoration using space-variant Gaussian scale mixtures in overcomplete Pyramids

- J. A. Guerrero-Colón, L. Mancera, J. Portilla: "Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids". IEEE Trans. Image Proc., vol. 17, no. 1, pp. 27-41, January 2008.



# Image Restoration Using Space-Variant Gaussian Scale Mixtures in Overcomplete Pyramids

Jose A. Guerrero-Colón, Luis Mancera, and Javier Portilla

**Abstract**—In recent years, Bayes least squares-Gaussian scale mixtures (BLS-GSM) has emerged as one of the most powerful methods for image restoration. Its strength relies on providing a simple and, yet, very effective local statistical description of oriented pyramid coefficient neighborhoods via a GSM vector. This can be viewed as a *fine* adaptation of the model to the signal variance at each scale, orientation, and spatial location. Here, we present an enhancement of the model by introducing a *coarser* adaptation level, where a larger neighborhood is used to estimate the local signal covariance within every subband. We formulate our model as a BLS estimator using space-variant GSM. The model can be also applied to image deconvolution, by first performing a global blur compensation, and then doing local adaptive denoising. We demonstrate through simulations that the proposed method, besides being model-based and noniterative, it is also robust and efficient. Its performance, measured visually and in L2-norm terms, is significantly higher than the original BLS-GSM method, both for denoising and deconvolution.

**Index Terms**—Bayesian estimation, Gaussian scale mixtures (GSM), image denoising, image restoration, overcomplete oriented pyramids.

## I. INTRODUCTION

**I**MAGE restoration is typically formulated as the estimation of an image given a linearly filtered version of the original corrupted by additive noise. This is a difficult, ill-posed, inverse problem, even if, as assumed in this work, the degradation parameters are known. A more tractable problem arises when we only consider additive noise in the degradation model (denoising problem). During the last few years, the state of the art in image denoising has experienced a formidable advance. The possibility of using image denoising techniques within more generic image restoration frames is, thus, becoming more and more attractive. In this paper, we present a new denoising-based

approximation to deal with both degradation problems (just noise, and blur plus noise). For previous instantiations of this work, see [1] and [2].

Wiener filtering is one of the simplest model-based restoration methods, and yet it provides optimal results when dealing with Gaussian signals and noise. Modeling noise as Gaussian, as we do in this work, is reasonable for many image processing situations. However, it is well known that image statistics are far from Gaussian [3]. One can, instead, assume that images are *locally* Gaussian. Spatially adaptive models have widely used this assumption for image denoising. In 1980, Lee [4] proposed an adaptive scalar Wiener method in the pixel domain, based on estimating the local variance at every spatial location of the image. With the introduction of *wavelets* (in the broad sense of the word, i.e., multiscale, multiorientation linear pyramidal representations), local adaptation in the new domain became much more powerful. Whereas many authors explored other estimations approaches in the new domain, which resulted in different types of shrinkage functions to be applied to the wavelet coefficients (e.g., [5]–[8]), others continued developing the local Wiener idea. Mihçak *et al.* [9] included a prior on the local variance, proposing a maximum *a posteriori* (MAP) estimation scheme. A similar Bayesian approach was followed in [10]. Portilla *et al.* [11] refined the Bayesian model by including the full signal and noise covariance in the model, and simplified the formulation of the problem by using a Gaussian scale mixture (GSM) [12], [13] frame [14]. These Wiener adaptive methods followed an empirical Bayes strategy: first, estimate the local signal variance, and then, apply a Wiener filter using that estimation. In a further work, Portilla *et al.* [15], instead, introduced a single-step Bayes least squares (BLS) estimator, which resulted in a significant performance increase. Up to now, all Wiener-adaptive methods either consider just the local variance, disregarding the covariance structure for each subband [9], [10] or consider the covariance fixed, up to a scale factor [11], [15]. In both cases, they perform a *fine* adaptation, in the sense that they estimate the local variance using small neighborhoods. On the other hand, there is broad literature on local covariance estimation and its applications (e.g., [16]). Covariance estimation allows for a selective characterization of salient local spectral features. However, as it requires a much larger window than variance estimation, it loses the ability to represent rapid variance spatial fluctuations in small areas. As a result, for typical images, the extra complexity of a covariance-adaptive model, by itself, does not pay off in performance terms with respect to simpler variance-adaptive models. Here, we describe a new approximation that provides both kinds of adaptation in the same model [1]. Hammond and Simoncelli [17] have also proposed an interesting extra adaptation step for the BLS-GSM model, in their case constrained to the local dominant orientation.

Manuscript received May 30, 2007; revised September 26, 2007. J. A. Guerrero-Colón was supported by the Ministerio de Educación y Ciencia (Spain) under Contract AP2005-2969 FPU and under Grant TEC2006/13845/TCM. L. Mancera was supported by the Ministerio de Educación y Ciencia (Spain) under Contract TIC2003-01504 FPI and under Grant TEC2006/13845/TCM. J. Portilla was supported by the Ministerio de Educación y Ciencia (Spain) under Grant TEC2006/13845/TCM grant and the Ramón y Cajal Program. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Minh. N. Do.

J. A. Guerrero-Colón and L. Mancera are with the Department of Computer Science and Artificial Intelligence, Universidad de Granada, Spain (e-mail: jaguerrero@decsai.ugr.es; mancera@decsai.ugr.es).

J. Portilla is with the Department of Images and Vision, Instituto de Óptica, CSIC, Madrid, Spain (e-mail: portilla@io.cfmac.csic.es).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIP.2007.911473

BLS-GSM [15] has provided the state-of-the-art in image Gaussian denoising for the last few years. During the last year, however, some new methods have achieved impressive results, both visually and in L2 norm terms, some of them clearly surpassing the BLS-GSM performance. Many exploit an important characteristic of typical image: they present a high spatial redundancy in the sense that a local feature is likely to be found in several (possibly disconnected) spatial locations within an image (e.g., a corner of a window, in a building façade, a straw bar, or, more simply, part of an edge along the edge line). An important part of the differences between these repeated local features may be caused by noise, so this fact can be exploited for noise removal. The usual approach is to consider that every neighborhood of pixels can be written as a linear combination of other neighborhoods. Buades *et al.* [18] do this matching for every pixel location, using the contextual similarities among neighborhoods around the pixels. Kervrann and Boulanger [19] improve this method by introducing an adaptive neighborhood as well as a statistical estimation of the model parameters. Hirakawa and Parks [20] use a total least squares formulation for square neighborhoods of different sizes. Dabov *et al.* [21] use block-matching and filter the matched blocks in 3-D. Once these blocks are 3-D filtered, they are denoised by a combination of hard-thresholding and local Wiener filtering. To the best of our knowledge these authors achieve the current state of the art in performance terms. However, their method is not based on a single model for the estimation, that is, it applies re-estimation. We refer here to *re-estimation* when the result of an estimation method is used as the input data for a *different* method with the *same* estimation purpose.<sup>1</sup> Though possibly effective and convenient in some practical situations, re-estimation is not a conceptually clean practice, as it implies the overlapping of two mutually incoherent models. Another successful path to image denoising is to adapt to the geometry of the image. Foi *et al.* [22] have presented a method based on a shape-adaptive discrete cosine transform (SA-DCT), which adapts the local support for the data being processed to the image contents. They use the same re-estimation method as in [21]. Elad *et al.* [23] achieve impressive results as well by creating an image-adaptive sparse representation. They used the K-SVD algorithm to obtain a dictionary that describes the image content very effectively. A big part of the high performance of this latter method is due to using a nonlinearly adaptive image representation, which requires a heavy computational load.

In contrast to the methods referred above, the one we propose here can use any linear multiresolution representation (though best results are obtained with multioriented translation invariant pyramids) and it does not make use of re-estimation. The underlying model is very simple, and the resulting method is noniterative, efficient and robust, inheriting and extending on the good properties of the original BLS-GSM, such as its ability to deal with any spectral noise density.

When approaching the deconvolution problem, we can see that most methods are iterative and global, e.g., [24]–[27]. Among the global methods, some authors (e.g., [28]–[33]) have approached the deblurring problem by decomposing it in two steps (see Fig. 1), a) a linear global compensation of the blur

<sup>1</sup>When both estimation methods are the same, we talk about iterative estimation, which is a conceptually clean practice (e.g., the E-M approach).

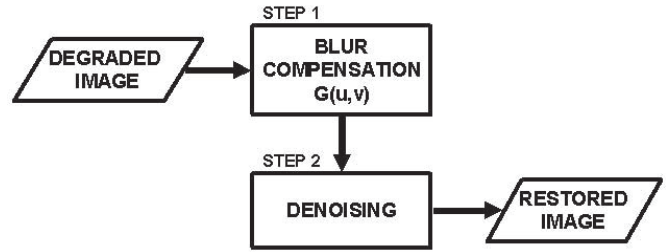


Fig. 1. Two-step scheme.

in the frequency domain and b) a nonlinear local denoising method. We emphasize that this procedure is not a form of re-estimation, because the problem addressed in the first step is different from the one addressed in the second step. The advantage of this decomposition is that it converts the deconvolution problem into an easier denoising problem, opening the door to recent powerful noniterative methods whenever they are able to represent and remove noise with arbitrary spectral density (PSD). Note that, by default, classical denoising assumes white noise. In contrast, here, even if the original noise is white, it gets colored by the blur compensation filtering in the first step.

Few authors instead, e.g., [34] and [35], have approached the deconvolution problem using local models, both for the image and for the degradation, with the advantage of simplifying the formulation of the problem. Portilla and Simoncelli [35] used a local GSM model within an efficient noniterative method which provided good performance in most cases. However, the local blur compensation used in this method has some intrinsic limitations (see Section III-B). Also, its performance depends critically on an accurate estimation of the power spectrum of the original image.

In this paper, we propose a noniterative two-step deblurring method based on the previously described spatially adaptive denoising. It provides a suboptimal, but very robust and efficient, choice for the prefiltering kernel. This approach preserves the advantages of using a local model for image denoising (simple and noniterative) whereas it overcomes the problems associated to locally compensate for the effects of convolution. It also increases both the performance and the robustness of its predecessors BLS-GSM restoration methods.

The outline of this paper is as follows. Section II describes our image statistical modeling, including a description of the image representations used. Section III explains how to use our model to do both denoising and deblurring. Implementation details are reported in Section IV, and Section V presents and discusses a broad set of simulations and their corresponding results. Section VI concludes the paper.

## II. IMAGE MODELING

### A. Image Representation

Image representation is a key issue for image processing, and its choice depends on the particular application. For image restoration, it has been widely demonstrated (see, e.g., [15], [36], and [37]) that translation invariant, overcomplete representations provide far better results than nonredundant representations, such as critically sampled wavelets. Among them, those based on a multiscale analysis of the image (usually in a

pyramid format), with orientation selectivity, typically provide the best results. Examples are the steerable pyramid [38], the curvelets [39], or the complex wavelet [40], among others. However, it has also been observed that each representation scheme fits best for processing certain image features, but none seems superior in absolute terms (see, e.g., [41]). Some authors have proposed to combine them by averaging their results [42], [1] although more powerful techniques are based on a nonlinear basis selection using a sparseness criterion [41]. A different approach is to use image content adaptive basis functions such as wedgelets [43], bandlets [44] or K-SVD overcomplete dictionaries [23], among others. In this work, we have used two representations exhibiting complementary features: the full steerable pyramid (FSP) [15] and a translation invariant version of the Haar wavelet [1] that we have termed translation invariant haar pyramid (TIHP).

1) *Full Steerable Pyramid*: The FSP [15] is an extension of Simoncelli and Freeman’s original steerable pyramid [38] that splits the original high-pass residual into the same number of orientations as the rest of the pyramid levels. Some good properties of the FSP are that a) their filters can be made highly selective in frequency and b) it is flexible in its number of orientations. Its drawbacks are coupled with these advantages: a) due to their high spectral selectivity, its filters have a large spatial support and b) due to the possibly very high number of subbands, it has a relatively high computational cost (best denoising results with BLS-GSM are reported for eight orientations [15], which results in a redundancy factor close to 19). According to these features, FSP is more suitable for processing texture-rich images.

2) *Translation Invariant Haar Pyramid*: The TIHP is an overcomplete version of the Haar wavelet [45]. It is translation invariant and it provides almost perfect reconstruction. For a detailed description, see the Appendix. In contrast to FSP, TIHP’s kernels are very small, providing excellent spatial localization. Its moderate redundancy factor (7) and reduced computational cost, make it fast and easy to operate with. TIHP’s main weaknesses are its poor oblique orientations selectivity and the low spectral localization of its kernels. These features made this representation suitable for images with little texture, most of the energy being concentrated along edges and lines (especially if oriented horizontally or vertically).

## B. Image Statistics and Gaussian Scale Mixtures

Image statistics were first described in terms of its spectral properties. Empirical studies showed that typical behavior followed a  $f^{\nu-2}$  law, being  $\nu$  a small correction factor [46], [47]. Thus, usual approximations, 50 years ago, were to consider images as samples from highly correlated Gaussian random fields. However, in 1989, Mallat [3] showed with empirical data that bandpass responses to typical images are far from Gaussian: its marginal probability density functions have a pronounced peak at zero, and heavy tails. This caused the sparse behavior of the wavelet coefficients: only a small proportion of them concentrates a high proportion of the total energy, and this occurs not only for the whole representation, but also for every subband. The histograms reflect high amplitude responses caused by most relevant localized image features (such as edges, lines,

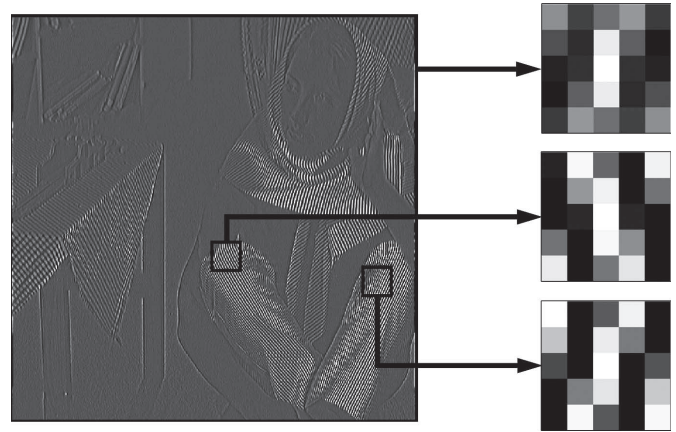


Fig. 2. Autocovariance variability within Barbara TIHP vertical band. We assume a  $3 \times 3$  neighborhood and compute each autocovariance patch using samples from the associated bounded region.

corners, etc.), whereas spatially dominant responses are typically small in amplitude, mainly corresponding to low-contrast texture within homogeneous patches in the scene. Therefore, the dominant source for non-Gaussianity of image statistics is the inhomogeneity of the image contents, which causes abrupt changes and localized high amplitude responses at the objects’ boundaries. In this way, it was definitely assessed the inadequacy of using global Gaussian distributions to model typical images. In 1997, Simoncelli [48] made another critical observation about image statistics: there is a strong coupling among neighbor *wavelet* coefficients in amplitude. This revealed explicitly a fundamental non-Gaussian joint statistical behavior that was coherent with the marginal densities studied so far. This feature has been exploited for image compression [49], [50] texture modeling and synthesis [51], and image restoration (all the predecessors of this work), among other applications. Two years later, Wainwright and Simoncelli found that Gaussian scale mixtures, GSM [12], was the proper mathematical tool to capture both the sparseness and the amplitude coupling of wavelet coefficients [14]. GSM’s have typically been used as a local description for neighbor coefficients in a pyramid (e.g., [15]), but they can also be used in a more powerful (but also more complex) global statistical description of the image in the transformed domain [52].

The GSM model in the wavelet domain [14] is able to represent the observed strong local variance fluctuations in neighborhoods of coefficients at the different spatial locations, scales and orientations. This spatial inhomogeneity of the local variance explains, on its own, the two basic nonlinear statistical patterns mentioned above, namely, sparseness of the responses and amplitude coupling among neighbors. The original GSM model assumed a fixed normalized covariance for all the neighborhoods at a given subband. However, it is clear that typical images have different spatial correlation patterns at different locations in a subband, even if the filtering associated with each subband greatly reduces the spatial variability of the local covariance in the image domain. We illustrate this in Fig. 2, where, using a TIHP vertical subband of *Barbara*, we have compared the local auto-covariance samples estimated for the whole subband (assuming a  $3 \times 3$  neighborhood) with that computed from two different regions, both having high-contrast features. In this



case, the corresponding auto-covariance samples reveal significant differences in local orientation. Note that the auto-covariance of the whole subband reflects the average behavior, reducing the potential of a more spatially selective signal/noise discrimination. Thus, to further increase the realism and performance of our model, it is justified to use a Gaussian scale mixture with a spatially adaptive covariance for the local signal, that is, a space-variant GSM.

Let us consider an image decomposed into oriented subbands at multiple scales. We denote as  $x_c^j(n, m)$  the coefficient corresponding to the subband  $j$ , centered at spatial location  $(n, m)$ . We denote as  $\mathbf{x}^j(n, m)$  a *neighborhood* of coefficients clustered around this *reference coefficient* (see neighborhood structure in Section IV). We model these neighborhoods as Gaussian scale mixtures (GSMs). A random vector  $\mathbf{x}$  is a GSM [12] if it can be expressed as the product of two independent random variables

$$\mathbf{x} = \sqrt{z}\mathbf{u}$$

where  $z$  is a positive scalar and  $\mathbf{u}$  is a zero-mean Gaussian vector. We model the desired space variant behavior by defining  $\mathbf{C}_u^{j,(n,m)}$  as the (space variant) covariance of  $\mathbf{u}^j(n, m)$ . The density of  $\mathbf{x}$  is determined uniquely by  $p_z(z)$  and  $\mathbf{C}_u^{j,(n,m)}$  as follows:

$$\begin{aligned} p^{j,(n,m)}(\mathbf{x}) &= \int p^{j,(n,m)}(\mathbf{x}|z)p_z(z)dz \\ &= \int \frac{\exp\left(-\mathbf{x}^T \left(z\mathbf{C}_u^{j,(n,m)}\right)^{-1} \mathbf{x}/2\right)}{(2\pi)^{N/2} \left|z\mathbf{C}_u^{j,(n,m)}\right|^{1/2}} p_z(z)dz \end{aligned}$$

where  $N$  is the dimensionality of  $\mathbf{x}$  and  $\mathbf{u}$  (the size of the neighborhood). Note that  $\mathbf{x}^j(n, m)$  is conditionally Gaussian for a given  $z$ . There are different possibilities for the mixing density  $p_z(z)$ , which are described in [15]. As in the latter work, we have assumed it constant in  $\log z$  for all the subbands in the working range, which corresponds to a noninformative Jeffrey's prior.

### III. IMAGE RESTORATION

#### A. Denoising Using Space-Variant GSMs

We aim to solve the classical denoising problem, where an image is corrupted by additive zero-mean independent Gaussian noise of known (but arbitrary) spectral density. As many other methods, we perform the denoising in the pyramid domain and obtain the image estimation by reconstructing the image from the estimated coefficients. The observation model is, in this case

$$\mathbf{y} = \mathbf{x} + \mathbf{w}$$

where  $\mathbf{y}$ ,  $\mathbf{x}$ , and  $\mathbf{w}$  are vectors (in a lexicographic order) of observed, original, and noise coefficients, respectively.

1) *Two-Level Adaptive Denoising*: Note that the model in [15] used a fixed signal covariance at each subband, providing only (fine) adaptation to the local variance (through the GSM hidden multiplier). Here we include another level of (coarse) adaptation by also letting the signal covariance to be a function of the spatial position at each subband. Following the GSM

model presented in Section II-B, we can express a neighborhood of  $N$  noisy coefficients as<sup>2</sup>

$$\mathbf{y} = \sqrt{z}\mathbf{u} + \mathbf{w}$$

where  $\mathbf{w}$  is the zero mean Gaussian noise vector with covariance matrix  $\mathbf{C}_w$ . Note that  $\mathbf{C}_w$  (unlike the signal covariance  $\mathbf{C}_u$ ) is assumed constant for all neighborhoods within a subband (but different for each subband). By GSM definition  $z$  and  $\mathbf{u}$  are independent, and by assumption  $\mathbf{w}$  is independent, too. Hence, the density of the observed neighborhood vector conditioned on  $z$  is Gaussian (zero-mean), with covariance  $\mathbf{C}_{y|z} = z\mathbf{C}_u + \mathbf{C}_w$

$$p(\mathbf{y}|z) = \frac{\exp\left(-\mathbf{y}^T (z\mathbf{C}_u + \mathbf{C}_w)^{-1} \mathbf{y}/2\right)}{((2\pi)^N |z\mathbf{C}_u + \mathbf{C}_w|)^{1/2}}. \quad (1)$$

2) *Parameter Estimation*: Given (1), signal and noise covariances estimations are needed. The neighborhood noise covariance,  $\mathbf{C}_w$ , is computed at each subband by applying the pyramidal representation to a deterministic function with the same sample autocorrelation as the noise (for instance, a scaled delta function for white noise). The elements of  $\mathbf{C}_w$  are computed by averaging the products of pairs of coefficients over all the neighborhoods of the subband. Note that, unlike noise, this deterministic function yields a result that is free from random fluctuations.

The neighborhood observation covariance,  $\mathbf{C}_y$ , can be computed from  $\mathbf{C}_{y|z}$  by taking the expectation over  $z$

$$\mathbf{C}_y = \mathbb{E}\{\mathbf{C}_{y|z}\} = \mathbb{E}\{z\}\mathbf{C}_u + \mathbf{C}_w.$$

We can set, without loss of generality,  $\mathbb{E}\{z\} = 1$  obtaining  $\mathbf{C}_y = \mathbf{C}_u + \mathbf{C}_w$ . To estimate the spatially variant signal covariance  $\mathbf{C}_u$  we assume that it changes smoothly over the image, and, thus, that it may be sampled by estimating it into a set of nonoverlapping regions.<sup>3</sup> We divide each subband into  $B \times B$  blocks, which are assumed uniform in covariance. For each of this blocks we estimate  $\mathbf{C}_u$  by subtracting  $\mathbf{C}_w$  from the sample covariance of its coefficients,  $\mathbf{C}_y$ .

3) *Coefficient Estimation*: For each neighborhood  $\mathbf{y}$  of observed coefficients, we estimate  $x_c$ , the reference coefficient, at the center of the neighborhood. The Bayes least squares (BLS) estimate can be expressed in this case [15] as

$$\mathbb{E}\{x_c|\mathbf{y}\} = \int_0^\infty p(z|\mathbf{y}) [\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\}]_{x_c} dz.$$

Thus, the solution is written as the average of the Bayes least squares estimate of  $x_c$  when conditioned on  $z$ , weighted by the posterior density of  $z$ ,  $p(z|\mathbf{y})$ . The advantage of the GSM model is that the coefficient neighborhood vector  $\mathbf{x}$  is Gaussian when conditioned on  $z$ . This fact, coupled with the assumption of ad-

<sup>2</sup>For notational simplicity from here on we drop the superscript  $j$  and the indices  $(n, m)$ . Note that the following equations are the same as in [15], but now  $\mathbf{C}_u$  represents a local covariance, as it depends on the spatial position.

<sup>3</sup>We have empirically tested that the negligible performance improvement achieved by overlapping the blocks does not pay off the large increase of computation.

ditive Gaussian noise means that the expected value inside the integral of (2) is simply a Wiener estimate

$$\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\} = z\mathbf{C}_u(z\mathbf{C}_u + \mathbf{C}_w)^{-1}\mathbf{y}.$$

We can simplify the dependence of this expression on  $z$  by diagonalizing the matrix  $z\mathbf{C}_u + \mathbf{C}_w$  (see [15]). Note that, unlike in [15], now this diagonalization depends on the spatial location (as  $\mathbf{C}_u$  is a function of  $(n, m)$ ), and, thus, it needs to be computed once per each considered spatial location of each subband.

### B. Deblurring-by-Denoising Using Space-Variant GSMs

We face the problem of image restoration, where prior to noise addition, the signal has been degraded by (typically) a low-pass filter. The observation is modeled as follows:

$$y(n, m) = h(n, m) * x(n, m) + w_0(n, m). \quad (2)$$

where  $h$  is a convolution kernel (assumed known) applied to the original image  $x$ , usually called Point Spread Function, PSF.  $w_0$  is Gaussian noise of known (but arbitrary) power spectral density. The same BLS-GSM local approach used in [15] for image denoising was used in [35] for image restoration. However, in this case, the fact of having a local description of image and degradation caused problems when trying to compensate for certain convolution kernels. Let us analyze this problem. Given a small neighborhood for the local model, we can only aspire to compensate for smoothly varying filters in the frequency domain having no zeros (so their regularized inverse filters will also be smooth in frequency, and, as a consequence, small in the image domain). A fixed absolute neighborhood size, thus, would very severely limit the deconvolution power of the method. A better option is to use a fixed neighborhood size for all the scales in a multiresolution representation. In such a way, we are able to represent larger effective neighborhoods as we go up in the pyramid. We explain this with a 1-D example in Fig 3.

We show the Fourier representation, in magnitude, of a spatially limited signal, which corresponds to the same neighborhood size, measured in number of coefficients, at different scales. As we go up in the scale, the frequency support becomes smaller, and, thus, more rapidly varying responses are obtained. So, thanks to multiresolution, we can use (and compensate for, whenever there are no zeros) smooth filters at high frequencies, less smooth at medium frequencies, and rapidly varying at low frequencies. Such constraint is compatible with many low-pass kernels (e.g., Laplacian, Gaussian, etc., see bottom left of Fig. 3). However, there are some practically important kernels, such as those integrating uniformly in their spatial supports, whose behavior in frequency is not flat at all at high frequencies, but present strong oscillations around zero. Trying to compensate for those zeros requires abrupt responses for the regularized inverse filtering, which implies necessarily a large spatial support (see bottom right of Fig. 3). Therefore, uniform blurring cannot be properly treated by using this kind of local multiresolution schemes. Another drawback of the referred method [35] is that, to work properly, it requires a reasonably accurate estimation of the cross covariance among the original and the filtered image. This implies, in turn, the need of estimating the power spectrum of the original image ( $P_x$ ) from the degraded observation, which is a difficult problem. Using

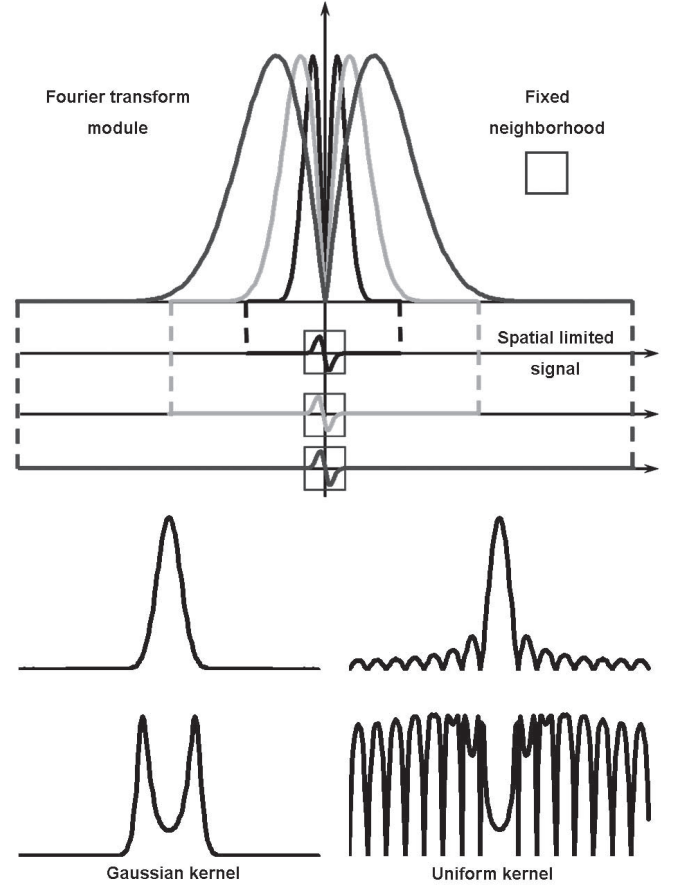


Fig. 3. Illustration of the restrictions on the shape in the frequency domain of the filters that we can represent using a local approach (fixed small neighborhood) with multiresolution. The same spatial limited signal is shown at three different scales. This scheme allows us to represent (and to compensate for) slowly varying filter responses at high frequencies, with no zeros, and more rapidly varying frequency responses at low frequencies. At the bottom, two typical 1-D low-pass filters are shown for comparison, together with a regularized version of their inverse filters. We can see that the Gaussian kernel can be well compensated for using this kind of local multiresolution approach, whereas the uniform kernel cannot.

a general model for  $P_x$  translates in this case into a significant performance fall.

1) *Deblurring-by-Denoising*: In order to keep the advantages of using a local model, while overcoming its drawbacks, we use here the two-step scheme depicted in Fig. 1. This deconvolution strategy was proposed more than ten years ago [28], and it has been followed by many authors [29]–[33]. The two steps are: a) blur compensation through global linear filtering, and b) local denoising of the blur-compensated observation.

We can express the observation model of (2) in the frequency domain as

$$Y_0(u, v) = H(u, v)X(u, v) + W_0(u, v)$$

where<sup>4</sup>  $H$  is the Fourier transform of the blur kernel  $h$ . Let  $G$  be the filter applied in step 1 (here in after, prefilter). In order to compute the optimal prefilter in MSE sense,  $G_{\text{opt}}$ , we must couple both steps

$$G_{\text{opt}} = \arg \min_G \left\{ \mathbb{E} \left\{ \|X - f(GY_0)\|^2 \right\} \right\} \quad (3)$$

<sup>4</sup>For notational simplicity from here on we drop the frequency indices  $(u, v)$ .

where  $f$  represents here the denoising operator (step 2), so  $f(GY_0)$  represents the output of the whole process. The expectation inside (3) can be expressed as

$$\begin{aligned} & \mathbb{E} \left\{ \|X - f(GY_0)\|^2 \right\} \\ &= \mathbb{E} \left\{ \|(X - GHX) + (GHX - f(GY_0))\|^2 \right\} \\ &\simeq \mathbb{E} \left\{ \|(X - GHX)\|^2 \right\} + \mathbb{E} \left\{ \|(GHX - f(GY_0))\|^2 \right\} \end{aligned}$$

where the r.h.s. of the latter approximation represents the sum of the quadratic errors coming from the deblurring and the denoising steps. This assumes that both errors are approximately decorrelated, which is true, at least under usual conditions.<sup>5</sup> Thus, to compute the optimal prefilter  $G$  we need an explicit error model for the denoising step. To obtain such an error model for a nonlinear adaptive denoising method is a major challenge, and, to the best of our knowledge, no one has proposed such a model to this date. In the absence of a globally optimal solution, next we follow a marginally optimal procedure which is, nevertheless, efficient and robust.

*Step 1: Global Blur Compensation:* This step provides a blur compensated image by applying a global filtering to the observed image. Instead of estimating an optimal filter we reduce the degrees of freedom from  $I_x \times I_y$  (image and filter size) to only one, corresponding to the regularization parameter,  $\alpha$ . We use, as in other works [29], [30], [33], a generalized Wiener filter

$$G = \frac{H^*}{|H|^2 + \alpha \cdot \left( \frac{P_{w_0}}{P_x} \right)}$$

where  $P_{w_0}$  and  $P_x$  are the power spectral densities of noise and signal respectively. This regularized inversion compensates for the filter phase response. The balance between noise and blur suppression is controlled by the regularization parameter  $\alpha$ . Low values yield in noisy estimations ( $\alpha = 0$  is just an inverse filter<sup>6</sup>) whereas high values tend to suppress too much signal. The higher is  $\alpha$ , the higher is the overall linear distortion, but the lower it is, the noisier is the estimation.  $\alpha = 1$  yields the Wiener filter, which is a far too high value when using typical images and nonlinear denoising methods in the second step, as we show in Section III-B2.

*Step 2: Local Denoising:* After step-1, the observation yields

$$Y_1 = GY_0 = GHX + GW_0 = H_r X + W_1.$$

Because  $G$  has been optimized in the first step, no further attempt to correct the remaining linear distortion  $H_r = GH$  is made, and what is left is to denoise the prefiltered image.

2) *Prefiltering Parameter Estimation:* Based on the typical images power spectrum [46], [47], we have used a simple model for  $P_x$ , namely  $\hat{P}_x = k/f^2$ , where  $f$  represents absolute frequency and  $k$  an arbitrary constant. In order to force the model

<sup>5</sup>We have empirically tested this decorrelation through simulations.

<sup>6</sup>We point out that it is critical for the well behavior of the filter  $G$  and for the final performance of the method to ensure that  $P_{w_0}$  is strictly positive on the whole frequency plane. This is a practical concern when  $P_{w_0}$  is estimated from real images.

to be consistent with the estimated original variance ( $\hat{\sigma}_x^2 = \sigma_{y_0}^2 - \sigma_{w_0}^2$ ), the following expression must hold:

$$\hat{\sigma}_x^2 = \int_{f>0} |H(f)|^2 \hat{P}_x f df = \int_{f>0} |H(f)|^2 \frac{k}{f} df.$$

Hence,  $k$  is set to

$$k = \hat{\sigma}_x^2 / \int_{f>0} |H(f)|^2 / f df.$$

Despite its simplicity, this model is effective and, in contrast to other methods (such as [35]), the associated  $P_x$  estimation error does not seem to have a critical impact on the overall performance.

Unlike other authors [28], [30] that used a model to choose the regularization parameter  $\alpha$ , we have computed it empirically, training with a set of standard test images (*House*, *Cameraman*, *Barbara*, *Boat*, and *Lena*), using Gaussian convolution kernels and white Gaussian noise of 0.4 to 3.2 and 1 to 16 standard deviations, respectively (in a geometric scale). For each combination of degradation parameters, we have computed the optimal  $\alpha$  through golden search. We also computed a quality decrease tolerance interval around the optimal value, according to a performance decrease of 0.05 dB with respect to the maximum. Finally, we chose the common  $\alpha$  value providing the highest proportion of experiments within the quality tolerance intervals. Fig. 4 shows the optimality ratio for each alpha value, showing a peak for  $\alpha = 0.3$ , for which value around 84% of the experiments provide quasi-optimal results. We have also tested the method with very different PSFs [2] (such as a uniform kernel), and the maximum of the histogram was still very close to 0.3, although the percentage dropped slightly (73% for the uniform kernel). We have also tested that the results are fairly independent on the particular images chosen for the training, by comparing these results with those obtained by excluding each time the image to be processed in the training set.

## IV. IMPLEMENTATION

### A. Boundary and Block Handling

As demonstrated in [15], careful boundary handling is crucial for achieving a high denoising performance. In this case, it is even more critical than for [15], because of the image partition into blocks. To illustrate this, let us consider a  $L \times L$  GSM neighborhood in a  $N_s$ -scale pyramid. In the upper part of Fig. 5, we can see the corner of a subband at the the highest scale in the pyramid ( $L = 3$  and  $N_s = 4$  in this example). All GSM neighborhoods with reference coefficients at the boundaries of the subband are incomplete, lacking those coefficients marked by the dashed lines. To solve this, the observation is extended, by mirror reflection with no repetition,  $2^{(N_s-1)}(L-1)/2$  pixels at each side, which yields the required extra coefficients for the subbands at the top of the pyramid (see dashed square in Fig. 5). When dealing with block-based methods, the boundary artifacts are potentially stronger, because of the larger proportion of boundary coefficients compared to the whole image. The

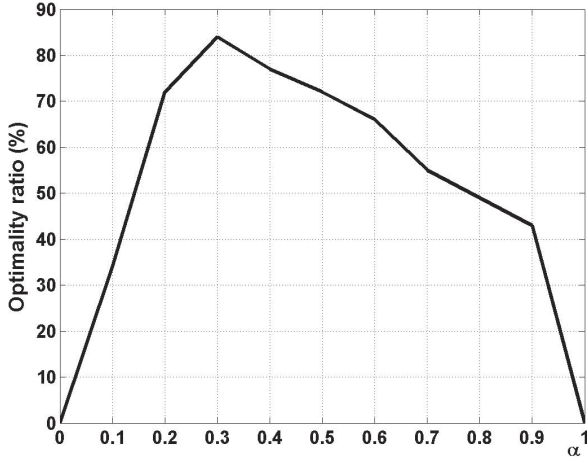


Fig. 4. Optimality ratio of the regularization parameter  $\alpha$ . See text for details.

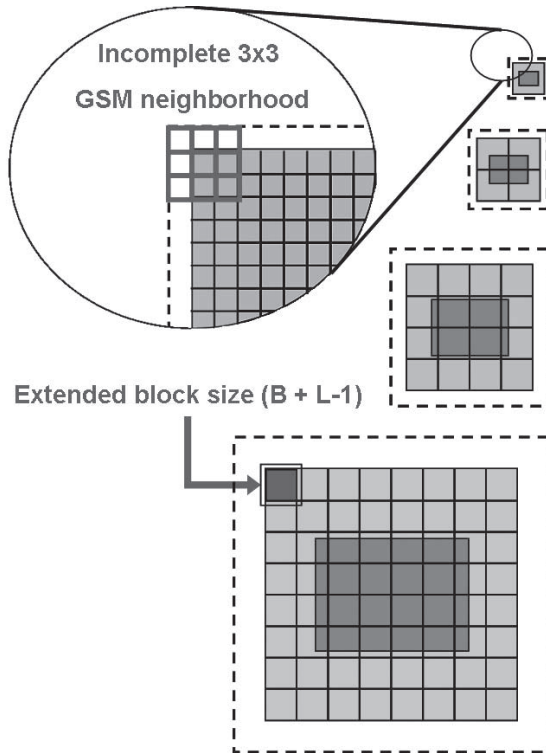


Fig. 5. Multiresolution block fitting.

same as before, each block must be extended  $(L - 1)/2$  coefficients at each side. The inner blocks are enlarged with true signal, whereas boundary blocks use the mirror extension made for the whole subband. This produces a small overlapping between extended blocks. We have tested that increasing this overlapping does not translate into any significant improvement.

Another practical issue is the block arrangement within subbands. In order to have complete blocks in all subbands, we should ensure that all the subbands have dimensions that are integer multiples of the block size, i.e.,  $\mathbf{I}_{x,y}/(2^s B)$  is an integer vector for  $s = 0 \dots N_s - 1$ , being  $\mathbf{I}_{x,y}$  the image dimension vector. As original image dimensions will not comply, in general, with this requirement (see dark shaded area in Fig. 5), they

will generally require to be extended (by mirror reflection) to also fit this constraint (see light shaded areas in Fig. 5). In practice, we apply a relaxed version of this criterion, processing the whole subbands when they are smaller than the chosen block size, so effectively reducing the range of  $s$  for imposing this constraint.

### B. Denoising Experiments

We have used the TIHP and the FSP as image representations. We chose the number of scales to 4 and 5 for  $256 \times 256$  and  $512 \times 512$  images, respectively. Note that the number of orientations is fixed for the TIHP (3), whereas we used eight orientations for the FSP (as in [15]). We used a  $3 \times 3$  GSM neighborhood around each coefficient (same as in [15]). In the case of FSP, we also include a coefficient at the same location and orientation at the next coarser scale (a *parent*). Different block sizes  $B$  were tested for estimating the signal covariance ( $B = 16, 32, 64, 128, F$ , where  $F$  means the whole image). Both the GSM neighborhood and the block size are the same in all pyramid levels, except for the subbands at the top of the pyramid, which have no parent, and those subbands with a size smaller than the block.

### C. Deblurring Experiments

We have implemented the global prefiltering (step 1) in the Fourier domain, using  $\alpha = 0.3$ . For the denoising (step 2), we have selected manually TIHP or FSP based on the relative amount of texture in the image: FSP was chosen for texture-rich and TIHP for poorly textured images. The number of orientations were kept the same as for denoising whereas the chosen number of scales was 4 for all images sizes. The GSM neighborhood was also  $3 \times 3$ , but with no parent now in either representations. The block size was fixed to  $32 \times 32$ .

### D. Computational Cost

As a guide, Table I shows the computation time (in seconds) of our unoptimized denoising and deblurring *Matlab 7.0* code, on *Windows XP* over a 3.0-GHz *Intel Pentium IV* CPU with 1-Gb RAM, for different representations, blocks, and images sizes. For comparison, we also include the computation time of the original BLS-GSM at the bottom part of the table. We extract three basic patterns: a) the cost scales approximately inversely with  $B$ , roughly a factor of 3 when we halve the block; b) the cost of the first step in the deblurring algorithm is negligible when compared to the cost of the second step; c) the computation time is significantly higher, but still comparable to that of original BLS-GSM. The main difference (in cost terms) between the original BLS-GSM and the proposed method is the increase of calls to functions. The proposed method usually makes one call per *block*, whereas the original BLS-GSM makes only one per *subband*. Note also that the number of calls are multiplied by a factor of 4 when we halve the block size. This has a big impact due to using unoptimized interpret code in our current implementation. Finally, the cost scales roughly with a factor of 8 when we double the image size.

## V. RESULTS AND DISCUSSION

We have used five standard test images to evaluate our algorithms: *Barbara*, *Lena*, *House*, *Cameraman*, and *Peppers*. We

TABLE I  
AVERAGED COMPUTATION TIME IN SECONDS (TEN TRIALS) COMPARING DIFFERENT IMAGES SIZES, BLOCK SIZES, AND REPRESENTATIONS FOR BOTH METHODS. FOUR SCALES ARE USED

Representation	FSP (8 Orientations)		TIHP (3 Orientations)	
Image size $\Rightarrow$	256 $\times$ 256	512 $\times$ 512	256 $\times$ 256	512 $\times$ 512
<i>Algorithm</i>	16 $\times$ 16			
Denoising	108.51	1090.07	46.18	451.29
Deblurring	107.00	1091.39	45.39	447.77
<i>Algorithm</i>	32 $\times$ 32			
Denoising	39.85	312.55	16.40	132.25
Deblurring	39.13	312.48	16.40	133.20
<i>Algorithm</i>	64 $\times$ 64			
Denoising	22.65	119.36	8.76	52.19
Deblurring	22.49	122.26	9.22	54.27
<i>Algorithm</i>	Original BLS-GSM			
Denoising	16.14	62.92	7.22	32.44
Deblurring	17.26	61.11	7.50	29.66

have chosen *Barbara* and *House* for all visual comparisons, as two examples with very different amount of texture.

#### A. Denoising

We selected four standard images: *Barbara*, *Lena* (both 512  $\times$  512), *House* and *Peppers* (both 256  $\times$  256). We have degraded them by adding additive white Gaussian noise of different standard deviations ( $\sigma = 5, 10, 15, 20, 25, 50$ ).

1) *Model Features*: In Table II, we present the results of our proposed algorithm, in terms of peak signal-to-noise ratio (PSNR),  $10 \log_{10}(255^2/\sigma_e^2)$  in decibels, where  $\sigma_e^2$  is the mean square error. We include results for the representations used, TIHP and FSP, and compare our method (column labeled two-level Adaptive, 2LA) with the standard version of BLS-GSM [15] (labeled one-level Adaptive, 1LA). We also include the block size,  $B$ , for which the method provided the best results. In Fig. 6, we show some visual results (cropped and zoomed, for artifacts visibility) on *Barbara* and *House* images. We include results for TIHP and FSP (second and third rows) and a comparison with standard BLS-GSM (by columns). We now present a discussion along the followings directions:

*Comparison to Standard BLS-GSM*: The enhancement of the model proposed in Section II-B provides us a general improvement of the results over standard BLS-GSM method, both visually and in MSE sense. Comparing the columns 1LA-2LA of Table II, we can see a significant performance increase. The new results are better (with no exception) for both representation TIHP and FSP. As we can see, the improvement is more important for images having local spectral peaks (like *Barbara*), whereas for other images (e.g., *Lena*), the improvement is more modest. This behavior is related to the amount of covariance variability among different regions in the image (see Fig. 2 for details). We can compare visually the differences between our method and the standard BLS-GSM, for *Barbara* and *House* images, in the columns of Fig. 6. Note how artifacts on the roof line of *House* disappear with the new method (second column). For *Barbara* we can see a very strong improvement, the new results having much less artifacts, and much more texture is recovered.

*Representation comparison: FSP Versus TIHP*: As we can see in Table II, the results confirm what we pointed out in Section II-A: Different representations fit better different images,

TABLE II  
DENOISING PERFORMANCE EXPRESSED AS PSNR IN DECIBELS. FIRST COLUMN SHOWS THE NOISE STANDARD DEVIATION AND THE CORRESPONDING PSNR. NEXT FOUR COLUMNS CORRESPOND TO FSP AND TIHP REPRESENTATIONS IN THE ONE-LEVEL ADAPTIVE (1LA) AND TWO-LEVEL ADAPTIVE (2LA) VERSION (HIGHLIGHTED THE BEST OF THOSE FOUR RESULTS). LAST COLUMN SHOWS THE BLOCK SIZE USED FOR THE 2LA METHOD, WHERE  $F$  MEANS THE FULL IMAGE

$\sigma_w$ / PSNR	FSP		TIHP		Block Size
	1LA	2LA	1LA	2LA	
<i>Lena</i>					
5/ 34.15	38.49	<b>38.55</b>	38.19	38.34	32
10/ 28.13	35.60	<b>35.66</b>	35.23	35.38	64
15/ 24.61	33.90	<b>33.96</b>	33.50	33.65	64
20/ 22.11	32.67	<b>32.71</b>	32.25	32.40	64
25/ 20.17	31.69	<b>31.72</b>	31.26	31.40	64
50/ 14.15	<b>28.61</b>	<b>28.61</b>	28.21	28.21	F
<i>Barbara</i>					
5/ 34.15	37.78	<b>38.19</b>	37.19	37.84	16
10/ 28.13	34.02	<b>34.60</b>	33.13	34.13	16
15/ 24.61	31.83	<b>32.49</b>	30.76	31.85	16
20/ 22.11	30.27	<b>30.94</b>	29.08	30.19	16
25/ 20.17	29.07	<b>29.76</b>	27.81	28.81	32
50/ 14.15	25.42	<b>26.04</b>	24.33	25.03	32
<i>House</i>					
5/ 34.15	38.65	<b>38.88</b>	38.23	38.55	16
10/ 28.13	35.29	35.44	35.32	<b>35.56</b>	16
15/ 24.61	33.54	33.64	33.73	<b>33.94</b>	32
20/ 22.11	32.30	32.34	32.54	<b>32.71</b>	32
25/ 20.17	31.32	31.30	31.59	<b>31.71</b>	32
50/ 14.15	28.22	28.18	28.36	<b>28.38</b>	128
<i>Peppers</i>					
5/ 34.15	37.30	37.43	37.57	<b>37.72</b>	16
10/ 28.13	33.74	33.81	34.06	<b>34.24</b>	16
15/ 24.61	31.7	31.72	32.02	<b>32.18</b>	16
20/ 22.11	30.27	30.24	30.57	<b>30.67</b>	16
25/ 20.17	29.17	29.10	29.45	<b>29.50</b>	16
50/ 14.15	25.94	25.94	<b>26.11</b>	<b>26.11</b>	F

depending on their contents. *House* and *Peppers*, two poorly textured images, are best denoised using the TIHP, whereas for *Lena* and *Barbara*, the best results are achieved with FSP. We can also see the significant difference in visual terms, by inspecting *House* and *Barbara* images in the second (FSP) and third (TIHP) rows of Fig. 6. In the case of *House*, the visual difference is significant (less artifacts in general), but not dramatic. However, in the case of *Barbara* the difference is striking: FSP provides a much better result than the one obtained with TIHP. If one of them should be chosen basing only on results (ignoring computational cost), under a minimax risk criterion, FSP would be the choice.

Another issue is the different relative gains between the new method when using TIHP versus FSP, with respect to the standard BLS-GSM. By comparing in Table II the performance difference between our method and the original BLS-GSM for each image and noise level (by rows), we see that the improvement is bigger when we use the TIHP. The main reason is that whereas FSP kernels are highly selective in frequency, TIHP kernels are not, and, hence, there is more spatial variability of the covariance inside each subband in the latter representation. This increases the difference between local covariances and the global one (see details in Section II-B). We have also tested the benefit of averaging FSP and TIHP results (in the same way as

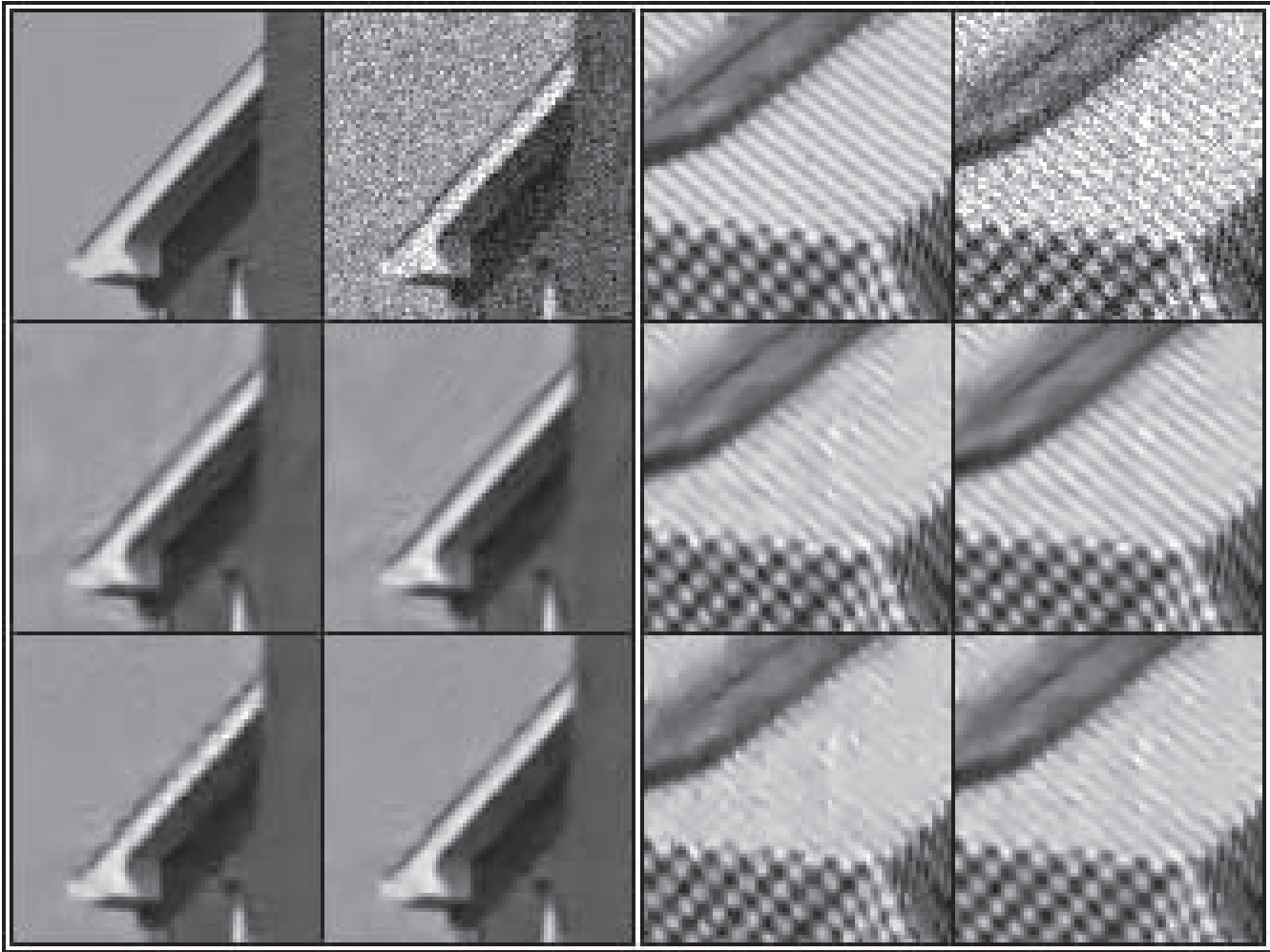


Fig. 6. Visual comparison of denoising results on *House* and *Barbara* images cropped to  $80 \times 80$  pixels (starting in pixel [1,78] and [1,221], respectively, [column, row]). We used  $32 \times 32$  and  $16 \times 16$  block sizes for *House* and *Barbara*, respectively. From left to right and top to bottom (House/Barbara PSNR in decibels): original; noisy ( $\sigma_w = 20$ ), (22.11/22.11); BLS-GSM [15] with FSP (32.30/30.27); our method with FSP (32.34/30.94); BLS-GSM [15] with TIHP (32.54/29.08); our method with TIHP (32.71/30.19).

other authors [42]), obtaining an average increase of 0.15 dB over the best one.

**Block Size Considerations:** The optimal block size depends on the image contents in the first place, but also on the noise level. The lower the SNR, the larger the optimal size. This makes sense, as local signal covariance estimation stops being reliable for small block sizes in the presence of high noise, so covariance spatial adaptation no longer pays off. In Fig. 7, we show, for *Barbara* and *House*, a plot of PSNR (with range adjusted between 0 and 1) when using  $B = 16, 32$ , and 128 versus the noise standard deviation. Here one corresponds to the best result obtained with the set of block sizes tested, and zero to the worst result. We can state, as expected, that the best result for low noise levels are achieved with the smallest size block ( $B = 16$ ), whereas  $B = 128$  provides better performance for high noise variance. Overall, it seems that a finer adaptation ( $B = 16$ ) provides better results than coarser covariance adaptations. However, differences are, in general, small, and, from a computational cost view, the smaller is the block the higher becomes CPU time. As we can see in Fig. 7, the curve associated to a block size of 32 is a good tradeoff in practice, providing best results for mod-

erate noise levels and not dramatically decreasing when in the extremes.

**2) Comparison to State-of-the-Art Methods:** In Fig. 8, we show a visual comparison of our method and, what to the best of our knowledge, are the three ones with highest PSNR performance available in the literature [21]–[23]. For *House*, our method behaves the worst in PSNR terms, mainly because the other methods recover the wall texture and ours not. Its visual appearance is, nevertheless, convincing. For instance, the roof line is sharper and less contaminated by artifacts in our result than in [22]. For *Barbara*, our method behaves similarly to the other three, both visually and in MSE terms. Fig. 9 shows, also for *Barbara* and *House*, a graphical comparison, in terms of input–output PSNR, among our method and the top three reporting the best results [21]–[23]. Our method is represented by a solid line, whereas the standard BLS-GSM is included as a reference using a dash line. As we can see, our method behaves competitively to current state-of-the-art [21], especially considering its moderate computational cost. We also refer the reader to [53] for a recent perceptual comparison of state-of-the-art denoising methods

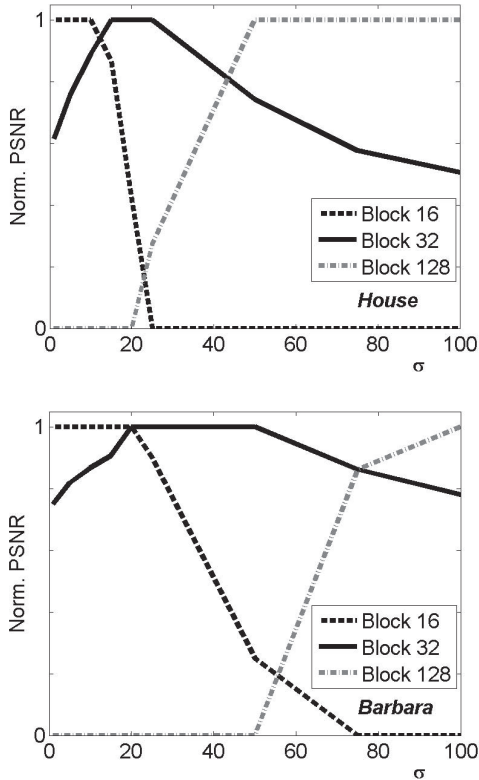


Fig. 7. PSNR versus noise standard deviation for *House* and *Barbara* images (adjusted between 0 and 1). Dashed black line  $16 \times 16$  block, solid line for  $32 \times 32$  block and dot-dashed gray one for  $128 \times 128$  one. Note that the “optimal” block size increases with the noise level.

## B. Deblurring

In Table III, we show averaged Increment in signal-to-noise ratio (ISNR) results of applying our deblurring method to the training data set (see Section III-B2 for details). As we can see, the results exhibit a significant improvement for the whole range of tested blur and noise degradations. In Table IV, we show the results for six deconvolutions experiments with *House*, *Cameraman* (using THP) and *Barbara* (using FSP) comparing our method with the previous BLS-GSM restoration [35], and with those which, to the best of our knowledge, represent the current state-of-the-art [24], [33]. In the first two columns, we have replicated the experiments from [32]. The kernel used (PSF1) is  $h_{i,j} = (1 + i^2 + j^2)^{-1}$ , for  $i, j = -7 \dots 7$ . The next column reproduces the experiment in [28], [30], which uses a  $9 \times 9$  uniform kernel (PSF2). The fourth column uses a  $5 \times 5$  separable kernel (PSF3) with coefficients  $[1,4,6,4,1]/16$  as in [54]. The last two columns use Gaussian kernels with 1.6 and 0.4 standard deviation (PSF4 and PSF5, respectively). The noise standard deviations are shown in the second row of the table. The results show that [33] provides the best results in a 44.4% of the experiments, [24] + [25] in a 16.7% and the proposed method in a 38.9% of the experiments. Comparing with the BLS-GSM restoration original method [35], the new method provides a very significant improvement for all experiments, especially for the uniform degradation (PSF2), where the local BLS-GSM approach

completely fails (due to reasons explained in Section III-B). Compared to [33], note that, in general, our method achieves better results with *Barbara*, and worse with *House* and *Cameraman*. Being [33] a two-step method with a similar first global deconvolution step, the different behavior mainly comes from the denoising step [22] (based on the SA-DCT transform). This method is specially fitted to preserve sharp edges and contours (present in *House* and *Cameraman*), whereas it is not as powerful as ours to characterize local spectra. This translates into a poorer performance when dealing with highly textured images (as *Barbara*).

In Fig. 10, we show a visual comparison among our method, the state-of-the-art methods [24], [33] and BLS-GSM standard restoration [35]. For *Barbara*, the degradation used is PSF1 with  $\sigma = \sqrt{2}$ . In this case our method provides the best result, both visually and in PSNR terms. Particularly, the diagonal lines of the table cloth are better recovered by our method than by our competitors. For the *House* image, we have used PSF2 (uniform kernel) with  $\sigma = \sqrt{0.308}$ . Here, we can see how the previous BLS-GSM restoration method [35] completely fails, whereas the new version provides results competitive to the state-of-the-art [24], [33].

Finally, we also tried our deblurring algorithm with real degradations. We restored a micro-display image captured under controlled laboratory conditions. We used a tilted lens to produce severe blur and the noise was reduced before processing by averaging 30 captures. In Fig. 11, we show the degraded image and the result of our method using an estimation of the degradation parameters. As we can see, the method behaves very well, substantially improving the visible detail in the image.

## VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented two methods, for image denoising and deblurring. The denoising algorithm, based on space variant Gaussian scale mixtures, is applicable to any over-complete *wavelet* representation. Although the proposed model is just a conceptually simple extension of the one presented in [15], it provides a significant improvement in performance terms, still with a moderate computational cost. We have shown that the reason of the improvement is that, by modeling the local covariance spatial variability, it provides a more realistic model and a higher noise filtering selectivity. The improvement depends on both the image content and the representation used. We have tested two complementary representations concluding that one of them is more suitable for highly textured images (FSP), whereas the other (TIHP) is better for processing images for which texture is not so relevant.

Our deblurring method is based on a two-step decoupling scheme: linear global blur compensation and nonlinear local nonwhite noise removal. This strategy provides flexibility to use any denoising method able to deal with nonwhite Gaussian noise, as the one presented here. The resulting algorithm overcomes the severe limitations imposed by local approaches to image deconvolution while keeping their advantages. In addition, and in contrast to some state-of-the-art methods, it provides

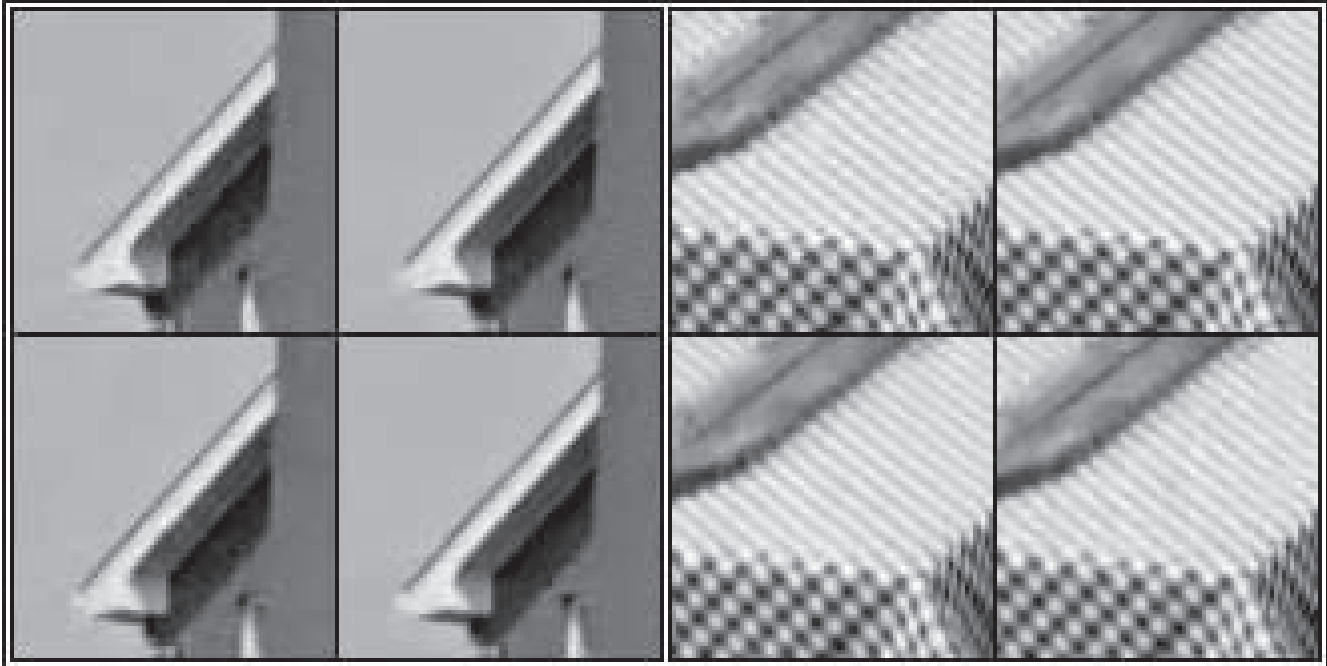


Fig. 8. Visual comparison of denoising results on *House* and *Barbara* images cropped to  $80 \times 80$  pixels (starting in pixel [1,78] and [1,221], respectively, [column, row]). Initial PSNR 28.13 dB in both cases ( $\sigma_w = 10$ ). From left to right and top to bottom (*House/Barbara* PSNR): results for [22] (35.98/33.48); results for [21] (36.71/34.98); results for [23] (35.94/34.42); our results using a  $16 \times 16$  block with TIHP/FSP representation (35.56/34.60).

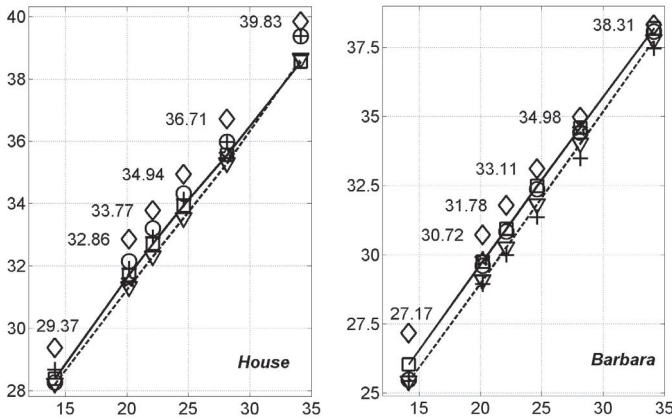


Fig. 9. PSNR output versus PSNR input, in decibels. We use TIHP for *House* and FSP for *Barbara*. Our results, for  $\sigma_w = \{5, 10, 15, 20, 25, 50\}$ , are in Table II. We compare to three state-of-the-art methods: diamonds for [21], crosses for [22], circles for [23], down triangles for [15], and squares for our new method. The latter two results are represented with dashed and solid lines, respectively.

TABLE III  
AVERAGED RESULTS AS ISNR, IN DECIBELS, OF THE TRAINING DATA SET (SEE SECTION III-B2 FOR DETAILS)

	$\sigma$	NOISE				
		1	2	4	8	16
B	0.4	2.86	1.86	2.81	5.03	7.63
L	0.8	6.65	4.84	3.73	3.92	5.85
U	1.6	3.87	3.18	2.72	3.01	4.73
R	3.2	3.21	2.85	2.56	2.66	3.87

a robust and simple solution to the problem of modeling and estimating the power spectral of the original image, avoiding the

TABLE IV  
PERFORMANCE COMPARISON WITH OTHER STATE OF THE ART METHODS IN TERMS OF ISNR, IN DECIBELS. FIRST ROW SHOWS THE BLURRING KERNEL USED (SEE TEXT FOR DETAILS), AND THE SECOND DENOTES NOISE STANDARD DEVIATION. BEST RESULTS ARE HIGHLIGHTED

Blur	<i>PSF 1</i>	<i>PSF 2</i>	<i>PSF 3</i>	<i>PSF 4</i>	<i>PSF 5</i>	
$\sigma_w \Rightarrow$	$\sqrt{2}$	$\sqrt{8}$	$\sqrt{0.308}$	7	2	8
<i>HOUSE</i>						
PSNR	25.62	25.47	24.11	28.10	27.83	30.01
[24]	8.47	6.63	<b>10.71</b>	4.22	<b>4.49</b>	4.76
[35]	8.46	6.93	-0.44	4.37	4.34	5.98
[33]	<b>9.05</b>	<b>7.64</b>	<b>10.71</b>	<b>5.10</b>	4.03	<b>6.13</b>
Ours	8.64	7.03	9.04	4.30	4.11	6.02
<i>CAMERAMAN</i>						
PSNR	22.23	22.16	20.77	24.63	23.36	29.83
[24]	7.46	5.24	8.16	2.84	3.18	3.65
[25]	6.93	4.88	7.59	<b>2.94</b>	-/-	-/-
[26]	7.40	5.15	8.10	2.85	-/-	-/-
[35]	6.84	5.29	-1.61	2.56	2.83	3.81
[33]	<b>8.25</b>	<b>6.34</b>	<b>8.57</b>	2.56	3.05	<b>5.15</b>
Ours	7.45	5.55	7.33	2.73	<b>3.25</b>	4.19
<i>BARBARA</i>						
PSNR	23.33	23.26	22.49	24.21	23.77	29.77
[24]	3.76	1.99	3.98	0.9	0.92	2.55
[35]	5.70	3.28	-0.27	1.44	0.95	4.91
[33]	5.73	3.01	4.88	1.58	0.91	4.04
Ours	<b>6.85</b>	<b>3.80</b>	<b>5.07</b>	<b>1.94</b>	<b>1.36</b>	<b>5.27</b>

necessity of estimating it from a first estimation of the original image (*re-estimation*).

As shown, the proposed empirical link between the two steps of the method (parameter setup) is suboptimal. It would have been much more elegant to obtain the absolute optimal prefilter by using an error model of the denoising method.



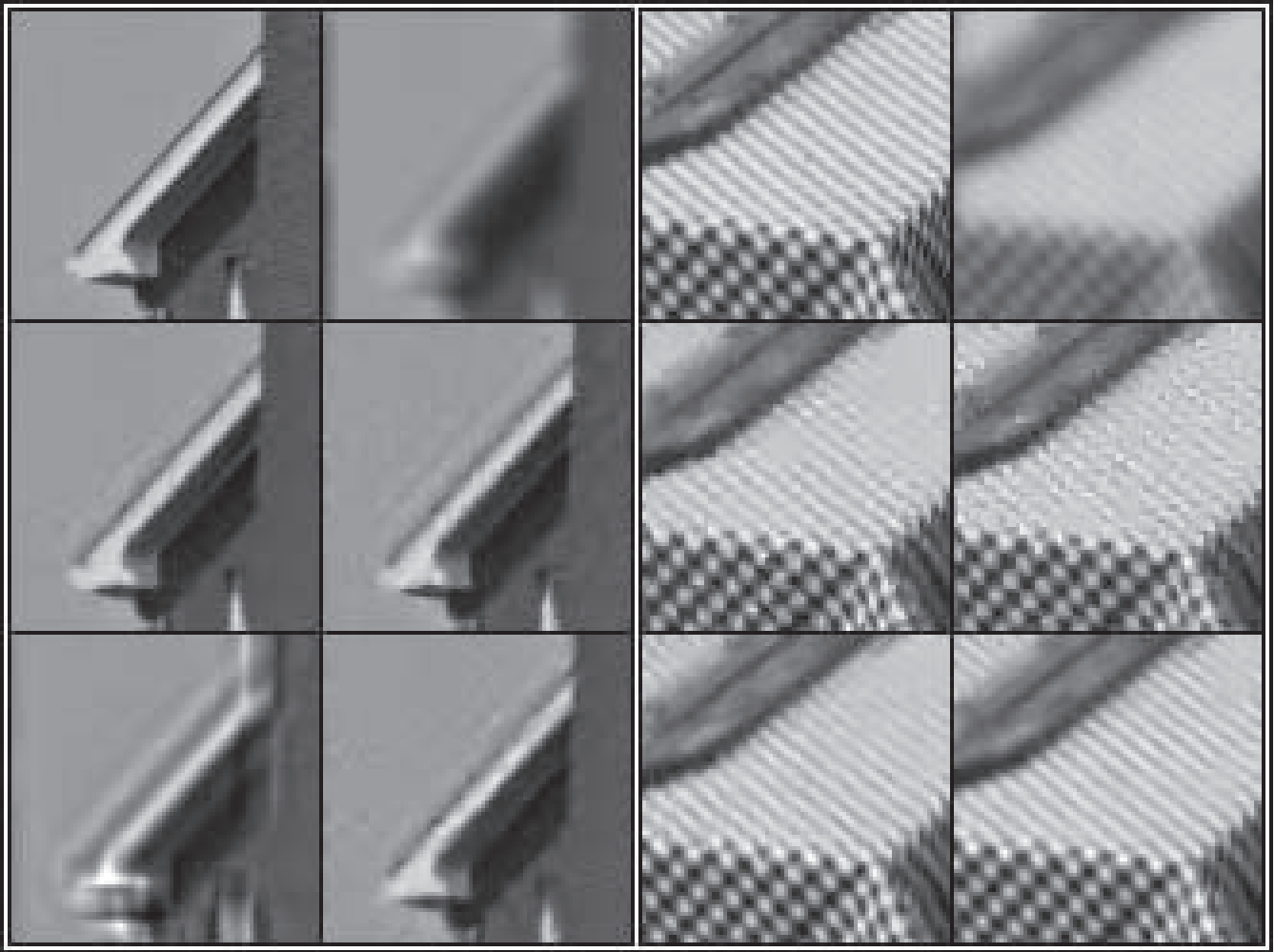


Fig. 10. Visual comparison of deblurring results on *House* and *Barbara* images cropped to  $80 \times 80$  pixels (starting in pixel [1,78] and [1,221], respectively, [column, row]). We include PSNR values, in decibels (in agreement with ISNR values of Table IV). From left to right and top to bottom (*House/Barbara*): original; degraded image (24.11/23.33) using PSF2 and  $\sigma_w = \sqrt{0.308}$  for *House* and PSF1 and  $\sigma_w = \sqrt{2}$  for *Barbara*; results for [33] (34.76/29.05); results for [24] (34.76/27.09); results of standard BLS-GSM [35] with FSP (23.68/29.02); our new method (*House* with TIHP, *Barbara* with FSP) (33.11/30.17).

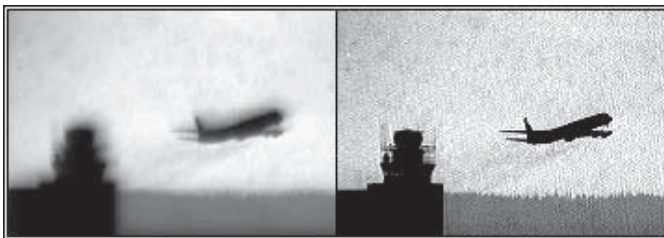


Fig. 11. Result of the method applied to a real degradation.

Unfortunately, that is a difficult goal, given the nonlinear character of high performance denoising methods. Nevertheless, the proposed empirical approach for choosing the prefiltering parameters is simple and robust: we have tested that the same prefiltering parameters can be used for different degradations without provoking a serious performance fall. This strategy is also very flexible, because the parameters can be easily tuned to any kind of degradation by training with kernels of the desired shape. All these features, coupled with the moderated

(and image-content independent) computational cost, make this method suitable to restore images in real situations [55].

Summarizing, the solutions presented here are model based, with few ad-hoc components that do not compromise a general conceptual coherence. Despite their simplicity, they provide state-of-the-art performance, both in denoising (slightly worse results than the best published ones) and deblurring (similar to the best ones on average).

A conceptually simple extension of the application of the methods presented here is to associate each spatial location with a different blurring kernel and/or with a different noise power spectral density. That would allow us to restore images with spatially variant degradations. This is interesting when dealing with low-quality optics, which cannot be accurately modeled as convolutions, because they are not translation invariant. Although this implies some technical challenges (mainly related to block's boundary handling), we believe that it is a feasible goal, with a potential impact on the imaging industry.

We believe that BLS-GSM-based models still have a strong potential for image processing. This work can be seen as a first

TABLE V

TOP: 1-D BUILDING FUNCTIONS, IN SPATIAL AND FREQUENCY DOMAIN. THE UNDERLINED COEFFICIENT REPRESENTS THE ONE AT THE ORIGIN. MIDDLE: DESCRIPTION OF THE 1-D REPRESENTATION RESPONSES. BOTTOM: TIHP ANALYSIS FILTERS IN TERMS OF EXTERNAL PRODUCTS OF THE 1-D BUILDING FUNCTIONS. THE SYNTHESIS FILTERS ARE THEIR COMPLEX CONJUGATES

ONE DIMENSIONAL ANALYSIS FILTERS	
<i>Building functions</i>	
Spatial Domain	Fourier Domain
$\phi_1 = [\underline{1}, 1]/2$	$\Phi_1(\omega) = (1 + e^{-j2\pi\omega})/2$
$\psi_1 = [\underline{1}, -1]/2$	$\Psi_1(\omega) = (1 - e^{-j2\pi\omega})/2$
$\phi_2 = [1, \underline{0}, 1]/2$	$\Phi_2(\omega) = \cos(2\pi\omega)$
$\psi_2 = [1, \underline{0}, -1]/2$	$\Psi_2(\omega) = j \sin(2\pi\omega)$
ONE DIMENSIONAL RESPONSES	
First scale	Second scale
$H_1(\omega) = \Psi_1(\omega)$	$H_2(\omega) = \Phi_1(\omega)\Psi_2(\omega)$
$L_1(\omega) = \Phi_1(\omega)$	$L_2(\omega) = \Phi_1(\omega)\Phi_2(\omega)$
<i>n<sup>th</sup> scale</i>	
$H_n(\omega) = L_{n-1}(\omega)j \sin(2^{n-1}\pi\omega), (\omega \leq 2^{1-n})$	
$L_n(\omega) = L_{n-1}(\omega) \cos(2^{n-1}\pi\omega), (\omega \leq 2^{1-n})$	
TIHP ANALYSIS FILTERS	
$A_v(u, v) = \Phi_1(u)\Psi_1(v)$	$B_v(u, v) = \Phi_2(u)\Psi_2(v)$
$A_h(u, v) = \Psi_1(u)\Phi_1(v)$	$B_h(u, v) = \Psi_2(u)\Phi_2(v)$
$A_d(u, v) = \Psi_1(u)\Psi_1(v)$	$B_d(u, v) = \Psi_2(u)\Psi_2(v)$
$A_l(u, v) = \Phi_1(u)\Phi_1(v)$	$B_l(u, v) = \Phi_2(u)\Phi_2(v)$

exploration stage, with promising results, towards a more ambitious goal, namely the generalization of the GSM model to include more than one covariance matrix simultaneously for each subband. This new model, a “mixture of Gaussian scale mixtures,” should be able to adapt to different textures in different spatial positions with no spatial constraints.

#### APPENDIX

##### TRANSLATION INVARIANT HAAR PYRAMID

We present an overcomplete version of the Haar orthogonal wavelet [45], which is translation invariant, due to the lack of aliasing, and which provides almost perfect reconstruction. Basically it is obtained by decimating its *à trous* [56] subbands from the third scale on, counting upwards. For simplicity, we start with the 1-D case, and after that, the 2-D extension is described.

At the top of Table V, the 1-D representation building functions are described (in spatial and frequency domains). Note how the samples at the origin are underlined.

The middle part of Table V shows the filter responses at the first and second scales in terms of the building functions. Fig. 12 shows the magnitude responses for four scales. The first two scales (two upper panels) coincide with those of the *à trous* representation (full resolution, no decimation). Further scales are decimated each time by a factor of 2, after applying an ideal low-pass filter to avoid intra-subband aliasing, as shown in the bottom half of Fig. 12. Dashed curves represent the corresponding *à trous* response. The vertical dashed lines indicate the cutoff frequency of the ideal low-pass filter. Frequencies on the right side are removed; thus, perfect reconstruction is no longer possible. This is not as damaging as it may seem, because image content is usually concentrated at low frequencies (e.g., we obtain reconstruction with PSNR of 59.59 dB for *Lena* and 44.60 dB for a  $512 \times 512$  white uniform noise in [0 255],

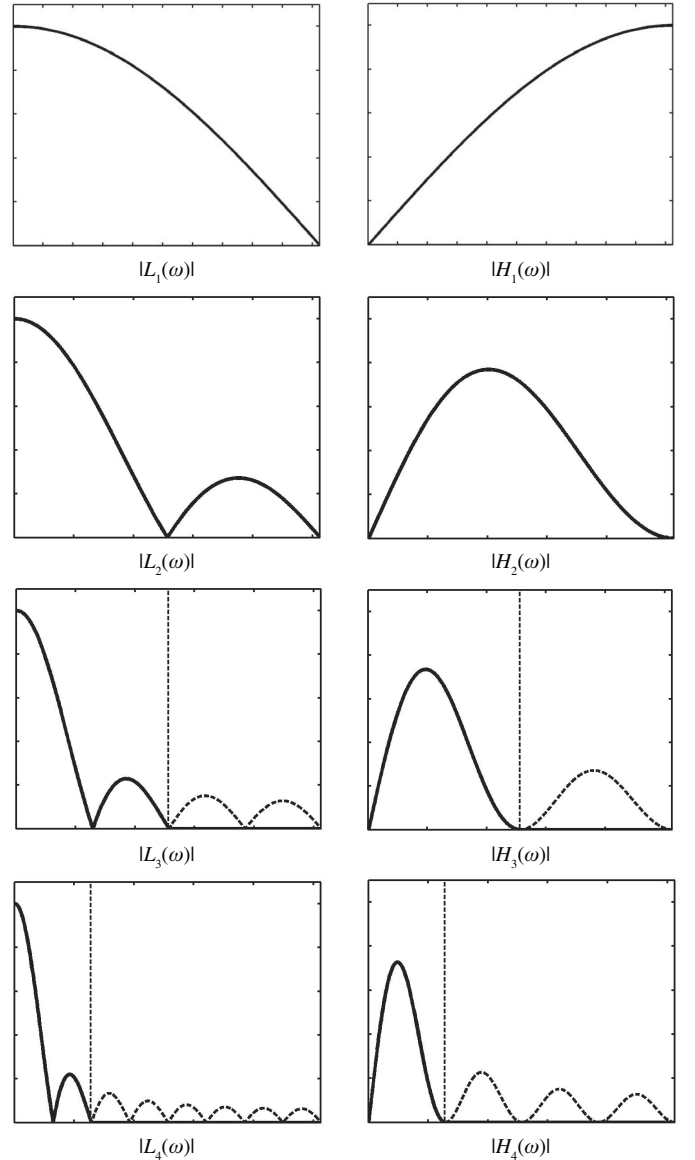


Fig. 12. Frequency response in module of the low and high/bandpass filters in a four-scale case. Dashed curves show the associated *à trous* response. Vertical straight lines depict the cutoff frequency at each scale. See middle part of Table V for filters descriptions.

both with five scales). Also, note that the cutoff frequencies coincide with a zero filter response in all cases, and also with a zero derivative for the high-pass filters. Thus, discontinuities are avoided and so they are most of the artifacts caused by them (ringing, poor spatial localization, etc.).

The extension to 2-D is done following the system diagram depicted in Fig. 13. The filters ( $A(u, v)$  and  $B(u, v)$  boxes) are built by properly doing external products of the 1-D filters, as shown at the bottom of Table V. The pyramid is implemented by recursively splitting the image into a set of oriented subbands (vertical, horizontal, and double diagonal) and a low pass band, which is subsampled by a factor of two along both axes. The reconstruction is achieved hierarchically by convolving each

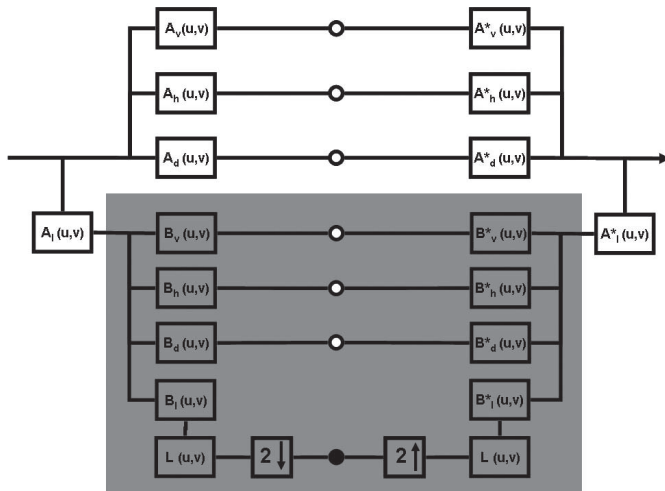


Fig. 13. System diagram for the translation invariant Haar pyramid. The input image is first split into six full resolution subbands by convolving with  $A$  and  $B$  filters. The recursion consists of inserting the diagram contents of the shaded region at the low-pass branch (solid circle).

subband with its complex-conjugated filter, and adding the outputs.

#### ACKNOWLEDGMENT

The authors would like to thank Prof. M. Elad, Dr. A. Foi, Dr. K. Dabov, and Prof. M. Figueiredo for providing them with their results and software for testing. They would also like to thank C. Dorronsoro and CIDA (Spain) for facilitating the capturing of real degraded images in their laboratories.

#### REFERENCES

- [1] J. A. Guerrero-Colón and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *Proc. IEEE Int. Conf. Image Processing*, Genoa, Italy, Sep. 2005, vol. 1, pp. 105–108.
- [2] J. A. Guerrero-Colón and J. Portilla, "Deblurring-by-denoising using spatially adaptive Gaussian scale mixtures in overcomplete pyramids," in *Proc. IEEE Int. Conf. Image Processing*, Atlanta, GA, Oct. 2006, pp. 625–628.
- [3] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989.
- [4] J. S. Lee, "Digital image enhancement and noise filtering by use of local statistics," *IEEE Pattern Anal. Mach. Intell.*, vol. PAMI-2, no. 3, pp. 165–168, Mar. 1980.
- [5] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.
- [6] S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," in *Proc. IEEE Int. Conf. Image Processing*, Chicago, IL, Oct. 1998, vol. 1, pp. 535–539.
- [7] A. Pižurica, W. Philips, I. Lemahieu, and M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Process.*, vol. 11, no. 5, pp. 545–557, May 2002.
- [8] L. Şendur and I. W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Trans. Signal Process.*, vol. 50, no. 11, pp. 2744–2756, Nov. 2002.
- [9] M. K. Mihçak, I. Kozintsev, K. Ramchandran, and P. Moulin, "Low-complexity image denoising based on statistical modeling of wavelet coefficients," *IEEE Trans. Signal Process.*, vol. 6, no. 12, pp. 300–303, Dec. 1999.
- [10] X. Li and M. T. Orchard, "Spatially adaptive image denoising under overcomplete expansion," in *Proc. IEEE Int. Conf. Image Processing*, Vancouver, BC, Canada, Sep. 2000, vol. 3, pp. 300–303.
- [11] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Adaptive Wiener denoising using a Gaussian scale mixture model in the wavelet domain," in *Proc. IEEE Int. Conf. Image Processing*, Thessaloniki, Greece, Oct. 2001, pp. 37–40.
- [12] D. Andrews and C. Mallows, "Scale mixtures of normal distributions," *J. Roy. Statist. Soc.*, vol. 36, pp. 99–102, 1974.
- [13] M. J. Wainwright, "Visual adaptation as optimal information transmission," *Vis. Res.*, vol. 39, pp. 3960–3974, 1999.
- [14] M. J. Wainwright and E. P. Simoncelli, "Scale mixtures of Gaussians and the statistics of natural images," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, May 2000, vol. 12, pp. 855–861.
- [15] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.
- [16] S. Mallat, G. Papanicolaou, and Z. Zhang, "Adaptive covariance estimation of locally stationary processes," *Ann. Statist.*, vol. 26, no. 1, pp. 1–47, 1998.
- [17] D. K. Hammond and E. P. Simoncelli, "Image denoising with an orientation-adaptive Gaussian scale mixture model," presented at the IEEE Int. Conf. Image Processing, Atlanta, GA, Oct. 2006.
- [18] A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005, vol. 2, pp. 60–65.
- [19] C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patch-based image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 2866–2878, Oct. 2006.
- [20] K. Hirakawa and T. W. Parks, "Image denoising using total least squares," *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2730–2742, Sep. 2006.
- [21] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [22] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1395–1411, May 2007.
- [23] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
- [24] M. Figueiredo and R. Nowak, "A bound optimization approach to wavelet-based image deconvolution," in *Proc. IEEE Int. Conf. Image Processing*, 2005, vol. 2, pp. 782–785.
- [25] M. Figueiredo and R. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Trans. Image Process.*, vol. 12, no. 8, pp. 906–916, Aug. 2003.
- [26] J. Bioucas-Dias, "Bayesian wavelet-based image deconvolution: A GEM algorithm exploiting a class of heavy-tailed priors," *IEEE Trans. Image Process.*, vol. 4, no. 4, pp. 937–951, Apr. 2006.
- [27] R. Molina, J. Mateos, A. K. Katsaggelos, and M. Vega, "Bayesian multichannel image restoration using compound Gauss–Markov random fields," *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1642–1654, Dec. 2003.
- [28] M. R. Banham and A. K. Katsaggelos, "Spatially adaptive wavelet-based multiscale image restoration," *IEEE Trans. Image Process.*, vol. 5, no. 4, pp. 619–634, Apr. 1996.
- [29] R. Neelamani, H. Choi, and R. G. Baraniuk, "Wavelet-domain regularized deconvolution for ill-conditioned systems," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, Oct. 1999, vol. 1, pp. 204–208.
- [30] R. Neelamani, H. Choi, and R. G. Baraniuk, "ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 418–433, Feb. 2004.
- [31] J. Kalifa and S. Mallat, "Mini-max restoration and deconvolution," in *Bayesian Inference in Wavelet Based Methods*. New York: Springer, 1999.
- [32] A. Jalobeanu, N. Kingsbury, and J. Zerubia, "Image deconvolution using hidden Markov tree modeling of complex wavelet packets," in *Proc. IEEE Int. Conf. Image Processing*, 2001, vol. 1, pp. 201–204.
- [33] A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-adaptive DCT for denoising and image reconstruction," in *Proc. SPIE Electronic Imaging, Image Processing: Algorithms and Systems V*, 2006, p. 6064A-18.
- [34] A. W. Stevenson, T. E. Gureyev, Y. I. Nesterets, and S. W. Wilkins, "A method for local deconvolution," *Appl. Opt.*, vol. 42, no. 32, pp. 6488–6494, Nov. 2003.

- [35] J. Portilla and E. P. Simoncelli, "Image restoration using Gaussian scale mixtures in the wavelet domain," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, Sep. 2003, vol. 2, pp. 965–968.
- [36] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multi-scale transforms," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 587–607, Mar. 1992.
- [37] R. R. Coifman and D. L. Donoho, "Translation-invariant de-noising," in *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, Eds. San Diego, CA: Springer-Verlag, 1995.
- [38] E. P. Simoncelli and W. T. Freeman, "The steerable pyramid: A flexible architecture for multi-scale derivative computation," in *Proc. IEEE Int. Conf. Image Processing*, Washington, DC, Oct. 1995, vol. III, pp. 444–447.
- [39] E. Candès and D. Donoho, *Curvelets: A Surprisingly Effective Non-adaptive Representation of Objects With Edges*. Nashville, TN: Vanderbilt Univ. Press, 1999.
- [40] N. Kingsbury, "Image processing with complex wavelets," *Phil. Trans. Roy. Soc. Lond. A*, Sep. 1999.
- [41] J. L. Starck, M. Elad, and D. L. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1570–1582, Oct. 2005.
- [42] J. L. Starck, D. L. Donoho, and E. Candès, "Very high quality image restoration," in *Proc. SPIE Conf. Signal and Image Processing*, San Diego, CA, Aug. 2001, vol. 4478, pp. 9–19.
- [43] J. K. Romberg, M. B. Wakin, and R. G. Baraniuk, "Multiscale geometric image processing," *Proc. SPIE*, vol. 5150, pp. 1265–1272, 2005.
- [44] E. L. Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 423–438, Apr. 2005.
- [45] A. Haar, "Zur Theorie der orthogonalen Funktionensysteme," *Ann. Math.*, vol. 69, pp. 331–371, 1910.
- [46] E. R. Kretzmer, "The statistics of television signals," *Bell Syst. Tech. J.*, vol. 31, pp. 751–763, 1952.
- [47] N. G. Deriugin, "The power spectrum and the correlation function of the television signal," *Telecommunications*, vol. 1, no. 7, pp. 1–12, 1956.
- [48] E. P. Simoncelli, "Statistical models for images: Compression, restoration and synthesis," in *Proc. 31st Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Nov. 1997, pp. 673–678.
- [49] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3445–3462, Dec. 1993.
- [50] R. W. Buccigrossi, "Compression and segmentation of images using an inter-subband wavelet probability model," Ph.D. dissertation, Dept. Comput. Inf. Sci., Univ. Pennsylvania, Philadelphia, PA, Jun. 1999.
- [51] J. Portilla and E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *Int. J. Comput. Vis.*, vol. 40, no. 1, pp. 49–70, 2000.
- [52] S. Lyu and E. P. Simoncelli, "Statistical modeling of images with fields of Gaussian scale mixtures," *Adv. Neural Inf. Process. Syst.*, vol. 19, May 2007.
- [53] E. Vansteenkiste, D. Van der Weken, W. Philips, and E. E. Kerre, "Perceived image quality measurement of state-of-the-art noise reduction schemes," *Adv. Concepts Intell. Vis. Syst.*, pp. 114–126, 2006.
- [54] J. Liu and P. Moulin, "Complexity-regularized image restoration," in *Proc. IEEE Int. Conf. Image Processing*, 1998, vol. 1, pp. 555–559.

- [55] J. Portilla, J. A. Guerrero-Colón, and C. Dorronsoro, Patente: Procedimiento de restauración de imágenes afectadas por imperfecciones, dispositivo para llevarlo a cabo y sus aplicaciones, P2006021116, Filed Aug. 2006.
- [56] M. J. Shensa, "The discrete wavelet transform: Wedding the à trous and Mallat algorithms," *IEEE Trans. Signal Process.*, vol. 40, no. 10, pp. 2464–2482, Oct. 1992.



**Jose A. Guerrero-Colón** was born in Málaga, Spain. He received the M.S. degree in computer science from the Escuela Técnica Superior de Ingeniería Informática, Universidad de Málaga, in 2003. He is currently pursuing the Ph.D. degree and working on several projects related to his main topic: new visual-statistical models applied to the restoration of photographic images corrupted by blur and noise.

He is a grant holder of an FPU contract, funded by the Ministerio de Educación y Ciencia, from the Spanish Government, at the Department of Computer

Science and Artificial Intelligence, Universidad de Granada, Spain.



**Luis Mancera** received the M.S. degree in computer science from the Escuela Técnica Superior de Ingeniería Informática, Universidad de Granada, Spain, in 2001. He is currently pursuing the Ph.D. degree in nonlinear sparse image representation methods and their applications.

For two years, he was with the European Centre for Nuclear Research (CERN), Geneva, Switzerland. He is a grant holder of an FPI contract at the Department of Computer Science and Artificial Intelligence, Universidad de Granada, Spain, funded by Ministerio de

Educación y Ciencia, from the Spanish Government.



**Javier Portilla** received the M.S. and Ph.D. degrees from the E.T.S.I. Telecomunicación, Universidad Politécnica de Madrid, Spain, in 1994 and 1999, respectively.

After a postdoctoral stay at New York University, he was an Assistant Professor at the Computer Science and Artificial Intelligence Department, Universidad de Granada, Spain. He obtained a "Ramón y Cajal" research position, which he still enjoys, now in the Instituto de Óptica, Consejo Superior de Investigaciones Científicas, Madrid.

He is interested in image statistical modeling and its degradation sources for designing Bayesian methods, as well as new, general-purpose nonlinear image representation schemes, inspired by biological vision, and their application to image processing.



### S.2.3. Image restoration using adaptive Gaussian scale mixtures in overcomplete pyramids

- J. Portilla, J. A. Guerrero-Colón, "Image restoration using adaptive Gaussian scale mixtures in overcomplete pyramids". Invited talk in Wavelets XII, in SPIE's International Symposium on Optical Science and Technology. SPIE's 52th Annual Meeting Proc. of the SPIE (August 2007).



# Image restoration using adaptive Gaussian scale mixtures in overcomplete pyramids

Javier Portilla<sup>a</sup> and Jose A. Guerrero-Colon<sup>b</sup>

<sup>a</sup>Dept. of Images and Vision, Instituto de Óptica, CSIC, Madrid, Spain;

<sup>b</sup>Dept. of Computer Science and Artif. Intel. (DECSAI), Universidad de Granada, Spain

## ABSTRACT

We describe here two ways to improve on recent results in image restoration using Bayes least squares estimation with local Gaussian scale mixtures (BLS-GSM) in overcomplete oriented pyramids. First one consists of allowing for a spatial adaptation of the covariance matrix defining the GSM model at each pyramid subband. This can be implemented in practice by dividing the subbands into spatial blocks. The other, more powerful, method is to generalize the GSM model to include more than one covariance matrices for each subband. The advantage of the latter method is its flexibility, as it allows for mixing Gaussian densities with different covariance matrices at every spatial location in every subband. It also allows for non-local selective processing, taking advantage of the repetition in the scene of image features that are not necessarily spatially grouped. We also describe an empirical method to adapt denoising algorithms for doing image restoration, with the only constraint on the denoising method of being applicable to non-white noise sources. Here we present mature results of the spatially adaptive method applied to denoising and deblurring, plus some estimation techniques and encouraging preliminary results of the multi-GSM concept.

**Keywords:** image statistics, image restoration, noise removal, Gaussian scale mixtures, Bayesian estimation

## 1. INTRODUCTION

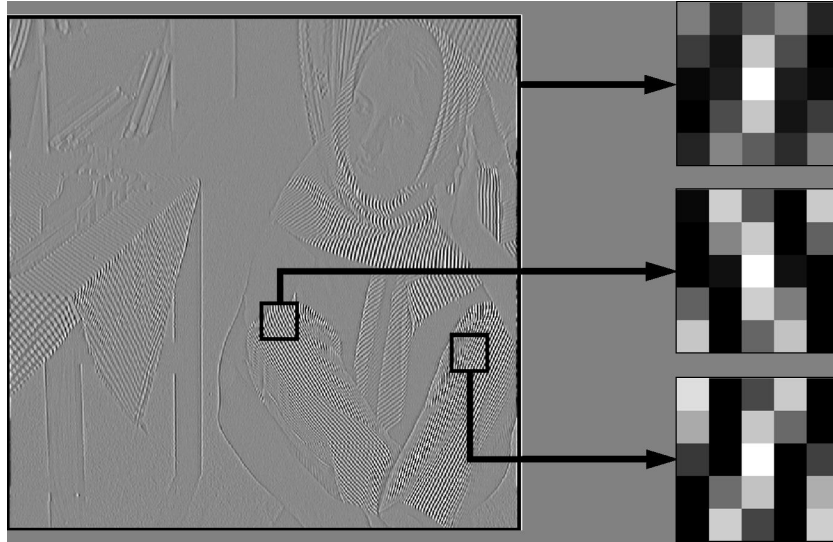
Naturalistic images may be described, in a first approximation, as made of homogeneous texture patches with regular boundaries.<sup>1,2</sup> When using a translation invariant overcomplete linear representation the filtering made by the analysis functions forces the local spectrum to be *similar* at different locations. This fact allows to use a single Gaussian Scale Mixture (GSM), with its corresponding single covariance matrix, to model the neighborhoods of coefficients in each subband of an oriented pyramid.<sup>3</sup> Differences in local amplitude (caused, in this model, by the different contrast of the texture patches and by their boundaries) are represented by the scale of the GSM model. But, how *similar* are typically the local spectra in a subband at different spatial locations? It depends on both the representation and the image.<sup>4,5</sup> When using highly selective in frequency representations (such as curvelets or the steerable pyramid with a high number of orientations) the covariance is less variable than when using representations more selective in the space and less selective in frequency (as, e.g., the Translation Invariant Haar Pyramid, an overcomplete version of the Haar wavelet that we have used in this work<sup>4-6</sup>). This variability within each subband is illustrated in Fig. 1. By adapting to the dominant local spectra we are allowing for a more selective filtering of the original signal from the noisy observation.

However, one can also criticize the "spatially connected texture patches" model as too simplistic. The boundaries of the different textured regions in natural images are often not sharply defined, and textures may interleave each other. Furthermore, it is normal to have similar features that are not spatially connected (e.g., coming from brick walls from different houses, bark from different trees, etc.), whereas we usually also have different features within a single texture. It is clearly suboptimal to use only a local neighborhood for characterizing the signal statistics, if the locally observed features are repeated, with more or less variability, at other spatial locations. We look for a way of exploiting this redundancy in our estimation. Some recent denoising methods with impressive results own big part of their success to exploiting this *non-local redundancy* (e.g.<sup>7-10</sup>). We have seen that for

---

JP (portilla@io.cfmac.csic.es) and JAGC (jaguerrero@decsai.ugr.es) are supported by TEC2006/13845/TCM grant. JP is under the "Ramón y Cajal" Program. JAGC is funded by AP2005-2969 FPU fellowship. All funds provided by the Spanish Ministry of Education and Science (M.E.C.)





**Fig. 1.** Autocovariance variability within Barbara in a high frequency vertical subband of a translation invariant version of the Haar pyramid (TIHP). We used a  $3 \times 3$  neighborhood and estimate the autocovariance central samples from the observed neighborhoods in small blocks.

characterizing this type of redundancy in the image contents neither a single GSM per subband<sup>3</sup> nor a simple space variant GSM<sup>4</sup> is efficient. Instead, we may allow for more than one covariance matrix at each subband, without constraining them to be locally estimated. Also, we do not want to lose the GSM ability to adapt to changes in amplitude, either rapid ones (edges, corners, etc.) or more gentle (shading, etc.). So here we propose a "mixture of Gaussian scale mixtures", with the power and flexibility of the standard GSM model, but with the extra ability (1) to exploit non-locally redundant features and (2) to adapt to multiple and/or spatially variant textures. The biggest advantage of this approach is that we can use a single covariance matrix to characterize similar spatial features found at different (in general, non-adjacent) spatial locations. Their differences, due in big part to the image degradation, will tend to have a modest influence in the model parameters, and, instead, their repetition will help to more accurately characterize the original image features. By using some simplified implementation of the multi-GSM model, we have recently obtained some encouraging preliminary results<sup>\*</sup>, which are presented here.

We also briefly describe in this paper the application of the denoising techniques explained here (or any other able to cope with non-white noise) to perform image restoration. That is, the case when, besides an additive noise source, there is a blurring kernel. For this purpose we have followed the "first deconvolving, then denoising" strategy<sup>5,11-13</sup> see Fig. 2. This approach requires to choose a global linear filtering for pre-compensating the blur. We have used here a simplified regularization approach which allows to apply an empirical optimization ("training") technique for choosing the pre-filtering kernel.<sup>5</sup>

In summary, in this work we further advance, on the one hand, in the direction of making increasingly realistic descriptions of the natural images, while trying to keep the estimation approach conceptually simple and model-based<sup>†</sup>. In our case, we base on (conditional) Gaussian densities and, thus, in all proposed variants for image denoising we estimate the original through advanced versions of the Wiener filtering. On the other hand, we extend the application of the presented techniques to image deconvolution. The locally adaptive BLS-GSM model was first described in,<sup>4</sup> and further applied, in combination with our "denoising-by-deblurring" technique in.<sup>5</sup> The referred contents are described in detail in a recently submitted journal paper.<sup>14</sup> The multi-GSM model presented here is, to the best of our knowledge, new.

<sup>\*</sup>In collaboration with Prof. Eero Simoncelli (New York University), see Section 3.3.

<sup>†</sup>The only non-model based part of this work is the empirical training used for optimizing the pre-filtering stage in the image deblurring. Nevertheless, the referred procedure is close to optimal under the assumed constraints

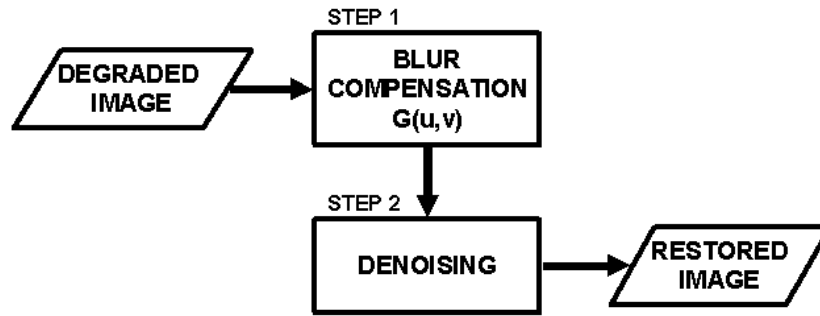


Fig. 2: Two step restoration scheme.

### 1.1. Image statistics and Gaussian scale mixtures

Usual approximation, fifty years ago, was to consider images as samples from highly correlated Gaussian random fields (e.g.,<sup>15</sup>). However, in 1987 Field<sup>16</sup> showed that oriented band-pass responses to typical images are far from Gaussian: only a small proportion of the coefficients concentrates a high proportion of the total energy. Relatively few high amplitude responses are caused by most relevant localized image features (such as edges, lines, corners, etc.), whereas spatially dominant responses are typically small in amplitude, mainly corresponding to low-contrast texture. Therefore, the dominant source for non-Gaussianity of image statistics is the inhomogeneity of the image contents, which causes abrupt changes and localized high amplitude responses at the objects' boundaries. In 1997, Simoncelli<sup>1</sup> made another critical observation about image statistics: there is a strong coupling among neighbor *wavelet* coefficients in amplitude. This revealed explicitly a fundamental non-Gaussian joint statistical behavior that was coherent with the marginal densities studied so far. This feature has been exploited for image compression,<sup>17,18</sup> texture modelling and synthesis,<sup>19</sup> and image restoration (all the predecessors of this work), among other applications. In 1999 Wainwright and Simoncelli found that Gaussian scale mixtures, GSM,<sup>20</sup> was the proper mathematical tool to capture both the sparseness and the amplitude coupling of wavelet coefficients.<sup>21</sup> GSM's have typically been used since then as a local description for neighbor coefficients in a pyramid (e.g.<sup>3</sup>), but recently they have also been used in a more powerful global frame in the transformed domain.<sup>22</sup>

## 2. LOCALLY ADAPTIVE BLS-GSM

In<sup>4</sup> we proposed enhancing the BLS-GSM denoising method by allowing for a local characterization of the signal covariance at each subband of the representation. Conceptually, this was an extremely simple extension of the original model, but it was nevertheless very effective in terms of improving the denoising performance. A similar successful approach, but applied to the local image orientation, has also been recently proposed.<sup>23</sup>

Let's consider an image decomposed into oriented subbands at multiple scales. We denote as  $x_c^j(n, m)$  the coefficient corresponding to the subband  $j$ , centered at spatial location  $(n, m)$ . We denote as  $\mathbf{x}^j(n, m)$  a *neighborhood* of coefficients clustered around this *reference coefficient*. We model the neighborhoods of a given subband as samples of a vector Gaussian scale mixture (GSM). A random vector  $\mathbf{x}$  is a GSM<sup>20</sup> if it can be expressed as the product of two independent random variables:

$$\mathbf{x} = \sqrt{z}\mathbf{u},$$

where  $z$  is a positive scalar and  $\mathbf{u}$  is a zero-mean Gaussian vector. We model the desired space variant behavior by defining  $\mathbf{C}_u^{j,(n,m)}$  as the (space variant) covariance of  $\mathbf{u}^j(n, m)$ . The density of  $\mathbf{x}$  is determined uniquely by  $p_z(z)$  and  $\mathbf{C}_u^{j,(n,m)}$  as follows:

$$\begin{aligned} p^{j,(n,m)}(\mathbf{x}) &= \int p^{j,(n,m)}(\mathbf{x}|z) p_z(z) dz \\ &= \int \frac{\exp\left(-\mathbf{x}^T(z\mathbf{C}_u^{j,(n,m)})^{-1}\mathbf{x}/2\right)}{(2\pi)^{N/2}|z\mathbf{C}_u^{j,(n,m)}|^{1/2}} p_z(z) dz, \end{aligned}$$

where  $N$  is the dimensionality of  $\mathbf{x}$  and  $\mathbf{u}$  (the size of the neighborhood). Note that  $\mathbf{x}^j(n, m)$  is conditionally Gaussian for a given  $z$ . We have used for the results shown in this paper a non-informative Jeffrey's prior for the mixing density  $p_z(z)$ , that is, a constant in  $\log z$  in the working range, for all the subbands.<sup>3</sup>

## 2.1. Denoising using space-variant GSMs

We aim to solve the classical denoising problem, where an image is corrupted by additive zero-mean independent Gaussian noise of known (but arbitrary) spectral density. As many other methods, we perform the denoising in the pyramid domain and obtain the image estimation by reconstructing the image from the estimated coefficients.

### 2.1.1. Observation model

The observation model is, in this case:

$$\mathbf{y} = \mathbf{x} + \mathbf{w},$$

where  $\mathbf{y}$ ,  $\mathbf{x}$  and  $\mathbf{w}$  are vectors (in a lexicographic order) of observation, original and noisy wavelet coefficients, respectively. Following the GSM model presented in section 1.1, we can re-express the neighborhood of  $N$  noisy coefficients as<sup>†</sup>:

$$\mathbf{y} = \sqrt{z}\mathbf{u} + \mathbf{w},$$

where  $\mathbf{w}$  is the zero mean Gaussian noise vector with covariance matrix  $\mathbf{C}_w$ . Note that  $\mathbf{C}_w$  (unlike the signal covariance  $\mathbf{C}_u$ ) is assumed constant for all neighborhoods within a subband (but different for each subband). By GSM definition  $z$  and  $\mathbf{u}$  are independent, and by assumption  $\mathbf{w}$  is independent too. Hence the density of the observed neighborhood vector conditioned on  $z$  is Gaussian (zero-mean), with covariance  $\mathbf{C}_{y|z} = z\mathbf{C}_u + \mathbf{C}_w$ :

$$p(\mathbf{y}|z) = \frac{\exp(-\mathbf{y}^T(z\mathbf{C}_u + \mathbf{C}_w)^{-1}\mathbf{y}/2)}{((2\pi)^N |z\mathbf{C}_u + \mathbf{C}_w|)^{1/2}}. \quad (1)$$

### 2.1.2. Parameter estimation

We need to estimate the signal and noise covariances of Eq. 1. The noise covariance,  $\mathbf{C}_w$ , can be computed at each subband by applying the pyramidal representation to a deterministic function with the same sample autocorrelation as the noise (for instance, a scaled delta function for white noise). The elements of  $\mathbf{C}_w$  are computed by averaging the products of pairs of coefficients over all the neighborhoods of the subband. Note that, unlike a noise sample, this deterministic function yields an estimation free from random fluctuations.

The observation covariance,  $\mathbf{C}_y$ , can be computed from  $\mathbf{C}_{y|z}$  by taking the expectation over  $z$ :

$$\mathbf{C}_y = \mathbb{E}\{\mathbf{C}_{y|z}\} = \mathbb{E}\{z\}\mathbf{C}_u + \mathbf{C}_w.$$

We set, without loss of generality,  $\mathbb{E}\{z\} = 1$  obtaining  $\mathbf{C}_y = \mathbf{C}_u + \mathbf{C}_w$ . To estimate the space variant signal covariance  $\mathbf{C}_u$  we assume that it changes smoothly over the image, and, thus, that it may be sampled by estimating it into a set of non-overlapping regions (we have tested that this assumption is correct by experiencing a very small improvement in denoising performance when using instead overlapping blocks). We divide each subband into  $B \times B$  blocks, assuming that the subband local covariance does not change much typically within a block. For each of this blocks we estimate  $\mathbf{C}_u$  by subtracting  $\mathbf{C}_w$  from the sample covariance of its coefficients,  $\mathbf{C}_y$ , and setting to zero any resulting negative eigenvalue.

<sup>†</sup>For notational simplicity, we drop the superscript  $j$  and the indices  $(n, m)$  in the following.

### 2.1.3. Coefficient estimation

For each neighborhood  $\mathbf{y}$  of observed coefficients, we estimate  $x_c$ , the reference coefficient, at the center of the neighborhood. The Bayes least squares (BLS) estimate can be expressed in this case<sup>3</sup> as

$$\mathbb{E}\{x_c|\mathbf{y}\} = \int_0^\infty p(z|\mathbf{y}) [\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\}]_{x_c} dz,$$

that is, the solution is written as the average of the Bayes least squares estimate of  $x_c$  when conditioned on  $z$ , weighted by the posterior density of  $z$ ,  $p(z|\mathbf{y})$ . A key feature of the GSM model is that the coefficient neighborhood vector  $\mathbf{x}$  is Gaussian when conditioned on  $z$ . This fact, coupled with the assumption of additive Gaussian noise means that the expected value inside the integral of (2) is simply the Wiener estimate:

$$\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\} = z\mathbf{C}_u(z\mathbf{C}_u + \mathbf{C}_w)^{-1}\mathbf{y}.$$

We can simplify the dependence of this expression on  $z$  by diagonalizing the matrix  $z\mathbf{C}_u + \mathbf{C}_w$  (see<sup>3</sup>). Note that, unlike in<sup>3</sup> now this diagonalization depends on the spatial location (as  $\mathbf{C}_u$  is a function of  $(n, m)$ ), and, thus, it needs to be computed once per each subband block.

## 3. A NON-LOCALLY ADAPTIVE APPROACH: MULTI-GSM

### 3.1. The multi-GSM signal model

Here we model the local neighborhoods of coefficients from a subband of the original image as a *mixture of Gaussian scale mixtures*:

$$\mathbf{x} = \sqrt{z_k}\mathbf{u}_k,$$

where  $z_k \sim p_k(z)$  represents a scalar multiplier,  $\mathbf{u}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_k)$  is a zero mean Gaussian  $N$ -dimensional vector, and the index  $k$  is drawn according to a given discrete probability  $P_k$ ,  $k \sim P_k$ . Therefore, the set of parameters of the multi-GSM subband model is: (1) the shape and size ( $N$ ) of the considered neighborhood; (2) the number of mixture components  $K$ ; (3) the set of  $K$  mixing scale densities  $p_k(z) = p_z(z|k)$ ; and (4) the set of  $K$  covariance matrices  $\mathbf{C}_k$ .

From this it is easy to derive the probability density function of the modelled neighborhood vector:

$$p(\mathbf{x}) = \sum_{k=1}^K P_k \int_z \frac{\exp(-\mathbf{x}^T(z\mathbf{C}_k)^{-1}\mathbf{x}/2)}{(2\pi)^{N/2}|z\mathbf{C}_k|^{1/2}} p_k(z) dz$$

### 3.2. Denoising using multi-GSM

#### 3.2.1. The observation model

Similarly as in 2.1.1, we have:

$$\mathbf{y} = \sqrt{z_k}\mathbf{u}_k + \mathbf{w},$$

where  $\mathbf{w}$  is a zero mean Gaussian noise vector with covariance matrix  $\mathbf{C}_w$ . The density of the observed neighborhood vector conditioned on  $z$  and  $k$  is Gaussian (zero-mean), with covariance  $\mathbf{C}_{\mathbf{y}|z,k} = z\mathbf{C}_k + \mathbf{C}_w$ :

$$p(\mathbf{y}|z, k) = \frac{\exp(-\mathbf{y}^T(z\mathbf{C}_k + \mathbf{C}_w)^{-1}\mathbf{y}/2)}{((2\pi)^N|z\mathbf{C}_k + \mathbf{C}_w|)^{1/2}}. \quad (2)$$

The density of the observed vector  $\mathbf{y}$  is:

$$p(\mathbf{y}) = \sum_{k=1}^K P_k \int_z \frac{\exp(-\mathbf{y}^T(z\mathbf{C}_k + \mathbf{C}_w)^{-1}\mathbf{y}/2)}{(2\pi)^{N/2}|z\mathbf{C}_k + \mathbf{C}_w|^{1/2}} p_k(z) dz$$

### 3.2.2. BLS Coefficient estimation

The Bayes Least Squares estimation for a given observation  $\mathbf{y}$  is  $\hat{\mathbf{x}}_{BLS}(\mathbf{y}) = \mathbb{E}\{\mathbf{x}|\mathbf{y}\}$ . For computing that we need to know the posterior density  $p(\mathbf{x}|\mathbf{y})$ , which in our case can be written as:

$$p(\mathbf{x}|\mathbf{y}) = \sum_{k=1}^K P(k|\mathbf{y}) \int_z p(\mathbf{x}|\mathbf{y}, k, z) p_k(z|\mathbf{y}) dz.$$

Then, the BLS estimator is:

$$\begin{aligned} \hat{\mathbf{x}}_{BLS}(\mathbf{y}) &= \int_{\mathbf{x}} \mathbf{x} p(\mathbf{x}|\mathbf{y}) d\mathbf{x} \\ &= \int_{\mathbf{x}} \mathbf{x} \sum_{k=1}^K P(k|\mathbf{y}) \int_z p(\mathbf{x}|\mathbf{y}, k, z) p_k(z|\mathbf{y}) dz d\mathbf{x} \\ &= \sum_{k=1}^K P(k|\mathbf{y}) \int_z \int_{\mathbf{x}} \mathbf{x} p(\mathbf{x}|\mathbf{y}, k, z) d\mathbf{x} p_k(z|\mathbf{y}) dz \\ &= \sum_{k=1}^K P(k|\mathbf{y}) \int_z \mathbb{E}\{\mathbf{x}|\mathbf{y}, k, z\} p_k(z|\mathbf{y}) dz, \end{aligned}$$

where

$$\mathbb{E}\{\mathbf{x}|\mathbf{y}, k, z\} = z\mathbf{C}_k(z\mathbf{C}_k + \mathbf{C}_w)^{-1}\mathbf{y},$$

which is a Wiener solution, due to the involved conditional densities are Gaussian. Thus, we can write the final estimation as a weighted average of the BLS-GSM model, according to the posterior probability of the mixture components:

$$\hat{\mathbf{x}}_{BLS}(\mathbf{y}) = \sum_{k=1}^K P(k|\mathbf{y}) \hat{\mathbf{x}}_{BLS-GSM}(\mathbf{y}; \mathbf{C}_k, p_k(z)).$$

The simplicity of this result is in contrast with the relative complexity of estimating the involved mixture components posterior probabilities, as shown next.

### 3.2.3. Parameter estimation

In order to obtain  $P(k|\mathbf{y})$  through the Bayes rule we need to previously have estimates of  $p(\mathbf{y}|k)$  and  $P_k$ . We can express

$$p(\mathbf{y}|k) = \int_z p(\mathbf{y}|k, z) p_k(z) dz,$$

where we know the (Gaussian) expression of  $p(\mathbf{y}|k, z)$ , but we do not know a priori  $p_k(z)$ . On the other hand, we can express  $P_k$  as

$$\begin{aligned} P_k &= \int_{\mathbf{y}} P(k|\mathbf{y}) p(\mathbf{y}) d\mathbf{y} \\ &\simeq \langle P(k|\mathbf{y}_m) \rangle_m. \end{aligned}$$

where  $\langle \cdot \rangle_j$  represents sample average in index  $j$ , and we are assuming here that we have a set of independent observations  $\{\mathbf{y}_m, m = 1 \dots M\}$  per subband. The average of the posteriors associated to each sample provides the maximum likelihood estimate of the involved probability. Now, by applying the Bayes rule to each posterior we arrive to the following iterative computation:

$$P_k^{(n+1)} = P_k^{(n)} \frac{1}{M} \sum_{m=1}^M \frac{p(\mathbf{y}_m|k)^{(n)}}{\sum_{j=1}^K p(\mathbf{y}_m|j)^{(n)} P_j^{(n)}} \quad (3)$$

where we can use

$$p(\mathbf{y}|k)^{(n)} = \int_z p(\mathbf{y}|k, z)p_k(z)^{(n)} dz. \quad (4)$$

The function  $p(\mathbf{y}|k, z)$  is known given  $z, k$  and  $\mathbf{C}_k$ . By now we are assuming that we know  $\{\mathbf{C}_k, k = 1 \dots K\}$  (but, as we explain below, that is not the usual case in practice, so we will also iteratively estimate  $p(\mathbf{y}|k, z)$ ). Then the only piece missing is an iterative expression for updating our  $p_k(z)$  estimates. Applying Bayes rule to  $p(\mathbf{y}|k)$  in

$$p_k(z) = \int_{\mathbf{y}} p(z|\mathbf{y}, k)p(\mathbf{y}|k) d\mathbf{y}$$

we get

$$p_k(z) = \frac{1}{P_k} \int_{\mathbf{y}} p(z|\mathbf{y}, k)P(k|\mathbf{y})p(\mathbf{y}) d\mathbf{y},$$

which becomes, after decomposing the posterior densities and expressing the ML estimate as their sample average:

$$p_k(z)^{(n+1)} = p_k(z)^{(n)} \frac{1}{MP_k^{(n)}} \sum_{m=1}^M \frac{p(\mathbf{y}_m|k, z)P(k|\mathbf{y}_m)^{(n)}}{\int_{z'} p(\mathbf{y}_m|k, z')p_k(z')^{(n)} dz'}. \quad (5)$$

Finally, to update our estimate of  $P(k|\mathbf{y})$  we simply apply, again, the Bayes rule:

$$P(k|\mathbf{y}_m)^{(n)} = \frac{p(\mathbf{y}_m|k)^{(n)}P_k^{(n)}}{\sum_{j=1}^K p(\mathbf{y}_m|j)^{(n)}P_j^{(n)}}. \quad (6)$$

Equations (3-6) give us a means to, starting from some initial guess for  $P_k$  and  $p_k(z)$ , and assuming we know  $\mathbf{C}_k$ , for every  $k$ , obtaining estimations of  $P(k|\mathbf{y})$  that locally maximize the global log-likelihood function. However, usually we do not know either the covariance matrices  $\mathbf{C}_k$ , so they must also be estimated iteratively in an (G)EM-like fashion. Here we have not enough space for the whole ML derivation, which gives rise to the following updating rule (to be used simultaneously to the previous ones):

$$\mathbf{C}_k^{(n)} = \left[ \left( \int_z \mathbf{C}_y(k, z)^{(n)} p_k(z)^{(n)} dz \right) - \mathbf{C}_w \right]_+, \quad (7)$$

where  $[\cdot]_+$  means here setting to zero the negative eigenvalues, and

$$\begin{aligned} \mathbf{C}_y(k, z)^{(n)} &= \frac{\sum_{m=1}^M P(k|\mathbf{y}_m)^{(n)} p_k(z|\mathbf{y}_m)^{(n)} \mathbf{y}_m \mathbf{y}_m^T}{\sum_{m=1}^M P(k|\mathbf{y}_m)^{(n)} p_k(z|\mathbf{y}_m)^{(n)}} \\ &\simeq z \mathbf{C}_k + \mathbf{C}_w. \end{aligned} \quad (8)$$

Without loss of generality, we have assumed that  $\int_z z p_k(z) dz = 1$ , for all  $k$ , and thus  $(\int_z z \mathbf{C}_k + \mathbf{C}_w p_k(z) dz) - \mathbf{C}_w = \mathbf{C}_k$ . The approximation (8) is justified by the GSM structure of the original data when conditioned on a given mixture component  $k$ .

### 3.3. Simplifications and approximations in a preliminary implementation

We, jointly with Prof. Eero Simoncelli from New York University, have made a first attempt to achieve a practical method based on the multi-GSM model. The estimation method chosen includes both approximations and simplifications of the theoretical results presented in previous section, which were actually worked on parallel. In particular, the covariance matrices estimation has been done as a first separated step, via a standard iterative clustering using a Gaussian mixture with additive Gaussian noise. Starting from some guess covariance matrices, every observation is ML-classified. At each step of the iterations, sample covariances of each class are computed and possible negative eigenvalues are corrected after having subtracted the noise covariance.  $P_k$ 's are simply estimated iteratively as the number of elements of each resulting cluster. Note that this approximation has disregarded the effect of the scale in the estimation, and that it applies a simple rule of "belonging to a class" for

the iterative estimation, instead of weighting by  $P(k|y_m)$ . It also applies the same "sharp" rule for the coefficient estimation, where each cluster of subband coefficients is processed independently through standard BLS-GSM using Jeffrey's prior for all clusters <sup>§</sup>. Despite the success of this preliminary implementation (see results in 5.1), we believe that applying the estimation technique of previous section, though more computationally costly, may significantly improve the results presented here.

#### 4. DEBLURRING-BY-DENOISING

We face the problem of image restoration, where prior to noise addition, the signal has been degraded by (typically) a low-pass filter. The observation is modelled as follows:

$$y(n, m) = h(n, m) * x(n, m) + w_0(n, m). \quad (9)$$

where  $h$  is the convolution kernel (assumed known) applied to the original image  $x$ , usually called Point Spread Function, PSF. Noise vector  $w_0$  is Gaussian of known (but arbitrary) power spectral density. The same BLS-GSM local approach used in<sup>3</sup> for image denoising was used in<sup>24</sup> for image restoration. However, in this case, the fact of having a local description of image and degradation caused problems when trying to compensate for certain convolution kernels (see a detailed explanation in<sup>5, 14</sup>). A much more flexible approach consists of first globally pre-compensating the blurring kernel (with a certain regularization in the inverse filtering), and then doing the denoising of the resulting image, see Fig. 2. This deconvolution strategy was proposed more than ten years ago,<sup>11</sup> and it has been followed by many authors.<sup>12, 13, 25-27</sup>

We can express the observation model of Eq. 9 in the frequency domain as

$$Y_0(u, v) = H(u, v)X(u, v) + W_0(u, v),$$

where<sup>¶</sup>  $H$  is the Fourier transform of the blur kernel  $h$ . Let  $G$  be the filter applied in step 1 (here in after, prefilter). If we wanted to compute the optimal prefilter in MSE sense,  $G_{opt}$ , we should couple both steps:

$$G_{opt} = \arg \min_G \{\mathbb{E}\{\|X - f(GY_0)\|^2\}\}, \quad (10)$$

where  $f$  represents here the denoising operator (step 2), so  $f(GY_0)$  represents the output of the whole process. However, to compute the optimal prefilter  $G$  we need an explicit error model for the denoising step.<sup>5, 14</sup> To obtain such an error model for a non-linear adaptive denoising method is a major challenge. In the absence of a globally optimal solution, next we follow a marginally optimal procedure which is, nevertheless, efficient and robust.

##### *Step 1: Global blur compensation*

Instead of estimating an optimal filter here we reduce the degrees of freedom from  $N \times N$  (image and filter size) to only one, corresponding to the regularization parameter,  $\alpha$ . We use, as in other works<sup>12, 25, 27</sup> a generalized Wiener filter:

$$G = \frac{H^*}{|H|^2 + \alpha \cdot \left(\frac{P_{w_0}}{P_x}\right)},$$

where  $P_{w_0}$  and  $P_x$  are the power spectral densities of noise and signal respectively. This regularized inversion compensates for the filter phase. The balance between noise and blur suppression is controlled by the regularization parameter  $\alpha$ . The higher is  $\alpha$ , the higher is the overall linear distortion, but the less noisy is the estimation.  $\alpha = 1$  yields the Wiener filter, which is a far too high value when using typical images and non-linear denoising methods in the second step, as we show in subsection 4.0.1.

---

<sup>§</sup>The idea of classifying the neighborhoods according to some features, and then applying to each class the standard BLS-GSM model had already been proposed by Tan Shan in a different context, see acknowledgments.

<sup>¶</sup>For notational simplicity from here on we drop the frequency indices (u,v).

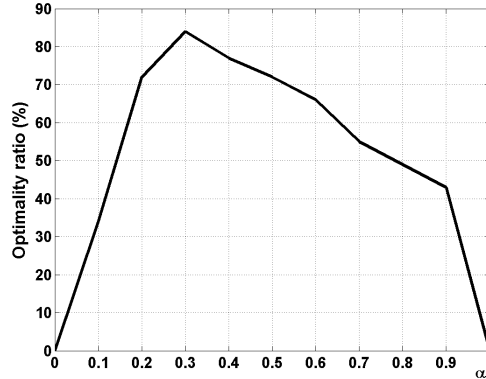


Fig. 3: Optimality ratio of the regularization parameter  $\alpha$ . See text for details.

**Step 2: Local denoising**

After step-1, the observation yields:

$$Y_1 = GY_0 = GHX + GW_0 = H_r X + W_1.$$

Because  $G$  has been optimized in the first step, no further attempt to correct the remaining linear distortion  $H_r = GH$  is made, and what is left is to denoise the pre-filtered image.

**4.0.1. Optimizing the pre-filtering parameter**

Based on the typical images power spectrum,<sup>15</sup> we have used  $\hat{P}_x = \kappa/f^2$ , where  $f$  represents absolute spatial frequency and  $\kappa$  is an arbitrary constant. In order to force the model to be consistent with the estimated original variance ( $\hat{\sigma}_x^2 = \sigma_{y_0}^2 - \sigma_{w_0}^2$ ), we have used:

$$\hat{\sigma}_x^2 = \int_{f>0} |H(f)|^2 \hat{P}_x f df = \int_{f>0} |H(f)|^2 \frac{\kappa}{f} df,$$

hence  $\kappa$  is set to:

$$\kappa = \hat{\sigma}_x^2 / \int_{f>0} |H(f)|^2 / f df.$$

Despite its simplicity, this model is effective and, in contrast to other methods (such as<sup>24</sup>), the estimation error in  $P_x$  does not seem to play a critical role for the overall performance.

We have empirically computed the regularization parameter  $\alpha$ , by training with a set of standard test images (*House, Cameraman, Barbara, Boat, and Lena*), using Gaussian convolution kernels and white Gaussian noise of 0.4 to 3.2 and 1 to 16 standard deviations, respectively (in a geometric scale). For each combination of degradation parameters we have computed the optimal  $\alpha$  through golden search. Next, we have set a quality decrease tolerance interval around it by finding the extremes of the intervals as those  $\alpha$  values providing a decrease in performance of 0.05 dB with respect to the optimum. Then we have chosen the common  $\alpha$  value that provides the highest proportion of experiments within the quality decrease tolerance intervals. Fig. 3 shows the optimality ratio for each alpha value, showing a peak for  $\alpha = 0.3$ , for which value around 84% of the experiments provided quasi-optimal results. We have also tested the training method with very different PSFs<sup>5</sup> (such as a uniform kernel), and the maximum of the histogram was still very close to 0.3, although the percentage dropped slightly (73% for the uniform kernel). We have also tested that the results are fairly independent on the particular images chosen for the training, by comparing these results with those obtained by excluding each time the image to be processed in the training set.



$\sigma = 10$			
Image/Meth	Original	SpaceVar	Multi-GSM
Lena	35.23	35.42	35.68
Barbara	33.13	34.01	34.47
Boat	33.48	33.57	33.69
House	35.32	35.54	35.99
Peppers	34.06	34.18	34.27
$\sigma = 25$			
Image/Meth	Original	SpaceVar	Multi-GSM
Lena	31.26	31.41	31.77
Barbara	27.81	28.81	29.78
Boat	29.27	29.32	29.48
House	31.59	31.71	31.99
Peppers	29.45	29.46	29.71

**Table 1.** Denoising performance in terms of peak signal-to-noise ratio (PSNR),  $10 \log_{10}(255^2/\sigma_e^2)$  in dB, where  $\sigma_e^2$  is the mean square error. for the three GSM methods, applied to a set of five test images, for  $\sigma = 10$  and  $\sigma = 25$ .

## 5. RESULTS

### 5.1. Denoising

We have used the translation-invariant Haar pyramid (TIHP) as image representation.<sup>5,6,14</sup> This representation has the advantage of being much faster than the full steerable pyramid used in.<sup>3</sup> However, this makes a difference also in the denoising quality, which usually improves for poorly textured images (like *House*) and decreases for texture-rich images (like *Barbara*). We chose the number of scales to 4 and 5 for  $256 \times 256$  and  $512 \times 512$  images respectively. We used a  $3 \times 3$  GSM neighborhood around each coefficient (same as in<sup>3</sup>) for the spatially adaptive method, and a  $7 \times 7$  neighborhood for the multi-GSM method. The block size for the spatially adaptive method was 32. Both the GSM neighborhood and the block size, are the same in all pyramid levels, except for those subbands with a size smaller than the block, for which the whole subband is taken. We compare results with the standard BLS-GSM method using the same parameters and representation, and with the method providing the current state-of-the-art in denoising performance measure as quadratic error on test images corrupted by simulated additive white Gaussian noise.<sup>8</sup>

Figure 4 shows a visual comparison for the images *Barbara* and *House* (cropped to  $128 \times 128$  for favoring the visibility of the artifacts), for the three different versions of the GSM methods implemented, using simulated white Gaussian noise of  $\sigma = 25$  and  $\sigma = 10$ . We can see the progressive and significant improvement (removing isolated basis functions, more natural edge appearance, a better texture recover, etc.) when introducing the spatial adaptation, first, and even more when using multiple GSM's per subband. Figure 5 shows the corresponding peak signal-to-noise ratio (PSNR) values (defined as  $10 \log_{10}(255^2/\sigma_e^2)$  in dB, where  $\sigma_e^2$  is the mean square error) graphically, and compare them to the results from Dabov et al.<sup>8</sup>

In Table 1 we include the denoising performance in PSNR terms, for the three versions of the BLS-GSM method, applied to a set of five test images, for  $\sigma = 10$  and  $\sigma = 25$ .

### 5.2. Deblurring

In Table 2 we show averaged Increment in Signal-to-Noise Ratio (ISNR) results of applying our deblurring method using the spatially variant BLS-GSM denoising algorithm to the training data set (see Section 4.0.1 for details). As we can see, the results exhibit a significant improvement for the whole range of tested blur and noise degradations.

In Table 3 we show the results for six deconvolutions experiments with *House*, *Cameraman* (using THP) and *Barbara* (using FSP) comparing our method with the previous BLS-GSM restoration,<sup>24</sup> and with those which, to the best of our knowledge, represent the current state-of-the-art.<sup>27,28</sup> In the first two columns we have



**Fig. 4.** Visual denoising results using the three compared version of the BLS-GSM denoising method implemented using the translation invariant Haar pyramid (TIHP). For each  $2 \times 2$  block, up left is the noisy simulated observation; up right is the result with the original BLS-GSM method<sup>3</sup>; bottom left is the result using the spatially adaptive version<sup>4</sup>; bottom right is the result using the new multi-GSM model presented here. Upper half correspond to  $\sigma = 10$  and lower half to  $\sigma = 25$ .

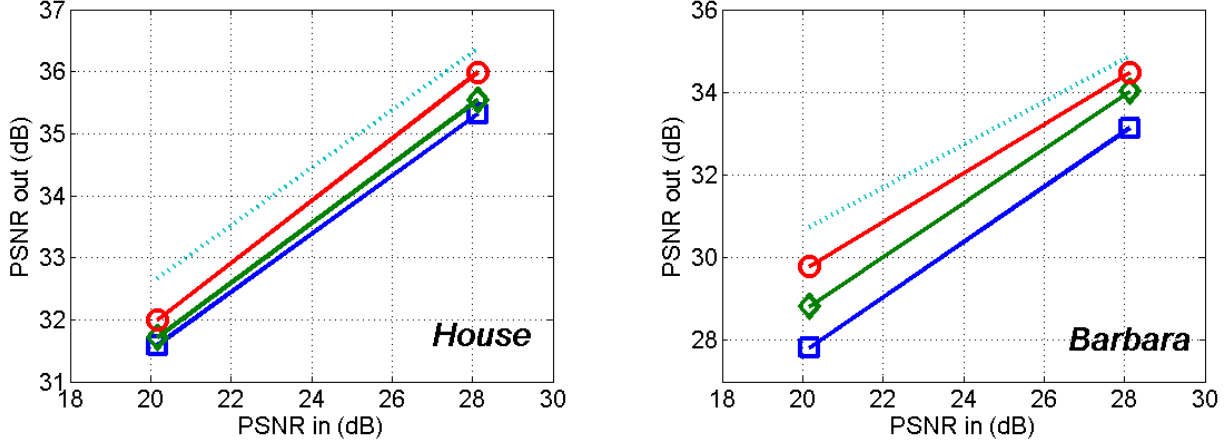


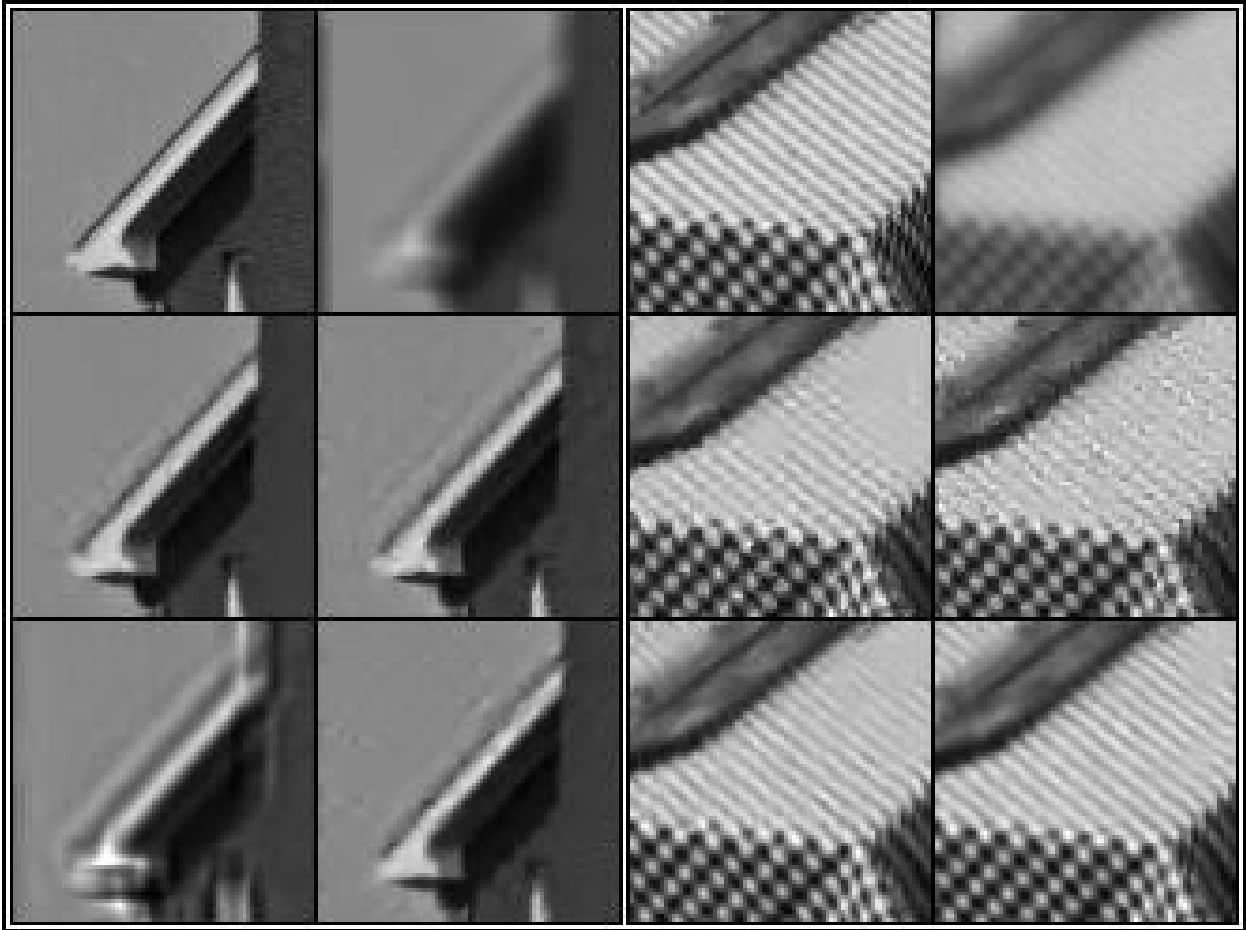
Fig. 5.  $PSNR_{out}$  vs.  $PSNR_{in}$  denoising performance for different BLS-GSM methods, using the TIHP, for 2 images. Squares represent the original BLS-GSM<sup>3</sup>; diamonds the space variant version<sup>4</sup>; circles our new preliminary results using a multi-GSM model. Dotted line draws current state of the art<sup>8</sup>

		NOISE					
		$\sigma$	1	2	4	8	16
B	0.4	2.86	1.86	2.81	5.03	7.63	
L	0.8	6.65	4.84	3.73	3.92	5.85	
U	1.6	3.87	3.18	2.72	3.01	4.73	
R	3.2	3.21	2.85	2.56	2.66	3.87	

Table 2. Averaged results as Increment of Signal-to-Noise Ratio (ISNR), in dB, of the training data set (see sec. 4.0.1 for details).

Blur	$PSF 1$		$PSF 2$	$PSF 3$	$PSF 4$	$PSF 5$
$\sigma_w \Rightarrow$	$\sqrt{2}$	$\sqrt{8}$	$\sqrt{0.308}$	7	2	8
Method	<i>HOUSE</i>					
28	8.47	6.63	<b>10.71</b>	4.22	<b>4.49</b>	4.76
24	8.46	6.93	-0.44	<i>4.37</i>	<i>4.34</i>	5.98
27	<b>9.05</b>	<b>7.64</b>	<b>10.71</b>	<b>5.10</b>	4.03	<b>6.13</b>
Ours	<i>8.64</i>	<i>7.03</i>	9.04	4.30	4.11	<i>6.02</i>
Method	<i>CAMERAMAN</i>					
28	<i>7.46</i>	5.24	<i>8.16</i>	2.84	<i>3.18</i>	3.65
29	6.93	4.88	7.59	<b>2.94</b>	-/-	-/-
30	7.40	5.15	8.10	<i>2.85</i>	-/-	-/-
24	6.84	5.29	-1.61	2.56	2.83	3.81
27	<b>8.25</b>	<b>6.34</b>	<b>8.57</b>	2.56	3.05	<b>5.15</b>
Ours	7.45	<i>5.55</i>	7.33	2.73	<b>3.25</b>	<i>4.19</i>
Method	<i>BARBARA</i>					
28	3.76	1.99	3.98	0.9	0.92	2.55
24	5.70	3.28	-0.27	1.44	0.95	4.91
27	5.73	3.01	4.88	1.58	0.91	4.04
Ours	<b>6.85</b>	<b>3.80</b>	<b>5.07</b>	<b>1.94</b>	<b>1.36</b>	<b>5.27</b>

Table 3. Performance comparison with other state of the art methods in terms of Increment of Signal-to-Noise ratio (ISNR), in dB. First row shows the blurring kernel used (see text for details), and the second denotes noise standard deviation. Best results are highlighted.



**Fig. 6.** Visual comparison results on *House* and *Barbara* images cropped to  $80 \times 80$  pixels (starting in pixel [1,78] and [1,221] respectively, [column, row]). From left to right and top to bottom (House / Barbara PSNR in dB): Original; Degraded imaged (24.10 / 23.33) using PSF2 &  $\sigma_w = \sqrt{0.308}$  for *House* and PSF1 &  $\sigma_w = \sqrt{2}$  for *Barbara* ; Results for<sup>27</sup> (34.76/29.05); Results for<sup>28</sup> (34.76/27.09); Results of standard BLS-GSM<sup>24</sup> with FSP (23.68/29.02); Our new method (House with TIHP, *Barbara* with FSP) (33.11/30.17).

replicated the experiments from.<sup>26</sup> The kernel used (PSF1) is  $h_{i,j} = (1 + i^2 + j^2)^{-1}$ , for  $i, j = -7 \dots 7$ . The next column reproduces the experiment in,<sup>11, 25</sup> which uses a  $9 \times 9$  uniform kernel (PSF2). The fourth column uses a  $5 \times 5$  separable kernel (PSF3) with coefficients  $[1, 4, 6, 4, 1]/16$  as in.<sup>31</sup> The last two columns use Gaussian kernels with 1.6 and 0.4 standard deviation (PSF4 and PSF5 respectively). The noise standard deviations are shown in the second row of the table. The results show that<sup>27</sup> performs the best in a 44.4% of the experiments,<sup>28 + 29</sup> in a 16.7% and the proposed method in a 38.9% of the experiments. Comparing with the BLS-GSM restoration original method,<sup>24</sup> the new method provides a very significant improvement for all experiments, especially for the uniform degradation (PSF2), where the local BLS-GSM approach completely fails.

In Fig. 6 we show a visual comparison among our method, the state-of-the-art methods<sup>27, 28</sup> and BLS-GSM standard restoration.<sup>24</sup> For *Barbara*, the degradation used is PSF1 with  $\sigma = \sqrt{2}$ . In this case our method provides the best result, both visually and in PSNR terms. Particularly, the diagonal lines of the table cloth are better recovered by our method than by our competitors. For the *House* image we have used PSF2 (uniform kernel) with  $\sigma = \sqrt{0.308}$ . Here we can see how the previous BLS-GSM restoration method<sup>24</sup> completely fails, whereas the new version provides results competitive to the state-of-the-art.<sup>27, 28</sup>

## 6. CONCLUSIONS

In this work we have presented two different strategies for increasing the power of the Bayes Least Squares - Gaussian Scale Mixtures (BLS-GSM) denoising method. First one consists of making the covariance of each GSM spatially adaptive, for each subband. This changes results in a significant performance improvement, especially for images with a strong component of spatially variant texture. Second proposed method is more sophisticated, and assumes a mixture of GSMs for each subband. This allows to take advantage of non-locally redundant features of the image, understanding them as those that occur repeatedly through the image, but not in a spatially localized way. Evolving from local to non-local adaptation means an important enrichment of the model capabilities. On the theory side, we have derived an iterative GEM-like estimation strategy for the model parameters. In practical terms, we have just made a preliminary, simplified and approximated model, which, though it is supposed to behave worse than a mature implementation, it is already significantly outperforming our previous results, and getting close to current state of the art. Thus, we have good reasons to believe that the proposed model may have a practical potential impact <sup>||</sup>.

We have also presented an empirical method that allows a robust and efficient image deblurring, by first compensating for the blurring kernel and then doing denoising on the resulting image. Although this strategy is by no means new, our version has proven in the experiments to behave extremely well, both visually, in terms of robustness and in MSE terms, compared to current state of the art. The deblurring results could even improve if, instead of having used our spatially adaptive version of BLS-SM we had used the very last version.

In summary, this paper demonstrates that the ceiling performance of models based on local Gaussian scale mixtures has not been reached yet, and there are some promising perspectives for the short term future.

## ACKNOWLEDGMENTS

Besides the direct participation of Prof. Eero Simoncelli in the search of a practical approach for applying the multi-GSM model, we want also to thank Dr. Tan Shan, from Xidiang University, China, for interesting discussions on the coefficient clustering and use of multiple GSMs ideas.

## REFERENCES

1. E. P. Simoncelli, "Statistical models for images: Compression, restoration and synthesis," in *Proc 31st Asilomar Conf on Signals, Systems and Computers*, pp. 673–678, IEEE Computer Society, (Pacific Grove, CA), November 1997. Available from <http://www.cns.nyu.edu/~eero/publications.html>.
2. A. B. Lee, D. Mumford, and J. Huang, "Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model," *International Journal of Computer Vision* **41**(1/2), pp. 35–59, 2001.
3. J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Proc.* **12**, pp. 1338–1351, November 2003.
4. J. A. Guerrero-Colon and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *IEEE Int'l Conf. on Image Proc.*, **I**, pp. 105–108, Genoa, Italy, September 2005.
5. J. A. Guerrero-Colon and J. Portilla, "Deblurring-by-denoising using spatially adaptive Gaussian scale mixtures in overcomplete pyramids," in *IEEE Int'l Conf. on Image Proc.*, pp. 625–628, Atlanta, GA (USA), October 2006.
6. J. Portilla, "Full blind denoising through noise covariance estimation using Gaussian scale mixtures in the wavelet domain," in *IEEE Int'l Conf. on Image Proc.*, IEEE, ed., pp. 1217–1220, Singapore, October 2004.
7. A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising.," in *IEEE Conf. on Comp. Vision and Pattern Rec.*, **2**, pp. 60–65, 2005.
8. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," in *Proc. SPIE Electronic Imaging: Algorithms and Systems V*, **6064A-30**, (San Jose, CA, USA), January 2006.

---

<sup>||</sup>Although probably it will never be, in its current form, a very fast method, due to its considerable computational load: current implementation is around 5 to 20 times slower than BLS-GSM, depending on the parameters chosen.

9. M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Proc.* **15**, pp. 3736–3745, December 2006.
10. K. Hirakawa and T. W. Parks, "Image denoising using total least squares," *IEEE Trans. Image Proc.* **15**, pp. 2730–2742, September 2006.
11. M. R. Banham and A. K. Katsaggelos, "Spatially adaptive wavelet-based multiscale image restoration," *IEEE Trans. Image Proc.* **5**, pp. 619–634, April 1996.
12. R. Neelamani, H. Choi, and R. G. Baraniuk, "Wavelet-domain regularized deconvolution for ill-conditioned systems," in *IEEE Int'l Conf. on Image Proc.*, **1**, pp. 204–208, (Kobe, Japan), October 1999.
13. J. Kalifa and S. Mallat, "Mini-max restoration and deconvolution," in *Bayesian inference in wavelet based methods*, Springer, 1999.
14. J. A. Guerrero-Colon, L. Mancera, and J. Portilla, "Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids," *IEEE Trans. Image Proc.*, submitted in May 2007.
15. E. R. Kretzmer, "The statistics of television signals," *Bell System Tech. J.*, vol. 31, pp. 751–763, 1952.
16. D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *J. Opt. Soc. Am. A* **4**(12), pp. 2379–2394, 1987.
17. J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans Sig Proc* **41**, pp. 3445–3462, December 1993.
18. R. W. Buccigrossi, *Compression and Segmentation of Images Using an Inter-Subband Wavelet Probability Model*. PhD thesis, University of Pennsylvania, Department of Computer and Information Science, Philadelphia PA, June 1999.
19. J. Portilla and E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *International Journal of Computer Vision* **40**(1), pp. 49–70, 2000.
20. D. Andrews and C. Mallows, "Scale mixtures of normal distributions," *J. Royal Stat. Soc.* **36**, pp. 99–102, 1974.
21. M. J. Wainwright and E. P. Simoncelli, "Scale mixtures of Gaussians and the statistics of natural images," in *Adv. Neural Information Processing Systems*, S. A. Solla, T. K. Leen, and K.-R. Müller, eds., **12**, pp. 855–861, MIT Press, (Cambridge, MA), May 2000.
22. S. Lyu and E. P. Simoncelli, "Statistical modeling of images with fields of Gaussian scale mixtures," in *Adv. Neural Information Processing Systems (NIPS)*, **19**, Adv. Neural Information Processing Systems, May 2007.
23. D. K. Hammond and E. P. Simoncelli, "Image denoising with an orientation-adaptive Gaussian scale mixture model," in *IEEE Int'l Conf. on Image Proc.*, IEEE Computer Society, (Atlanta, GA), October 2006.
24. J. Portilla and E. P. Simoncelli, "Image restoration using Gaussian scale mixtures in the wavelet domain," in *IEEE Int'l Conf. on Image Proc.*, **2**, pp. 965–968, (Barcelona, Spain), September 2003.
25. R. Neelamani, H. Choi, and R. G. Baraniuk, "ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Trans. Signal Proc.* **52**, pp. 418–433, February 2004.
26. A. Jalobeanu, N. Kingsbury, and J. Zerubia, "Image deconvolution using hidden Markov tree modeling of complex wavelet packets," in *IEEE Int'l Conf. on Image Proc.*, **1**, pp. 201–204, 2001.
27. A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-adaptive DCT for denoising and image reconstruction," in *SPIE Electronic Imaging 2006, Image Processing: Algorithms and Systems V*, pp. 6064A–18, 2006.
28. M. Figueiredo and R. Nowak, "A bound optimization approach to wavelet-based image deconvolution," in *IEEE Int'l Conf. on Image Proc.*, **2**, pp. 782–785, 2005.
29. M. Figueiredo and R. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Trans. Image Proc.* **12**, pp. 906–916, August 2003.
30. J. Bioucas-Dias, "Bayesian wavelet-based image deconvolution: a gem algorithm exploiting a class of heavy-tailed priors," *IEEE Trans. Image Proc.* **4**, pp. 937–951, April 2006.
31. J. Liu and P. Moulin, "Complexity-regularized image restoration," in *IEEE Int'l Conf. on Image Proc.*, **1**, pp. 555–559, 1998.



## S.2.4. Image denoising using mixtures of Gaussian scale mixtures

- J. A. Guerrero-Colón, E. P. Simoncelli and J. Portilla, "Image denoising using mixtures of Gaussian scale mixtures", IEEE Int'l Conf. on Image Proc., 2008, Submitted.





# IMAGE DENOISING USING MIXTURES OF GAUSSIAN SCALE MIXTURES

Jose A. Guerrero-Colón

Dept. of Comp. Science and A.I.  
Universidad de Granada  
Granada (Spain)  
jaguerrero@decsai.ugr.es

Eero P. Simoncelli

Center for Neural Science, and  
Courant Inst. of Math. Sciences  
New York University (USA)  
eero.simoncelli@nyu.edu

Javier Portilla

Instituto de Óptica  
CSIC  
Madrid (Spain)  
portilla@io.cfmac.csic.es

*The local statistical properties of photographic images, when represented in a multi-scale basis, have been described using Gaussian scale mixtures (GSMs). In that model, each spatial neighborhood of coefficients is described as a Gaussian random vector modulated by a random hidden positive scaling variable. Here, we introduce a more powerful model in which neighborhoods of each subband are described as a finite mixture of GSMs. We develop methods to learn the mixing densities and covariance matrices associated with each of the GSM components from a single image, and show that this process naturally segments the image into regions of similar content. Similar methods can be applied in the presence of additive Gaussian noise, and the resulting fitted model may be used as a prior for Bayesian noise removal. Simulations demonstrate this model substantially outperforms the original GSM model.*

**Index Terms**— Image denoising, Image modelling, Gaussian scale mixture.

## 1. INTRODUCTION

Natural images are highly inhomogeneous: smooth regions are interrupted by features such as texture, edges, lines, corners, etc. This heterogeneity produces highly non-Gaussian statistics, which are readily apparent when the image is transformed from the pixel domain to a linear multi-scale representation. Marginal histograms of coefficients reveal highly kurtotic behavior with heavy tails [1, 2]. In addition, there is a strong non-linear coupling between amplitudes of coefficients at nearby positions, orientations, and scales [3, 4]. A Gaussian scale mixture (GSM) model [5], in which clusters of coefficients are modeled as a product of a Gaussian vector and a positive scaling variable, captures both the amplitude coupling and the kurtotic marginals of wavelet coefficients [6]. Since the subbands are computed through band-pass filtering, it is a reasonable starting point to assume that spatial neighborhoods within a subband share the same local covariance [7]. Better modeling may be obtained by allowing the neighborhoods of the image subbands to have a spatially varying covariance matrix. In previous work, this has been achieved by estimating covariance over a subregion of the image surrounding each neighborhood (known as SVGSM) [8], or by defining the covariances in a local coordinate system rotated to match the local dominant orientation (known as OAGSM) [9].

Several recent denoising methods demonstrate impressive results by exploiting the similarity of neighborhood structures scat-

tered throughout a given image [10, 11, 12]. Here, we introduce a finite mixture of GSM (MGSM) model that can effectively adapt the local covariance in a GSM model to nonlocal image structures. We develop an algorithm for adapting the parameters (in particular, the covariances of each of the components) to the data in a single image. We show first that fitting the model to an image produces a natural segmentation into regions of similar content. We then develop a Bayesian Least Squares estimator for removing additive Gaussian noise. We demonstrate through simulations that the resulting denoising performance is substantially better than that of a single GSM model (as reported in [7]), and nearly as good as the best results reported in recent literature.

## 2. MODELLING NON-LOCAL STATISTICS: MGSM

The modeling capabilities of a GSM can be significantly improved by estimating the covariance matrices locally [8, 9], because the neighborhoods over which the estimation is performed are more likely to be statistically similar. But this reduces the number of neighborhoods used in the estimate (thus increasing the estimation error), and still does not *guarantee* that they are not contaminated by neighborhoods drawn from different statistical sources (e.g., when estimated near a texture boundary). Instead, we would like to identify the dominant statistical structures occurring throughout the image, and then compute the contribution of each of these identified “modes” to the contents of each local neighborhood. We achieve this by introducing a *mixture of Gaussian scale mixtures* (MGSM) model. As with the GSM model, we write a (vectorized) coefficient neighborhood as:

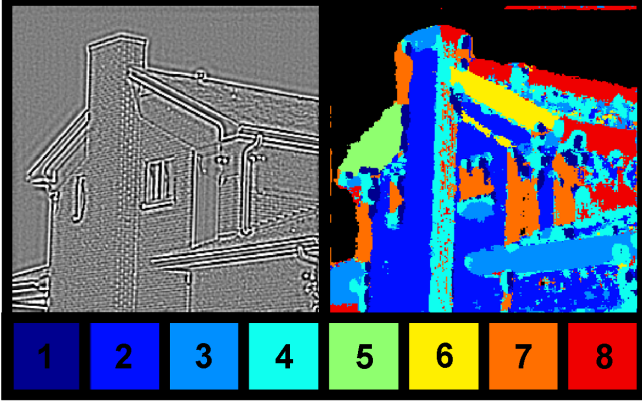
$$\mathbf{x} = \sqrt{z}\mathbf{u},$$

but unlike a simple GSM, we assume a hidden discrete index variable  $k \sim P_k$  is selected for each neighborhood and determines the mixing density of the scalar multiplier  $p(z|k) = p_k(z)$ , as well as the covariance matrix  $\mathbf{C}_k$  of zero-mean Gaussian vector  $\mathbf{u}$ . We assume  $\mathbf{u}$  and  $z$  are independent when conditioned on  $k$ . According to this, the density of the modelled neighborhood vector is:

$$\begin{aligned} p(\mathbf{x}) &= \sum_{k=1}^K P_k p(\mathbf{x}|k) = \sum_{k=1}^K P_k \int_z p(\mathbf{x}|k, z) p_k(z) dz \\ &= \sum_{k=1}^K P_k \int_z \frac{\exp(-\mathbf{x}^T(z\mathbf{C}_k)^{-1}\mathbf{x}/2)}{(2\pi)^{N/2}|z\mathbf{C}_k|^{1/2}} p_k(z) dz. \end{aligned} \quad (1)$$

In this paper, we model neighborhoods for each subband separately. For each subband, and assuming a neighborhood size  $N$  and number of mixture components  $K$ , the parameters of the MGSM model are: (a) the  $K$  probability masses  $P_k$ ; (b) the  $K$  mixing scale densities

JAGC was supported by AP2005-2969 FPU contract and TEC2006/13845/TCM grant, both funded by the Ministerio de Educación y Ciencia (Spain). EPS was supported by the Howard Hughes Medical Institute. JP was supported by TEC2006/13845/TCM grant and the Ramón y Cajal Program, both funded by the Ministerio de Educación y Ciencia (Spain).



**Fig. 1.** Image subband labelled according to dominant MGSM component. Left: a non-oriented steerable pyramid subband of the *House* image. Right: color-coded indication of dominant component in an MGSM model with 8 components and  $7 \times 7$  neighborhoods. Black pixels indicate neighborhoods with energy below a threshold.

$p_k(z)$ ; and (c) the  $K$  covariance matrices  $\mathbf{C}_k$ . For a given image, we estimate these parameters by maximizing the likelihood of the data given the model. The procedure is an iterative EM-like algorithm, and a very similar version of it is described in Section 3 for the case when there is additive Gaussian noise in the observed image.

The MGSM model is able to capture similarities in neighborhood structures, even if they are non-adjacent. This is illustrated in Fig. 1, which shows a non-oriented image subband together with the result of fitting the MGSM model with  $K = 8$  and a  $7 \times 7$  GSM neighborhood. The color of each pixel identifies the dominant GSM mixture component for the neighborhood centered at that position. Specifically, we estimated the model parameters and then chose, for each neighborhood, the index  $k$  maximizing the posterior  $P(k|\mathbf{x})$ . Some mixture components can be seen to represent edges of a given orientation (components 3, 5, 6, 7 and 8), some represent homogeneous texture regions (2 and 4), and one captures corners (1). We find it remarkable that this model, which does not include any explicit definition of features or attributes, can perform an automatic segmentation of the image contents into these feature classes.

### 3. IMAGE DENOISING USING MGSM

The MGSM model provides a useful substrate for the application of denoising, where it can be used as a prior model in a Bayesian estimation scheme. Consider an image corrupted by additive Gaussian noise of known covariance (not necessary white). As with many other methods, we estimate the original image by transforming the noisy observation into an overcomplete multi-scale representation, denoising each subband, and then inverting the transform. We model the noisy neighborhoods of each subband as a mixture of Gaussian scale mixtures, plus a Gaussian noise term:

$$\mathbf{y} = \sqrt{z}\mathbf{u} + \mathbf{w},$$

where  $\mathbf{w}$  is a zero mean Gaussian noise vector with covariance matrix  $\mathbf{C}_w$ . The density of the observed neighborhood vector conditioned on  $z$  and  $k$  is Gaussian (zero-mean), with covariance  $\mathbf{C}_{\mathbf{y}|z,k} = z\mathbf{C}_k + \mathbf{C}_w$ . Hence, the density of the observed vector  $\mathbf{y}$

is:

$$p(\mathbf{y}) = \sum_{k=1}^K P_k \int_z \frac{\exp(-\mathbf{y}^T(z\mathbf{C}_k + \mathbf{C}_w)^{-1}\mathbf{y}/2)}{(2\pi)^{N/2}|z\mathbf{C}_k + \mathbf{C}_w|^{1/2}} p_k(z) dz.$$

#### 3.1. BLS Coefficient estimation

As in previous work, we compute the Bayes least squares (BLS) estimate (also known as the MMSE) of each coefficient given the surrounding block. The BLS estimate is the mean of the posterior distribution,  $p(\mathbf{x}|\mathbf{y})$ , which may be written in this case as:

$$\hat{\mathbf{x}}_{BLS}(\mathbf{y}) = \sum_{k=1}^K P(k|\mathbf{y}) \int_z \mathbb{E}\{\mathbf{x}|\mathbf{y}, k, z\} p_k(z|\mathbf{y}) dz.$$

The value of the integral over  $z$  is the BLS estimate for a single GSM model (as found in [7]), and the full estimate is a weighted sum of these over the  $K$  GSM mixture components. The integrals are relatively simple to compute since the embedded expectation is taken over a conditionally Gaussian density, and thus is just the Wiener solution,  $\mathbb{E}\{\mathbf{x}|\mathbf{y}, k, z\} = z\mathbf{C}_k(z\mathbf{C}_k + \mathbf{C}_w)^{-1}\mathbf{y}$ .

#### 3.2. Parameter estimation under additive noise

We adaptively estimate the parameters for each noisy subband, taking the set of all neighborhoods,  $\{\mathbf{y}_m; m = 1 \dots M\}$ , to represent a set of independent samples from the noise-corrupted MGSM density. We estimate the parameters that maximize the global log-likelihood expression of the observations:

$$L(\{\mathbf{y}_m\}) = \sum_{m=1}^M \log \left[ \sum_{i=1}^K P_{k_i} \int_z p(\mathbf{y}_m|k_i, z) p_{k_i}(z) dz \right]$$

In order to simplify the numerical optimization problem, we reduce the hidden multiplier densities  $p_k(z)$  to a finite number of discrete values. We maximize the likelihood using an iterative coordinate ascent method, optimizing each set of parameters in turn. The ascent steps are:

- *Discrete probabilities  $P_k$ :*

$$P_k^{(n+1)} = P_k^{(n)} \frac{1}{M} \sum_{m=1}^M \frac{p(\mathbf{y}_m|k)^{(n)}}{\sum_{j=1}^K p(\mathbf{y}_m|j)^{(n)} P_j^{(n)}}, \forall k \quad (2)$$

- *Mixing scale densities  $p_k(z)$ :*

$$p_k(z)^{(n+1)} = \frac{p_k(z)^{(n)}}{M P_k^{(n)}} \sum_{m=1}^M \frac{p(\mathbf{y}_m|k, z)^{(n)} P(k|\mathbf{y}_m)^{(n)}}{p(\mathbf{y}_m|k)^{(n)}}, \forall k \quad (3)$$

Note that, in contrast to original GSM model where Jeffrey's non-informative prior was used for the mixing density, this model estimates it directly from the data, as in [13].

- *Covariance matrices  $\mathbf{C}_k$ :* For the noisy case we have not found an efficient ML expression for  $\mathbf{C}_k$ . Instead we have used a consistent estimator which gives rise to the following updating rule:

$$\mathbf{C}_k^{(n)} = \left[ \left( \int_z \mathbf{C}_y(k, z)^{(n)} p(z)^{(n)} dz \right) - \mathbf{C}_w \right]_+, \quad (4)$$

where the operator  $[\cdot]_+$  enforces positive definiteness by setting to zero the negative eigenvalues, and

$$\begin{aligned} \mathbf{C}_y(k, z)^{(n)} &= \frac{\sum_{m=1}^M P(k|\mathbf{y}_m)^{(n)} p(z|\mathbf{y}_m, k)^{(n)} \mathbf{y}_m \mathbf{y}_m^T}{\sum_{m=1}^M P(k|\mathbf{y}_m)^{(n)} p(z|\mathbf{y}_m, k)^{(n)}} \\ &\simeq z\mathbf{C}_k + \mathbf{C}_w. \end{aligned}$$

Without loss of generality, we have assumed that  $\int_z z p_k(z) dz = 1$  for all  $k$ , which leads directly to Eq. (4).

The ascent algorithm requires an initial set of parameter values. For both  $P_k$  and  $p_k(z)$  we assume a uniform distribution. The latter density is discretized using the sampling described in [7]. The covariances,  $\mathbf{C}_k$ , are initialized from the image content as follows: The first one,  $\mathbf{C}_1$ , is set to the global sample covariance. The remaining initial guesses for  $\mathbf{C}_k$ ,  $k = 2..K$  are obtained from high energy areas in the subband, selected using a heuristic.

To update the parameters accordingly, we need to compute, among others, the expression  $p(z|\mathbf{y}_m, k)$ . Using Bayes formula:

$$p(z|\mathbf{y}_m, k) = \frac{p(\mathbf{y}_m|k, z)p_k(z)}{p(\mathbf{y}_m|k)}.$$

This requires us to compute  $p(\mathbf{y}_m|k, z)$ , which is a zero-mean Gaussian distribution with covariance  $z\mathbf{C}_k + \mathbf{C}_w$ , and  $p(\mathbf{y}_m|k)$ , which is obtained by numerically integrating  $\int_z p(\mathbf{y}_m|k, z)p_k(z)dz$ . Both of these are involved in the update expressions, as well as  $P(k|\mathbf{y}_m)$ , which is also computed directly using Bayes formula.

## 4. RESULTS AND DISCUSSION

### 4.1. Implementation details

We decomposed the images with the Translation Invariant Haar Pyramid [8] (TIHP), which has 3 orientations, using 4 and 5 scales for  $256 \times 256$  and  $512 \times 512$  images respectively. We used  $K = 10$  GSMs in the mixture and, in contrast to [7], we increased the GSM neighborhood size from  $3 \times 3$  to  $7 \times 7$ . On each iteration of the parameter estimation algorithm, we prune the number of GSMs in the mixture by classifying the samples according to  $P(k|\mathbf{y}_m)$  and deleting those GSMs with few samples assigned (less than N).

We implemented the algorithm in *Matlab*. The current computational cost is quite high: roughly one hour for a  $256 \times 256$  image and six hours for a  $512 \times 512$  image, on a computer with two dual-core 3Ghz Intel Xeon processors and 32 Gbytes of memory. We believe a significant improvement should be possible with a more careful implementation.

### 4.2. Denoising results

We used four standard test images to evaluate our algorithm, known as *Barbara*, *Boat*, *House* and *Peppers*. In Table 1 we present the results of our proposed algorithm, in terms of peak signal-to-noise ratio (PSNR) over a wide range of noise levels. Figure 2 shows a visual comparison for *Barbara* and *House*, two images with different amounts and types of texture, using simulated white Gaussian noise of  $\sigma = 25$ . Results are shown for the BLS-GSM method on TIHP (instead of full steerable pyramid, as in [7]), the SVGSM method [8], and the MGSM. All results are cropped to  $128 \times 128$  to promote the visibility of the artifacts, which are generally informative as to the relative advantages of denoising methods. We can see the progressive and significant improvement (removing isolated basis functions, more natural edge appearance, better texture recovery,

$\sigma_w$ / PSNR	<i>Barbara</i>	<i>Boat</i>	<i>House</i>	<i>Peppers</i>
5/ 34.15	38.02	37.30	39.30	37.75
10/ 28.13	34.45	33.78	36.06	34.30
25/ 20.17	29.80	29.62	32.12	29.74
50/ 14.15	26.19	26.55	28.89	26.37
100/ 8.13	22.79	23.91	25.53	22.99

**Table 1.** Denoising performance expressed as peak signal-to-noise ratio (PSNR) in dB. First column shows the noise standard deviation and the PSNR of the noisy image.

etc.) when introducing the spatial adaptation, and even more when using the mixture of GSMs.

Finally in Fig. 3 we show a graphical comparison of PSNR improvements (averaged over the 4 test images) with respect to the BLS-GSM method [7] on TIHP. Included are results for SVGSM [8], four of the current best published methods (labelled 3DBM [12], FoGSM [14], K-SVD [15], and SA-DCT [11]), and the MGSM model introduced here. The MGSM is seen to provide substantial improvement over the GSM and SVGSM, for all the images and noise levels. The average improvement is roughly 0.70 dB with respect to GSM and 0.40 dB with respect to SVGSM. Relative to other methods our results are significantly better than SA-DCT, and generally better than K-SVD, but they are still worse than FoGSM and 3DBM.

## 5. CONCLUSIONS AND FUTURE WORK

We have presented a denoising algorithm based on a mixture of Gaussian scale mixtures (MGSM). The proposed model provides a flexible and conceptually clean means of capturing non-local redundancy within subbands, while retaining the statistical strength of GSM-based models. In contrast to the original GSM and SVGSM models, there is no need for priors on the hidden scaling variables, as these are systematically ML-estimated for each subband. The proposed model leads to a denoising algorithm with consistent and significant improvement over GSM and SVGSM, and is competitive with methods proposed in recent literature.

We are currently investigating a variety of potential improvements. The pruning of under-utilized mixture components should be refined. The number of mixture models,  $K$ , should be selected adaptively per band, depending on both the content and the number of neighborhoods. And the model should be extended to exploit shared features across subbands, either by enlarging the GSM neighborhoods to include other bands, or sharing mixing variables,  $k$ , across bands. We are also investigating more general issues of adaptive image representation that are inspired by this model.

## 6. REFERENCES

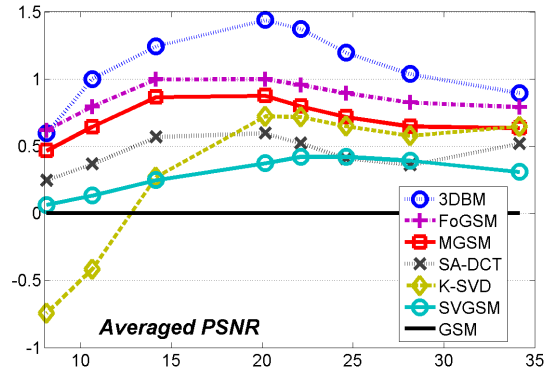
- [1] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Comm.*, vol. COM-31, no. 4, pp. 532–540, April 1983.
- [2] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *J. Opt. Soc. Am. A*, vol. 4, no. 12, pp. 2379–2394, 1987.
- [3] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans Sig Proc*, vol. 41, no. 12, pp. 3445–3462, December 1993.
- [4] E. P. Simoncelli, "Statistical models for images: Compression, restoration and synthesis," in *31st Asilomar Conf. on Signals*,



**Fig. 2.** Example denoised images. We compare three GSM-based methods, all implemented using TIHP, for *House* and *Barbara* images both with  $\sigma = 25$ . Images are cropped to  $128 \times 128$  pixels for visibility (starting in pixel [24,64] and [288,64] respectively [column, row]). For each  $2 \times 2$  panel, from left to right and top to bottom the PSNR values (*House / Barbara*) are: noisy observation (20.17/20.17); BLS-GSM method [7] (31.59/27.81); SVGSM [8] (31.71/28.81); our MGSM result (32.12/29.80).

*Systems and Computers*, Pacific Grove, CA, November 1997, pp. 673–678.

- [5] D. Andrews and C. Mallows, “Scale mixtures of normal distributions,” *J. Royal Stat. Soc.*, vol. 36, pp. 99–102, 1974.
- [6] M. J. Wainwright and E. P. Simoncelli, “Scale mixtures of Gaussians and the statistics of natural images,” in *Adv. Neural Information Processing Systems (NIPS\*99)*, S. A. Solla, T. K.



**Fig. 3.** Comparison of denoising performance. Each curve shows PSNR improvement of one method, relative to GSM [7], as a function of input PSNR, averaged over 4 test images. Methods are: 3DBM [12]; FoGSM [14]; SA-DCT [11]; K-SVD [15]; GSM [7]; SVGSM [8] and our MGSM algorithm with  $K = 10$ . GSM, SVGSM, and MGSM are all implemented on a TIHP.

Leen, and K.-R. Müller, Eds., Cambridge, MA, May 2000, vol. 12, pp. 855–861, MIT Press.

- [7] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, “Image denoising using scale mixtures of Gaussians in the wavelet domain,” *IEEE Trans. Image Proc.*, vol. 12, pp. 1338–1351, November 2003.
- [8] J. A. Guerrero-Colón, L. Mancera, and J. Portilla, “Image restoration using space-variant gaussian scale mixture in over-complete pyramids,” *IEEE Trans. Image Proc.*, vol. 17, no. 1, pp. 27–41, January 2008.
- [9] D. K. Hammond and E. P. Simoncelli, “Image denoising with an orientation-adaptive Gaussian scale mixture model,” in *IEEE Int’l Conf. on Image Proc.*, Atlanta, GA, October 2006.
- [10] A. Buades, B. Coll, and J. M. Morel, “A review of image denoising algorithms, with a new one,” *Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 490–530, July 2005.
- [11] A. Foi, V. Katkovnik, and K. Egiazarian, “Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images,” *IEEE Trans. Image Proc.*, vol. 16, no. 5, pp. 1395–1411, May 2007.
- [12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3-d transform-domain collaborative filtering,” *IEEE Trans. Image Proc.*, vol. 16, no. 8, pp. 2080–2095, August 2007.
- [13] J. Portilla, “Full blind denoising through noise covariance estimation using gaussian scale mixtures in the wavelet domain,” in *IEEE Int’l Conf. on Image Proc.*, IEEE, Ed. Singapore, October 2004, pp. 1217–1220.
- [14] S. Lyu and E. P. Simoncelli, “Statistical modeling of images with fields of Gaussian scale mixtures,” in *Adv. Neural Information Processing Systems 19*, B Schölkopf, J Platt, and T Hofmann, Eds. May 2007, vol. 19, MIT Press.
- [15] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Trans. Image Proc.*, vol. 15, no. 12, pp. 3736–3745, December 2006.

## S.2.5. Deblurring-by-Denoising using spatially adaptive Gaussian scale mixtures in overcomplete pyramids

- J. A. Guerrero-Colón and J. Portilla, "Deblurring-by-Denoising using spatially adaptive Gaussian scale mixtures in overcomplete pyramids", IEEE Int'l Conf. on Image Proc., Atlanta, vol. I, pp. 625-628, Atlanta (GA USA), October 2006.



# DEBLURRING-BY-DENOISING USING SPATIALLY ADAPTIVE GAUSSIAN SCALE MIXTURES IN OVERCOMPLETE PYRAMIDS

Jose A. Guerrero-Colon \* and Javier Portilla †

Visual Information Processing Group  
Dept. of Computer Science and Artificial Intelligence  
Universidad de Granada  
{jaguerrero, javier}@decsai.ugr.es

In a previous work, we presented an extension of the original Bayes Least Squares - Gaussian Scale Mixtures (BLS-GSM) denoising algorithm that also compensated the blur. However, that method had some problems: a) it could not compensate for some blurring kernels; b) its performance depended critically on having an accurate estimation of the original power spectral density (PSD); and c) it could not be easily adapted to a spatially variant description of the image statistics. In this work we propose a two-step restoration method that overcomes these problems by first performing a global blur image compensation, and then applying a spatially adaptive local denoising, in an overcomplete pyramid. Our method is efficient, robust and non-iterative. We demonstrate through simulations that it provides state-of-the-art performance.

**Index Terms**— Image restoration, wavelet transforms

## 1. INTRODUCTION

Image restoration is a classical problem which most often aims to estimate an image given a linearly filtered, noisy version of it. This is a difficult ill-conditioned inverse problem, even if, as in this work, the degradation parameters are assumed known. The Bayesian estimation, by making an explicit formulation of our prior knowledge about natural images, provides a powerful tool for this task. Typically, restoration methods are iterative and global, e.g. [1, 2, 3]. Among global methods, some authors (e.g. [4, 5, 6, 7, 8]) have approached the deconvolution problem by decomposing it in two steps (see Fig. 1): 1) a global blur compensation, and 2) a pure denoising step. The main advantage of this decomposition is that the deconvolution problem is so converted into an easier non-white noise removal problem, at the likely price of losing global optimality.<sup>1</sup> Given the increasingly high performance of current denoising methods, this approach has be-

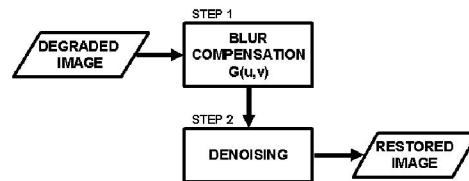


Fig. 1: Restoration Scheme.

come an appealing practical alternative.

A less common path (see e.g. [9, 10]), has been to approach the restoration problem by using local models. A local description of the signal may simplify the formulation and the resolution of the problem, but it also imposes some limitations: how can we compensate for a large blurring kernel using a small neighborhood for the local description? As we will see in Section 2, multi-resolution does not give a complete solution. In [10] we used a local GSM [11] model which provides high performance using an efficient non-iterative method. However this approach has problems inherent to its local nature, and it requires the estimation of the original spectral density from the degraded observation.

In this work we propose a two-step deblurring method based on a recent spatially adaptive local GSM model [12]. By choosing properly the global pre-filtering, our method achieves a close to optimal trade-off between (global) blur compensation and (local) noise removal, improving in performance and robustness with respect to previous methods.

## 2. DEBLURRING USING BLS-GSM

Our observation is modelled as follows:

$$\mathbf{y}_0 = \mathbf{H}\mathbf{x} + \mathbf{w}_0, \quad (1)$$

where  $\mathbf{H}$  is a block-circulant matrix implementing a convolution with  $h$  (the Point Spread Function, PSF, typically a low-pass filter) and the vectorized (following a lexicography order) original signal  $\mathbf{x}$ . The vector  $\mathbf{w}_0$  is Gaussian noise of known PSD. We search for a plausible estimate of  $\mathbf{x}$  given the observation, assuming that both the noise PSD ( $P_{\mathbf{w}_0}$ ) and  $h$  are known.

\*JAGC is supported by 2245-1065 grant contract, funded by Iifa Sistemas S.L. and Ministerio de Defensa (Spain).

†JP is supported by TIC2003-01504 grant, funded by the Ministerio de Educación y Ciencia (Spain) and the "Ramón y Cajal" program.

<sup>1</sup>Note that even if we model noise as white, it will no longer be white after the blur compensation.



## 2.1. One step Deblurring

The power of the BLS-GSM approach for denoising has been widely proved [13, 12]. The same approach has been used for image restoration in [10] with good results. The strength of that restoration method relies on using a local model for spatial neighbors within subbands of overcomplete pyramids. The pyramidal representation, together with the local filtering, results in a very efficient method. However, using a local model imposes also some limitations. Given a small neighborhood, we can only compensate for smoothly varying (in frequency) filters. A fixed neighborhood size for all the scales gives raise to larger effective neighborhoods as we go up in the pyramid. So, thanks to multi-resolution, we can use (and compensate for) smooth filters in frequency at low scales, less smooth at medium scales and peaky at high scales (low frequencies). Such behavior is suitable for many low-pass kernels, like Gaussian or Laplacian. However, there are some practically important kernels (e.g., uniform) whose behavior in frequency is the other way around, i.e., smooth at low frequencies and oscillating at medium/high frequencies. For those cases, the local multi-scale blur compensation fails. Another issue is that [10] requires an accurate estimation of the original PSD ( $P_x$ ). This is a serious problem, especially if we want to make the model spatially adaptive by using blocks, as we did in [12] from [13].

## 2.2. Two-step Deblurring

To preserve the strength of the referred local model while overcoming its problems, we use the scheme depicted in Fig. 1, followed by other authors [4, 6, 7, 8]. Our steps are: 1) global blur compensation and, 2) local denoising of the compensated observation. We have experienced that this approach increases the robustness because it reduces the dependency on the estimated  $P_x$ , and it allows us to take advantage of a state-of-the-art spatially variant local GSM denoising [12].

In order to get an LS-optimal result, we should properly couple both steps. From Eq. 1 we can express the observation model in the frequency domain as

$$Y_0(u, v) = H(u, v)X(u, v) + W_0(u, v) \quad (2)$$

where<sup>2</sup>  $H$  is the Fourier transform of kernel  $h$ . Let  $G$  be the filter applied in step 1. We define  $E_{step-2}^2$ , an estimate of the Mean Square Error (MSE) of the denoising block output w.r.t. the original, as a function of its input, its equivalent PSF, and its noise PSD. The optimal  $G$  is, thus,

$$G_{opt} = \arg \min_G E_{step-2}^2(GY_0, GH, |G|^2 P_{w_0}). \quad (3)$$

At this time we are still working on an MSE model of the BLS-GSM denoising algorithm. Meanwhile, we propose the following simplified approach:

<sup>2</sup>For notational simplicity we will drop the indices (u,v).

### Step 1: Global blur compensation

This step provides a blur-compensated image by applying a global filtering to the observed image. Instead of estimating the optimal filter as a whole (Eq. 3), we reduce the degrees of freedom to only one, called the regularization parameter,  $\alpha$ . We use, as in other works [5, 6] a generalized Wiener filter:

$$G = \frac{H^*}{|H|^2 + \alpha \cdot \left(\frac{P_{w_0}}{P_x}\right)}. \quad (4)$$

The  $\alpha$  parameter controls the noise/blur suppression. Low values result in very noisy intermediate images ( $\alpha = 0$  is just an inverse filter) whereas high values smooth out the high frequencies. For  $\alpha = 1$  the compensation is the Wiener filtering, which provides too smooth global restoration results. Therefore, we will consider  $\alpha$  values lying in the (0, 1) interval.

### Step 2: Local denoising

After step-1, the observation yields,  $Y_1 = H_r X + W_1$ , where  $H_r = GH$  and  $W_1 = GW_0$ . We could approach the problem as another deblurring step, but in practice the residual blur  $H_r$  is small and can be neglected. For the second step we have used our spatially adaptive BLS-GSM denoising algorithm [12], which is robust, non-iterative and provides very high performance. The only difference compared to [13] is that the latter assumes a fixed signal covariance for all GSM neighborhoods in a subband, and the former uses a signal covariance that changes at different regions (blocks). The observed neighborhood centered in position  $(n, m)$  at subband  $j$  is  $\mathbf{y}_1^{j,(n,m)} = \mathbf{x}^{j,(n,m)} + \mathbf{w}_1^{j,(n,m)} = \sqrt{z}\mathbf{u}^{j,(n,m)} + \mathbf{w}_1^{j,(n,m)}$  where  $\mathbf{u}^{j,(n,m)}$  and  $\mathbf{w}_1^{j,(n,m)}$  are zero mean Gaussian vectors with covariance matrices  $\mathbf{C}_{\mathbf{u}}^{j,(n,m)}$  and  $\mathbf{C}_{\mathbf{w}_1}^{j,(n,m)}$ , and  $\sqrt{z}$  is the hidden multiplier controlling the variance of the local signal  $\mathbf{x}^{j,(n,m)}$ . The BLS-GSM method consists of estimating the original central coefficient from the observed neighborhood through:

$$\mathbb{E}\{x_c|\mathbf{y}_1\} = \int_0^\infty p(z|\mathbf{y}_1) \mathbb{E}\{x_c|\mathbf{y}_1, z\} dz \quad (5)$$

The expected value inside the integral is a local Wiener estimate, and the posterior  $p(z|\mathbf{y}_1)$  is computed in terms of  $p(\mathbf{y}_1|z)$  (Gaussian) and the Jeffrey's prior on  $z$  [13, 12].

## 2.3. Pre-filtering parameter estimation

For the  $P_x$  we have used a simple model that falls as  $k/f^2$  in the frequency domain [14]. We have made the model consistent with the estimated original variance ( $\hat{\sigma}_x^2 = \sigma_{y_0}^2 - \sigma_{w_0}^2$ ) by setting  $k = \hat{\sigma}_x^2 / \int_{f>0} (|H(f)|^2 / f) df$ . Despite its simplicity, this model is effective, and, unlike in [10], we have experienced that the global performance does not depend critically on an accurate  $P_x$  estimation.

Some methods [4, 6] use a model to choose the regularization parameter of the blur compensation. Lacking a proper

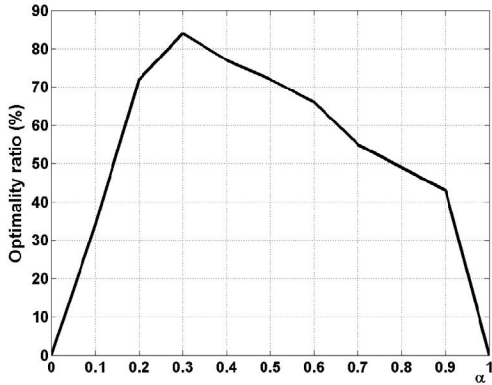


Fig. 2: Optimality ratio for each  $\alpha$  value.

model, we have done it empirically, training with a set of five images (*House*, *Cameraman*, *Barbara*, *Boat*, and *Lena*) and Gaussian PSFs and noise of 0.4 to 3.2 and 1 to 16 standard deviations, respectively (in logarithmic steps). For each blur/noise combination we have computed the optimal  $\alpha$  using the golden section algorithm and an interval around it with the  $\alpha$  values providing a decrease in performance less than 0.05 dBs w.r.t. the optimum. Then we have chosen the common  $\alpha$  value that provides the highest proportion of results within the intervals. Fig. 2 shows the optimality ratio for each alpha value. We have obtained  $\alpha_{opt} = 0.3$ , achieving roughly 84% of optimality. This high value means that in most cases we obtain results very close to optimal. We have also tested that this result is fairly independent on the particular images chosen for the training, so there is no significant *overfitting*.

### 3. RESULTS AND DISCUSSION

For all the experiments, we implemented step 1 in the Fourier domain, using  $\alpha_{opt} = 0.3$ . For step 2 we have used two representations: Trapezoidal Haar Pyramid (THP) and Full Steerable Pyramid (FSP). FSP is an extension of Simoncelli’s original steerable pyramid [13] and THP is an overcomplete version of the Haar wavelet [12]. We have used 3 and 8 orientations for THP and FSP, respectively, and 4 scales. The representation choice depends on the relative amount of texture: texture-rich images are better denoised using FSP, whereas THP is superior for poorly textured images [12]. We have used a  $3 \times 3$  GSM neighborhood and a  $32 \times 32$  local block.

In Table 1 we show, in terms of averaged Increment in Signal-to-Noise Ratio (ISNR), the result of applying our method to the training data set (see Section 2.3). In Table 2 we show a comparison to several state-of-the-art methods with six different experiments over *House*, *Cameraman* (using THP) and *Barbara* (using FSP). We compare with the previous BLS-GSM restoration method [10], and with whom, to the best of our knowledge, is the best-performing method [2]<sup>3</sup>. The comparison can be extended to other methods [15, 3, 4, 6, 8, 16],

<sup>3</sup>We are very grateful to Prof. Figueiredo for providing us with his results.

		NOISE					
		$\sigma$	1	2	4	8	16
B		0.4	2.86	1.86	2.81	5.03	7.63
L		0.8	6.65	4.84	3.73	3.92	5.85
U		1.6	3.87	3.18	2.72	3.01	4.73
R		3.2	3.21	2.85	2.56	2.66	3.87

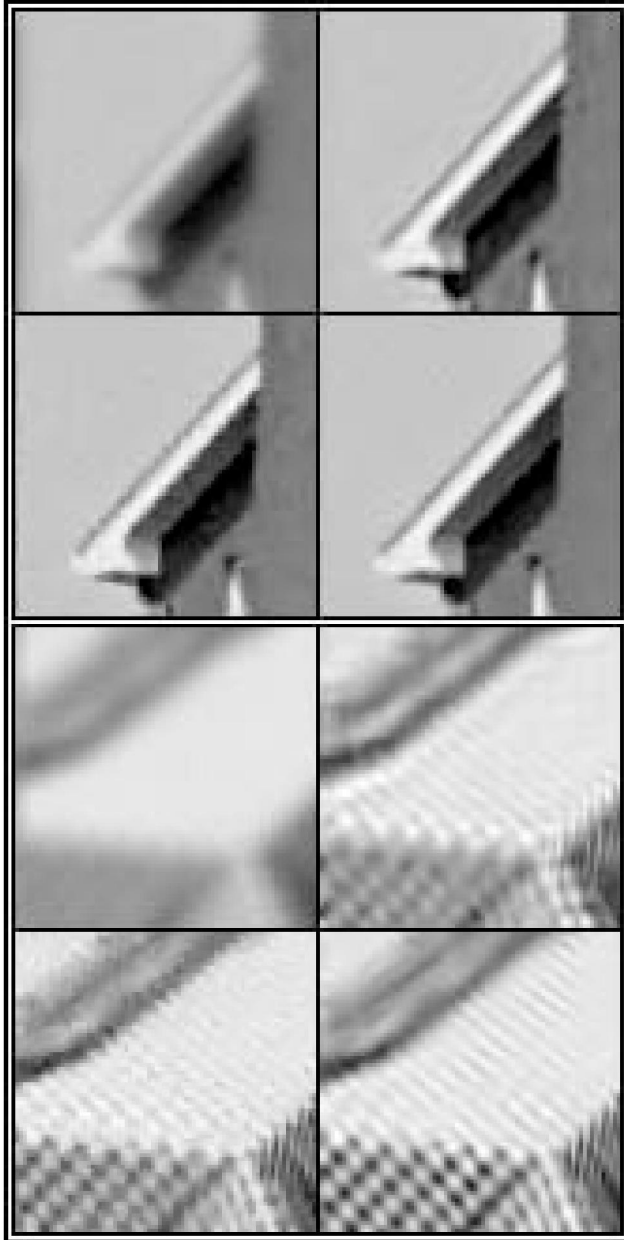
Table 1. Averaged results as Increment of Signal-to-Noise Ratio (ISNR), in dB., of the training data set (see sec. 2.3).

Blur	$\sigma_w \Rightarrow$	PSF 1	PSF 2	PSF 3	PSF 4	PSF 5	
		$\sqrt{2}$	$\sqrt{8}$	$\sqrt{0.308}$	7	2	8
Method	HOUSE						
[2]	8.47	6.63	<b>10.71</b>	4.22	<b>4.49</b>	4.76	
[10]	8.46	6.93	-0.44	<b>4.37</b>	4.34	5.98	
Ours	<b>8.64</b>	<b>7.03</b>	9.04	4.30	4.11	<b>6.02</b>	
Method	CAMERAMAN						
[2]	<b>7.46</b>	5.24	<b>8.16</b>	2.84	3.18	3.65	
[15]	6.93	4.88	7.59	<b>2.94</b>	-/-	-/-	
[3]	7.40	5.15	8.10	2.85	-/-	-/-	
[10]	6.84	5.29	-1.61	2.56	2.83	3.81	
Ours	7.45	<b>5.55</b>	7.33	2.73	<b>3.25</b>	<b>4.19</b>	
Method	BARBARA						
[2]	3.76	1.99	3.98	0.9	0.92	2.55	
[10]	5.70	3.28	-0.27	1.44	0.95	4.91	
Ours	<b>6.85</b>	<b>3.80</b>	<b>5.07</b>	<b>1.94</b>	<b>1.36</b>	<b>5.27</b>	

Table 2. Performance comparison in terms of Increment of Signal-to-Noise ratio (ISNR), in dB. First row shows the blurring kernel used (see text for details), and the second denotes noise standard deviation. Best results are highlighted.

the best results among them have been included in Table 2. In the first two columns we have replicated the experiments in [8]. The kernel used (PSF1) is  $h_{i,j} = (1 + i^2 + j^2)^{-1}$ , for  $i, j = -7 \dots 7$ . Next column reproduces the experiment in [4, 6], that is a  $9 \times 9$  uniform kernel (PSF2). Fourth column uses a  $5 \times 5$  separable kernel (PSF3) with coefficients [1, 4, 6, 4, 1]/16 as in [16]. The last two columns correspond to Gaussian kernels with 1.6 and 0.4 standard deviation (PSF4 and PSF5, respectively). The noise standard deviations are shown in the second row of the table. These results show that [10] performs best in a 5.6% of the cases, [2] is best in a 22.2% and the proposed method is best in a 66.7% of the experiments. Fig. 3 shows a visual comparison of *House* and *Barbara* with the experiments of columns 1 and 3 from Table 2, respectively. As we can see, the roof line in our result is sharper than in the other two. *Barbara* results shows the fall in performance of [10] due to its inability to compensate for the blur of a uniform kernel (see Section 2.1). As visible in the tablecloth, our result recovers much more information than the competitors. The improvement is remarkable, both in visual and PSNR terms.

About computational cost, our method takes about 30 sec. with THP and 70 sec. with FSP in a 2.0GHz PC using Matlab© on a  $256 \times 256$  image, whereas our most direct competitor [2] requires on average a similar time under the same conditions,



**Fig. 3.** Visual comparison results on *House* and *Barbara* images (cropped to  $80 \times 80$ ). From left to right and top to bottom: Degraded (*House*: PSF1 &  $\sigma_w = \sqrt{2}$ ; *Barbara*: PSF2 &  $\sigma_w = \sqrt{0.308}$ ); BLS-GSM [10] with FSP; [2]; Our method (*House* with THP, *Barbara* with FSP). In the same order, Peak Signal-to-Noise ratio in dB. (*House* / *Barbara*): 25.62 / 22.49; 34.08 / 22.22; 34.09 / 26.47; 34.25 / 27.56.

but, unlike our method, its amount of computation depends on the degradation parameters.

Summarizing our method is robust and efficient, achieving state-of-the-art performance over a wide range of types and amounts of degradation. We still expect a significant increase in performance when including an LS-optimal pre-filtering, based on an MSE model for our denoising method, instead of the empirical approach used here.

#### 4. REFERENCES

- [1] R. Molina, J. Mateos, A.K. Katsaggelos, and M. Vega, "Bayesian multichannel image restoration using compound gauss-markov random fields," *IEEE Trans. Image Proc.*, vol. 12, no. 12, pp. 1642–1654, Dec. 2003.
- [2] M. Figueiredo and R. Nowak, "A bound optimization approach to wavelet-based image deconvolution," in *IEEE Int'l Conf on Image Proc.*, vol. 2, pp. 782–785, 2005.
- [3] J. Bioucas-Dias, "Bayesian wavelet-based image deconvolution: a gem algorithm exploiting a class of heavy-tailed priors," *IEEE Trans. Image Proc.*, vol. 15, no. 4, pp. 937–951, Apr. 2006.
- [4] M. R. Banham and A. K. Katsaggelos, "Spatially adaptive wavelet-based multiscale image restoration," *IEEE Trans. Image Proc.*, vol. 5, pp. 619–634, Apr. 1996.
- [5] R. Neelamani, H. Choi, and R. G. Baraniuk, "Wavelet-domain regularized deconvolution for ill-conditioned systems," in *IEEE Int'l Conf on Image Proc.*, Kobe, Japan, vol. 1, pp. 204–208, Oct. 1999.
- [6] R. Neelamani, H. Choi, and R. G. Baraniuk, "ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Trans. Signal Proc.*, vol. 52, no. 2, pp. 418–433, Feb. 2004.
- [7] J. Kalifa and S. Mallat, "Mini-max restoration and deconvolution," in *Bayesian inference in wavelet based methods*. Springer, 1999.
- [8] A. Jalobeanu, N. Kingsbury, and J. Zerubia, "Image deconvolution using hidden markov tree modeling of complex wavelet packets," in *IEEE Int'l Conf on Image Proc.*, vol. 1, pp. 201–204, 2001.
- [9] A.W. Stevenson T.E. Gureyev, Y.I. Nesterets and S.W. Wilkins, "A method for local deconvolution," *Applied Optics*, vol. 42, no. 32, pp. 6488–6494, Nov. 2003.
- [10] J. Portilla and E. P. Simoncelli, "Image restoration using Gaussian scale mixtures in the wavelet domain," in *Proc IEEE Int'l Conf on Image Proc.*, Barcelona, Spain, September 2003, vol. 2, pp. 965–968, IEEE Computer Society.
- [11] D. Andrews and C. Mallows, "Scale mixtures of normal distributions," *J. Royal Stat. Soc.*, vol. 36, pp. 99–102, 1974.
- [12] J.A. Guerrero-Colon and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *IEEE Int'l Conf on Image Proc.* Genoa, Italy, vol. I, pp. 105–108, Sep. 2005.
- [13] J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Proc.*, vol. 12, pp. 1338–1351, Nov. 2003.
- [14] D. L. Ruderman, "The statistics of natural images," *Network: Computation in Neural Systems*, vol. 5, pp. 517–548, 1996.
- [15] M. Figueiredo and R. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Trans. Image Proc.*, vol. 12, no. 8, pp. 906–916, Aug. 2003.
- [16] J. Liu and P. Moulin, "Complexity-regularized image restoration," in *IEEE Int'l Conf on Image Proc.*, vol. 1, pp. 555–559, 1998.

## S.2.6. Low-cost wavefront coding using coma and a denoising-based deconvolution

- C. Dorronsoro, J. A. Guerrero-Colón, M. C. de la Fuente, J. M. Infante, J. Portilla, "Low-cost wavefront coding using coma and a denoising-based deconvolution" Comunicación Oral en SPIE Europe Symposium Optics/Photonics in Security & Defence. Electro-Optical and Infrared Systems: Technology and Applications. (September 2007).



# Low-cost Wavefront Coding Using Coma and a Denoising-based Deconvolution

Carlos Dorronsoro<sup>1</sup>, Jose A. Guerrero-Colon<sup>1,2</sup>, Marta C. de la Fuente<sup>3</sup>, Jose M. Infante<sup>3</sup>,  
Javier Portilla<sup>1,4</sup>

<sup>1</sup>Imatrics Image Technologies, Madrid, Spain

<sup>2</sup>Department of Computer Science and Artificial Intelligence, Universidad de Granada, Spain

<sup>3</sup>Indra Sistemas, Aranjuez, Madrid, Spain

<sup>4</sup>Instituto de Óptica, Consejo Superior de Investigaciones Científicas, Madrid, Spain

## ABSTRACT

Wavefront coding (WFC) is a powerful hybrid optical-numerical technique for increasing the depth of focus of imaging systems. It is based on two components: (1) an optical phase element that codifies the wavefront, and (2) a numerical deconvolution algorithm that reconstructs the image. Traditionally, some sophisticated optical WFC designs have been used to obtain approximate focus-invariant point spread functions (PSFs). Instead, we present a simple and low cost solution, implemented on infrared (IR) cameras, which uses a decentred lens inducing coma as an adjustable and removable phase element. We have used an advanced deconvolution algorithm for the image reconstruction, which is very robust against high noise levels. These features allow its application to low cost imaging systems. We show encouraging preliminary results based on realistic simulations using optical PSFs and noise power spectral density (PSD) laboratory models of two IR imaging systems. Without induced optical phase, the reconstruction algorithm improves the image quality in all cases, but it performs poorly when there are both in and out-of-focus objects in the scene. When using our coding/decoding scheme with low-noise detectors, the proposed solution provides high quality and robust recovery even for severe defocus. As sensor noise increases, the image suffers a graceful degradation, its quality being still acceptable even when using highly noisy sensors, such as microbolometers. We have experienced that the amount of induced coma is a key design parameter: while it only slightly affects the in-focus image quality, it is determinant for the final depth of focus.

Keywords: Wavefront Coding, IR Cameras, Coma, Image deconvolution, Image denoising, Image deblurring.

## 1. INTRODUCTION

In standard cameras, the optical system forms an image on the sensor plane, which directly provides the final image. Slight defocus produces a substantial drop of the image quality, especially with high numerical apertures. In infrared (IR) cameras, defocus may be produced not only by not being the target within the in-focus distance range, but also by a change of the external temperature and a subsequent dilation/contraction of the lenses or the mechanical elements supporting them. Low cost IR systems are incompatible with sophisticated (and expensive) athermalization mechanisms, thus avoiding precise focusing rings is desirable.

Wavefront coding (WFC) is a powerful hybrid optical-numerical technique for increasing the depth of focus of imaging systems<sup>1,2</sup>. WFC is based on two key components: (1) An optical element that introduces a controlled amount of phase in the wavefront and (2) a digital procedure based on a numerical deconvolution algorithm that reconstructs the optical image. The phase element produces an intermediate image at the detector plane which is optically coded, i.e. degraded in a controlled manner by a known Point Spread Function (PSF). Numerical procedures (image processing algorithms) are then needed to restore the intermediate image and provide the final image. The optical phase element is designed in such a way that its associated PSF at the detector plane does not appreciably change when defocus is introduced<sup>3</sup>. Although the intermediate image at the detector has lower quality than the corresponding focussed image of a standard system, it can be restored by deconvolution, as the PSF is known. Therefore, the great advantage of WFC is that for adequate combinations of phase masks and restoration algorithms the intermediate images are insensitive to defocus and so they

are the final restored images. While some in-focus quality may be lost (especially for high noise regimes<sup>4,5</sup>), this loss is made up for, since a much better out-of-focus quality is obtained.

WFC has been successfully demonstrated in visible light, for example in microscopy, or in low cost imaging cameras affected by chromatic aberrations<sup>2,6</sup>. Recently, some studies<sup>2,7,8</sup> propose the use of WFC imaging techniques with IR radiation. Some characteristics of IR image cameras that make them good candidates for being enhanced using WFC are:

- (1) IR systems are, due to their nature, unstable in terms of focus: variations in temperature, and IR cameras including athermalization mechanisms (sophisticated optomechanical hardware to avoid defocus of the system as external temperature changes) are expensive.
- (2) The optical system represents a high percentage of the final cost of IR cameras.
- (3) IR cameras already have computational capacity, as they typically have embedded processors for running image processing algorithms to enhance the image.

WFC is, thus, a promising technique to reduce the complexity of the optics, not only by avoiding costly optomechanical athermalization or refocusing mechanisms, but also by simplifying the optical design and even relaxing the manufacturing tolerances of the lenses. The goal is to loose some of the effort traditionally devoted to achieve a high image quality at the detector and shift it from image formation towards postprocessing (image restoration), which can be embedded in already existing electronic hardware. That is, WFC can be applied not only to improve the performance out of focus, but also to dramatically reduce the complexity (and, thus, the cost) of the optics while only introducing slight modifications in the embedded hardware.

The application of WFC in the infrared region is inherently more complex than visible and raises some important issues, both in the optical and in the postprocessing stages. Traditionally, both in visible light and IR, focus-invariant point spread functions (PSFs) are obtained with sophisticated phase plate designs<sup>3</sup>. In IR, manufacturing plates to induce such asymmetric phase maps entail important complications regarding new materials and methods for surface shaping. On the other hand, IR cameras have both lower spatial resolution and higher noise than visible cameras. Thus, imposing and assessing certain PSF designs, in which relies the image deconvolution, cannot be as precise in IR as in visible light.

There is an extensive scientific literature regarding postprocessing algorithms. The most used for WFC are variations of the Wiener deconvolution, as it represents a good trade-off between image quality and complexity. Recently, some studies<sup>4,5</sup> have tackled the issue of wavefront coding in presence of high detector noise. Although these studies are limited to Gaussian (white) noise, they have highlighted the impact of noise in the process of image restoration. IR imaging systems are usually characterized by high relative noise levels and a noise power spectral density far from flat (i.e., non-white), which makes the process of image restoration more demanding.

In this work we present a simple and low cost solution to implement wavefront coding in IR cameras, which uses 1) a de-centred lens inducing coma as an adjustable and removable phase element (Section 2.1) and 2) a two-step denoising-based deconvolution algorithm for the image reconstruction<sup>9,10,11,12</sup> (Section 2.2). This approach has two potential advantages:

- (1) Although the PSF of an optical system affected by coma is, from a theoretical point of view, less efficient for WFC than other PSFs already proposed, it also has a relatively low sensitivity to focus and it has no zeros in the spatial frequency domain, which favors the deconvolution quality<sup>13</sup>. Especially for low cost applications, where the images are affected by low resolution detectors, noise and uncontrolled blur, the approximation of using coma instead of other sophisticated designs may be practically interesting. However, the most important advantage of this solution is the practical simplicity of generating a comatic PSF. The optical phase element proposed in this study induces the optical coding with tilted or decentred conventional lenses, with spherical surfaces. Thus, the issue of manufacturing expensive asymmetric phase plates in the IR is avoided. Furthermore, as no extra element is added to the system, transmission is not degraded. Controlling the tilt or decentring results in a removable and adjustable WFC mechanism.
- (2) Using an advanced deconvolution algorithm<sup>9,10,11,12</sup>, which is very robust against high noise levels and can deal with non-white noise distributions, extend the range of WFC applicability with respect to using conventional Wiener-like deconvolution techniques, especially when using noisy detectors.

In this paper we explore the feasibility of the solution for certain combinations of optical coding and detector parameters. The optical phase element and the reconstruction algorithms are described in section 2. Section 3.1 describes an experimental proof of concept with visible images captured in controlled conditions. Section 3.2 addresses the problem in the IR, with the simulation of intermediate images using the measured parameters of the detector and the design parameters of the optics. Section 3.3 tests the image reconstruction algorithm on these images. Section 3.4 shows the WFC results for realistic noise of a low-cost detector. The procedure is also applied to a high quality IR detector (Section 3.5). The resulting thru-focus final images of these experiments are compared with the corresponding simulated images of a conventional IR camera. Section 4 discusses these results, and explores the importance of the amount of coma as a key design parameter of the system.

## 2. METHODS

Figure 1 shows a schematic diagram of the low cost IR imaging system proposed in this study, incorporating WFC. It is based on a conventional low cost IR camera (solid rectangles) with some additional elements (dashed rectangles) that incorporate WFC.

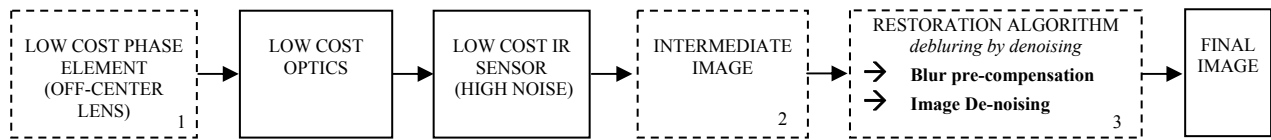


Figure 1. Schematic diagram of the low cost IR imaging system proposed, incorporating wavefront coding.

The first element (box number 1) is the low cost optical phase element (see section 2.1 for details). In a WFC scheme, the image at the detector (box 2) needs to be reconstructed by a restoration algorithm (box 3, see section 2.2) to obtain the final image.



Figure 2. On-axis-coma generator plus conventional thermal camera.

### 2.1 Low Cost Optical Coding

This work proposes to use decentred or tilted internal elements within the optical path inducing coma, as low cost optical phase elements<sup>13</sup>. If enough coma is introduced, then the PSF is made robust against aberrations and defocusing, while image processing algorithms can restore the coma-degraded image at the detector. In order to evaluate this approach, a simple and versatile system has been designed. Two elements, one plano-concave and one convex-plano lens with the same radius of curvature, are decentred one to each other by the same magnitude but opposite sign. A property of this configuration is that, being afocal, it can be placed in front of any conventional camera without affecting the camera focus adjustment. But, more importantly, the phase modulation can be adjusted by means of a variation of the lateral displacement between the two optical elements, as can be seen in figure 2. This system is a variation of that proposed by Hellmut et al.<sup>14</sup>. It can be demonstrated<sup>15</sup> that phase change in the (X,Y) plane (i.e., at the exit pupil) is given by:

$$Ph\_coma\_plate(x,y) = \frac{2\pi}{\lambda} \left\{ -(n-1) \left( \frac{2d}{R} - \frac{d^3}{R^3} \right) y + (n-1) \frac{d}{R^3} (x^2 + y^2) y \right\}$$

where  $R$  is the common curvature radius and  $d$  the displacement of each plate from the optical axis.  $R$  and  $d$ , are then, through coma aberration coefficient, the main design parameters in the PSF of the system. This work describes the



procedure to establish the amount of coma that optimizes the performance of the system. For that, the optical part has to be taken into account in combination with the detector characteristics, the deconvolution algorithm, the camera application, and the optical quality required.

## 2.2 Image Restoration

Image restoration is typically formulated as the estimation of an image given a linearly filtered version of the original corrupted by additive noise. This process is usually modelled in the frequency domain as follows:

$$Y_0(u, v) = H(u, v)X(u, v) + W_0(u, v),$$

being  $H(u,v)$ ,  $X(u,v)$  and  $W_0(u,v)$  the Fourier transforms of the PSF, signal and noise respectively. For simplicity, indexes  $(u, v)$  are drop hereinafter. Due to the low-cost sensor and the strong PSF of the system, the restoration algorithm must be able to deal with heavy blur in presence of high noise levels. In this work we have used a two-step restoration method proposed by Guerrero-Colon and Portilla<sup>10,11,12</sup>:

Step 1 - Global blur compensation: Performed by applying a generalized (global) Wiener filter:

$$G = \frac{H^*}{|H|^2 + \alpha \cdot \left(\frac{P_{w_0}}{P_x}\right)},$$

where  $P_{w_0}$  and  $P_x$  are the power spectral densities of noise and signal, respectively. The balance between noise and blur suppression is controlled by the regularization parameter  $\alpha$ .

Step 2 - Local denoising: After step-1 the observation yields:

$$Y_1 = GY_0 = GHX + GW_0 = H_rX + W_1.$$

Because  $G$  has been optimized in the first step, no further attempt to correct the remaining linear distortion  $H_r = GH$  is made, and what is left is to denoise the prefiltered image using a non-linear algorithm<sup>9</sup>.

The method assumes known both degradation parameters (PSF and  $P_{w_0}$ ), hence only two parameters ( $\alpha$  and  $P_x$ ) must be estimated.  $\alpha$  is estimated off-line by training with a set of standard images, which yields a very robust value for a wide range of degradation (both blur and noise). We use for  $P_x$  the simple (and well known)  $k/f^2$  model, where  $f$  represents absolute spatial frequency and  $k$  is a constant that ensures consistency with the original estimated variance.

Summarizing, the chosen method is very effective, and it has shown robustness against a wide variety of noise levels and PSF's. It is also efficient and non-iterative, so it has a strong potential to be used as a real-time restoration algorithm, when combined with efficient denoising algorithms. Furthermore, to the best of our knowledge, it provides state-of-the-art performance for the common simulation experiments in the literature<sup>10,11</sup>.

## 2.3 Experiments

### 2.3.1 Visible camera

The goal of these initial studies in visible light was to test if the algorithms selected for deconvolution<sup>10,11,12</sup> were able to restore a real image severely affected by coma, its robustness against noise, sampling, or bit depth reduction. For this purpose, a tilted singlet ( $f=50\text{mm}$ ) was placed in front of a high quality digital CCD camera (Hamamatsu ORCA 100). The PSD of the noise was calculated, following standard procedures, from images of uniform scenes. This was made for single frames, and also for averages of images resulting in highly reduced noise. The PSF of the system was experimentally obtained from images of pinholes with variable diameters and luminance.

Different objects, generated by a high brightness minidisplay (LiteEye), were placed in the object plane of the system (3 meters away from the lens). This procedure allows a precise control of the objects to be imaged, and an immediate reference for algorithm evaluation: the ideal non-degraded version of the object. With this distance configuration, each

pixel of the camera subtended more than 10 pixels of the minidisplay, so sampling effects can be neglected. 30 images were taken for each object.

Different combinations of coma and defocus were tested, by changing the tilt and the object position, respectively, to demonstrate the capability of the algorithm to manage the resulting PSFs. To study blur compensation in ideal conditions the full resolution of the image was used, and 30 frames were averaged to reduce noise. The noise effect was then assessed by processing single frames. Sampling influence was studied by reducing the image resolution to a half, and also the effect of a bit depth reduction from 12 to 8 bits (see results in section 3)

### 2.3.2 Simulation of IR images

Realistic simulated IR images were obtained considering the coma subsystem described in section 2.1, in combination with a conventional IR objective ( $f/1$ , EFL 75mm and FOV  $11^\circ \times 8^\circ$ ), and a  $320 \times 240$  microbolometer sensor ( $45\mu\text{m}$  pitch), which provides 8 bit visible images, affected by non-white noise. Some of the PSFs used in the simulations are shown in figure 3. They were obtained from the optical design of the system, including the low cost adjustable/removable phase element with variable  $R$  and  $d$  for the coma subsystem. Polychromatic IR radiation was used to obtain the PSFs, that were then scaled to the pitch size of the detector and normalized. The PSD of the noise was calculated from laboratory images of uniform backgrounds. A different random noise pattern was computer-generated for each frame, and added to the simulated intermediate (optically degraded) image. Both PSF and noise PSD were applied to a set of high resolution and low noise IR Images (from Merlin InSb 9705 and Phoenix LF InSb 9803).

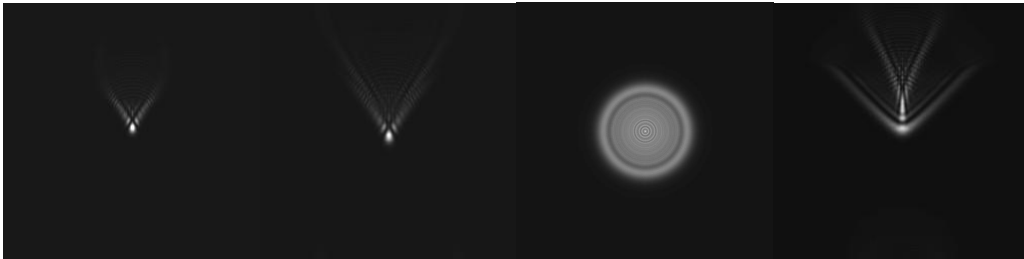


Figure 3. Some of the PSFs used for the simulation, with variable amounts of coma and defocus

### 2.3.3 Simulation of the Wavefront Coding process

Several experiments were performed to assess the limits of the restoration process. Different amounts of optical coma were simulated, by applying different spatial scale factors (8x, 4x, 2x, 1x and  $\frac{1}{2}x$ ) to the realistic PSF. 8x corresponds to the high resolution PSF provided by the optical design program (Code V). This simulates different spatial extents of the PSF relative to the detector grid. Several noise factors were applied (2x, 1x, 0.5x, 0.25x, 0.10x) to the realistic simulated noise obtained with the measured PSD. The two-step deconvolution algorithm was compared to conventional Wiener filtering.

Two cases were studied in detail: a realistic low-cost camera and a low-noise sensor with the same optics. To study the first case, the images were simulated using the measured noise characteristics of the existing microbolometer-based MVT IR Camera (INDRA, Aranjuez, Spain), and the design parameters of the optics (modified with the addition of the comatic phase element). Thru-focus PSFs were used, to explore the image quality out of focus. The second case (high quality sensor) was simulated with the same parameters, except the amount of noise, which was diminished by a factor of 10.

In the restoration/deconvolution process was either assumed that the amount of defocus was known (which doesn't really reflect the real conditions) or a blind approach (as in wavefront coding typical scenario) where the defocus is unknown and, thus, the same PSF (typically the in-focus-PSF) is used for the deconvolution of all the intermediate images. The resulting images were compared to the conventional IR camera approach, that doesn't use any phase element.

### 3. RESULTS

#### 3.1 Image Reconstruction: Visible

Figure 4.a shows the intermediate image captured with the 12-bits visible CCD detector (one frame). The lens tilt was adjusted to get an intermediate image severely affected by coma. The measured PSF subtended more than 50 pixels on the image. The resulting images after the two step denoising based deconvolution are shown: 4.b) low noise (the intermediate is obtained after averaging 30 frames), 4.c) high noise (one single frame), 4.d) High noise and subsampling to half the initial image resolution in both spatial dimensions. A bit depth reduction from 12 to 8 bits has no appreciable effect on the final images. As an example, Figure 4.d shows the result of restoring a high noise and reduced resolution image, from a 12 bit (up) or 8 bit (down) intermediate image.

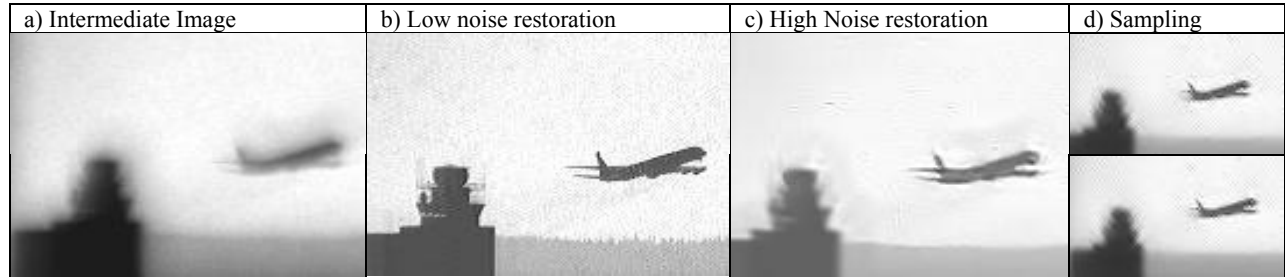


Figure 4: Image restoration of real images affected by coma: See text for details.

In low-noise conditions (4.b), the reconstruction method produces a very important visual improvement in the image, as it is able to recover much detail that was lost by optical blur in the intermediate image (4.a). When high noise is present (4.c), some wavelet-type artifacts appear, but the improvement is still important. The method is robust against subsampling of the intermediate image to half resolution (4.d up) and quantization to 8 bits (4.d down).

#### 3.2 IR simulated images

Figure 5.a shows a high contrast image of an homogeneous object, taken with the MVT IR Camera (up), and simulated from the noise PSD of its detector (down). Both images, real and simulated, are different because correspond to different noise samples, but share the same statistics.

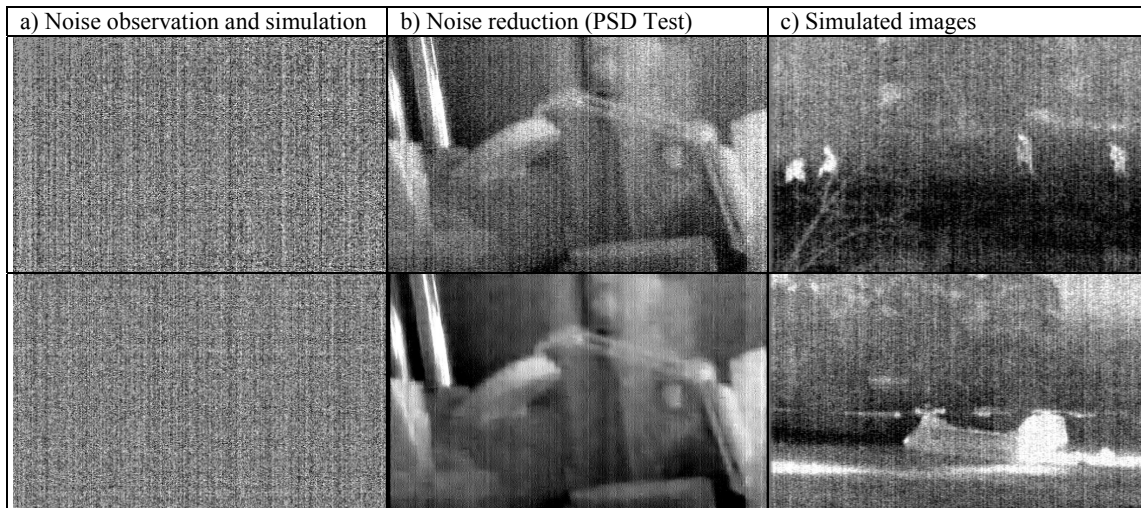


Figure 5: Noise simulation. a) laboratory observed (up) and simulated (down) detector noise; b) real laboratory image without optical coding (up) and denoised image using the measured noise PSD (down); c) Two simulated noisy IR images with realistic noise.

In order to validate the noise description, the PSD was used to denoise a real image (Figure 5.b). The two-step deblurring by denoising method was applied to the real image (up), with a Dirac delta as PSF, to inactivate the deblurring process. The resulting image (down) represents a good estimation of the scene, with noise dramatically reduced. This result represents a simultaneous validation of the PSD used, and the second step of the restoration algorithm for IR images. Figure 5.c shows two examples of simulated images with realistic noise. These simulated images have similar appearance to the real image shown in 5.b up.

No optical blur has been simulated in figure 5. To simulate blur, image is convolved with the simulated PSF prior to the addition of the noise. Different combinations of noise and blur can be obtained by scaling the PSF (spatial support) and applying a multiplicative factor to the simulated noise. Figure 6 shows three examples. From left to right: low degradation (PSF  $\frac{1}{2} \times -7 \times 7$  pixels- and noise  $\frac{1}{4} \times$ ); realistic degradation (PSF  $1 \times -16 \times 16$  pixels- PSD  $1 \times$ ); and severe degradation (PSF  $2 \times -32 \times 32$  pixels- PSD  $2 \times$ ).



Figure 6: Examples of simulated degradation, including detector noise and optical blur: low degradation (left), realistic simulation (center), and severe degradation (right).

### 3.3 Image reconstruction: IR

The two-step deblurring-by-denoising was applied to the simulated degraded images, and compared to Wiener filtering. In all cases, Wiener filtering introduces severe artifacts in the image. Figure 7 shows the results for two images with different levels of degradation. In the rest of this manuscript, the results of the Wiener filtering are not shown, as this method is worse adapted to these image degradations.

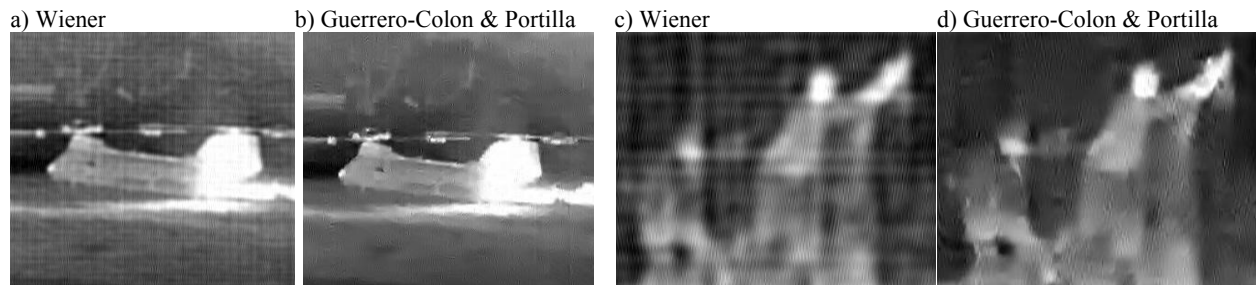


Figure 7: Comparison of restoring algorithms (see text for details).

### 3.4 Wavefront Coding process with realistic noise

Figure 8 shows the results of the image restoration of the intermediate images simulated with realistic parameters for the MVT Camera with the low cost phase element. Even when there is no need for blur compensation, in focus and without optical coding, the algorithm improves the quality of the images by reducing the noise, as already seen in figure 5.b. Two out-of-focus positions are shown in figure 8. “High defocus” corresponds to the full defocus shift expected for thermal dilation/constriction of the optomechanical elements, and “moderated defocus” to half of that. Without optical coding and out of focus, the intermediate (degraded) image at the detector suffers a substantial drop in optical quality. As

expected, the defocus blur cannot be correctly compensated and many artifacts are introduced in the reconstruction process, even if the defocus PSF is precisely known. The reason is that the defocus PSF has zeros in the frequency domain which make the deblurring step difficult.

With optical coding, the quality of the intermediate image is much more stable thru-focus. Thus, the final image quality out of focus is much better, after reconstruction, than that of a standard system. This is due to the characteristics of the comatic PSF, better suited for image deblurring than the defocus PSF of a corrected system. As a consequence of the PSF stability thru focus induced by coma, the guided reconstruction (in which the amount of defocus is known, and therefore there is a precise knowledge of the PSF) and the blind reconstruction (in which there is no a priori knowledge of the defocus state) end up with similar image quality. This result is decisive for the viability of a low cost design based on the solution proposed in this work. The in focus PSF can then be used in a broad range of focus positions, and neither correction nor control of the focus state is needed.

Images with asterisk in figure 8 illustrate the main advantages and drawbacks of using WFC with low cost detectors. The in-focus image quality of a conventional system usually has the maximum resolution allowed by the detector, but is affected by noise. When WFC is used, the process of optical blurring and digital deconvolution in presence of noise yields an image with slight resolution loss, but also with reduced noise. With high defocus, the conventional system produces a completely blurred image, while the WFC system retains most of the image quality in focus.

The artifacts appearing in all the reconstructed images, are related to the presence of high noise in the deblurring process. The following section explores how a better detector with lower noise figures could improve the reconstruction process.

### 3.5 Wavefront Coding process with low noise

As expected and shown in figure 9, WFC provides much better results when detector noise decreased. In these simulations, the same PSFs were used, but the noise factor used was 0.1. Comparing the images of a standard system (first row) with the final images in a WFC scheme (last row), the conclusions are different now than those of previous section. With low noise, the image quality in focus with WFC has the same quality as that of a conventional system. Out of focus, the differences are dramatic. While in a conventional system high spatial frequencies are lost, with WFC the effective resolution of the image is almost unaltered, and only minor artifacts are introduced in the image.

## 4. DISCUSSION

The performance of the proposed WFC system strongly relies both on the correct optical coding induced by the low-cost phase element and on the restoration process by the deconvolution algorithm. Despite the latter had been carefully verified for simulated experiments involving natural images in the visible spectrum<sup>11</sup>, it was crucial to test it under real conditions of IR images. Experiments in section 3.1 show that the method can correctly deal with real degradations in visible detectors, and that it is also robust against other kind of degradations such as sampling, and bit-rate reduction, usually found in IR. Nevertheless it must be noted that the final performance is conditioned by the quality of the estimation of the degradation parameters (PSD and PSF). As shown in Figure 5.b, the algorithm is also suitable for restoring real IR images under heavy noise, which in turn means that the algorithm is appropriate for low-cost IR detectors. Note that in this case no blur compensation has been applied to the image, as the noise widely dominates the image degradation. Finally, Figure 7 includes a full restoration example in IR with our method (with simulated images including different combinations of noise and optical blur), which clearly surpasses other well known method.

Regarding the image quality, we have shown that the algorithm is able to successfully compensate heavy noise and optical blur separately. A denoising example with a real IR image can be seen in figure 5.b, and Figure 8 -in focus, with no optical coding- shows the result on a realistic simulated IR image. Blur compensation results (in low noise) can be seen in figure 2.b for real visible images and figure 9 -last row- shows three results on simulated IR images. But the combination of both, blur and noise, may be harmful. In general, the quality of the blur compensation relies heavily on the noise level and the little the noise, the better the performance. In all the restored images of Figure 8, we can detect wavelet-like artifacts, which are due to the deconvolution in presence of high noise levels. Figure 9, in contrast, has no artifacts, except for the high defocus case.

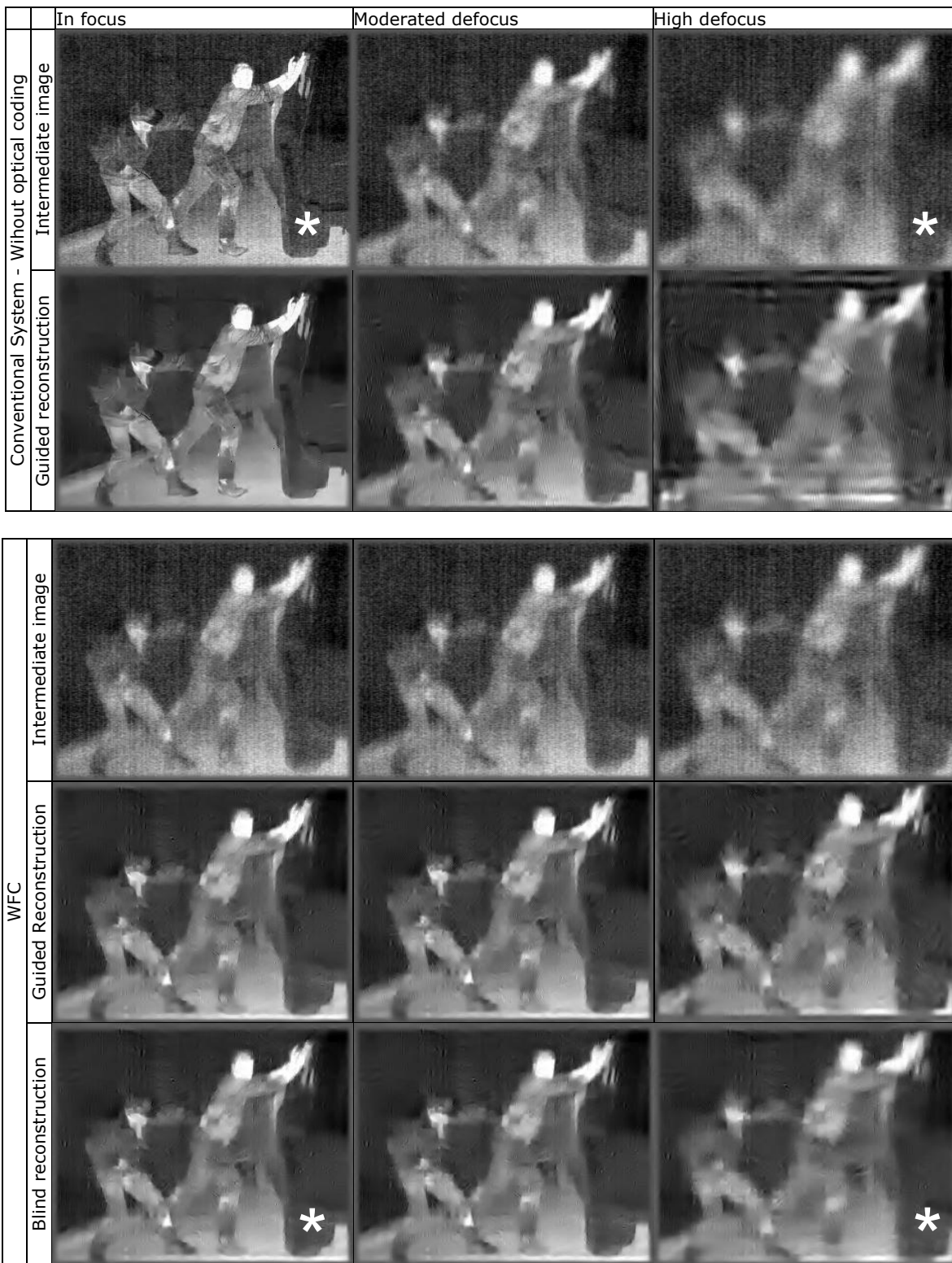


Figura 8: Restoration results. See text for explanation.

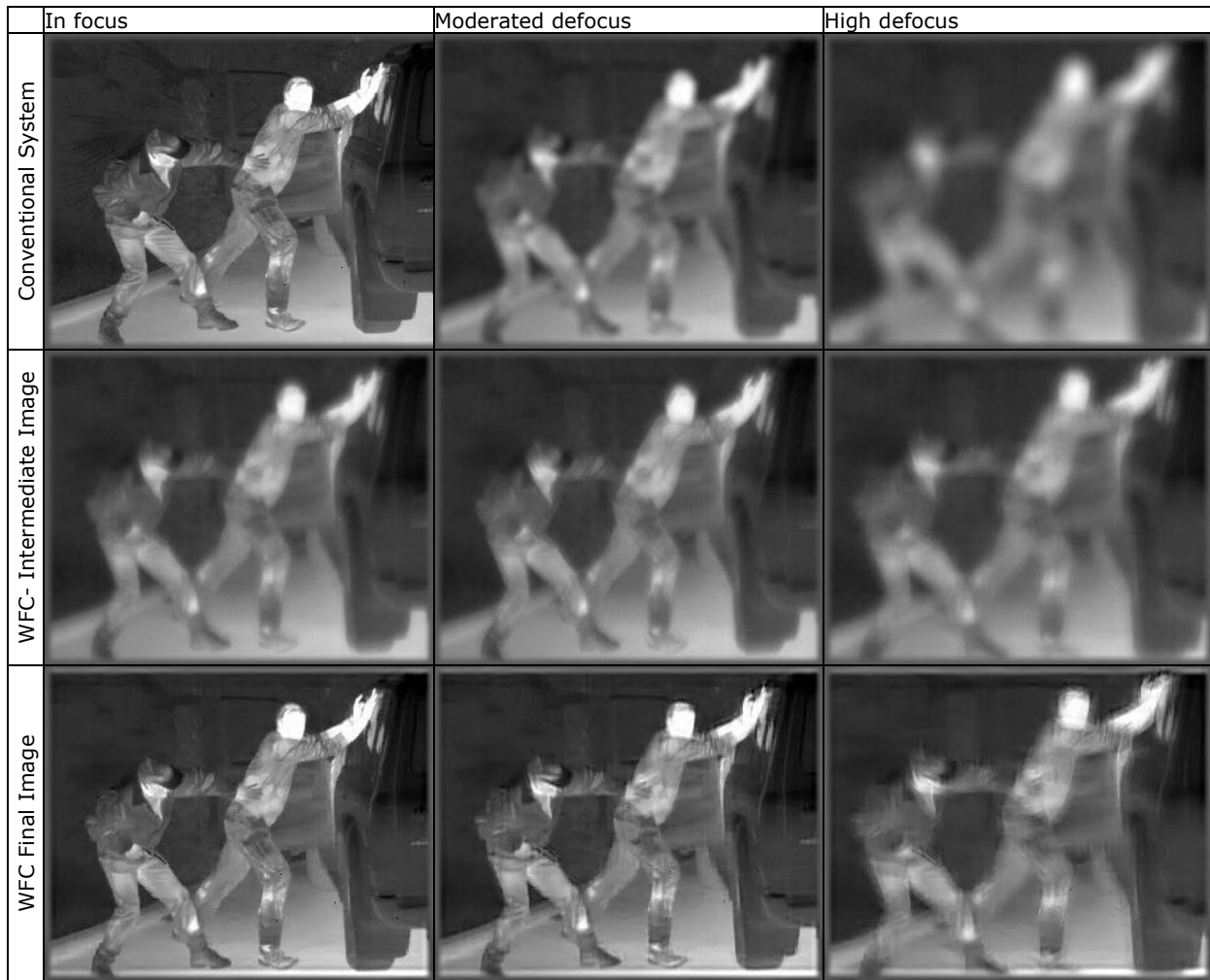


Figure 9. Low Noise restoration

As already pointed out, the key design parameter in this WFC scheme is the amount of coma. High coma yields in large PSF's. This has a double effect; on the one hand, the intermediate images become less sensitive to defocus; on the other hand, the in-focus image quality decreases. Figure 10 illustrates this fact. Different amounts of coma (4x, 2x, 1x, 1/2 x) produce, for the same amount of realistic noise, a progressive decrease of in-focus image quality. We point out that this effect strongly depends on the noise level, and in the case of very high quality detectors, this counter-effect is negligible (as shown in Figure 9, bottom left panel).

When dealing with low-cost detectors for WFC, is critical to find a good compromise between the detector noise (PSD) and the amount of blur (PSF). In that way, the scheme presented here is very flexible. The use of tilted or decentred optical elements, besides being a simple and inexpensive solution, allows us to adjust the amount of optical coding, and to explore the trade-off between depth of field and in focus image quality. Furthermore, the optical blur, and thus the WFC effect can be reduced to zero. As all the denoising capabilities of the electronics are maintained, the maximum image quality can be recovered at any time. Then, the coma-based optical coding proposed here represents an appealing alternative to other solutions like phaseplates, which although could produce a better WFC effect, are neither removable nor adjustable, and they may be difficult to manufacture in IR. In the example presented here both detector and opto-mechanical layout were defined in advance, and the optimal performance is obtained with an amount of coma whose PSF subtends around 15 pixels on the detector, depending on the image content. As future work, this key design parameter will be used to process real IR noisy intermediate images. There are some issues, not appearing in simulated images, that

could arise in a real implementation. Among them: variations of the PSF along the field of view, changes in the noise PSD with the electronic gain or the non-uniformity correction applied to the image, sampling/aliasing problems, or artifacts due to bad pixels. These are important practical issues for further research.



Figure 10: In-focus image quality versus the amount of coma, for realistic noise

## 5. CONCLUSIONS

We have presented a simple and low cost wavefront coding (WFC) solution, implemented on infrared (IR) cameras, which uses a decentred lens inducing coma as an adjustable and removable phase element. We have used an advanced deconvolution algorithm for the image reconstruction, which is very robust against high noise levels. We have demonstrated through realistic simulations that the proposed scheme significantly reduces the final image quality degradation caused by defocus. We have tested, with different amounts of defocus, two situations: heavy noise and moderate-low noise. In both scenarios the added optics combined with the numerical deconvolution result into a strong image quality improvement. Severe defocus and/or noise produce a graceful image degradation. For highly noisy IR simulated images, even in the absence of defocus, the proposed scheme dramatically improves the image quality, because of the powerful denoising applied.

We have experienced that the amount of induced coma is a key parameter design. Although it only moderately affects the in-focus image quality, it is determinant for the final depth of field. It also has a direct influence on other design parameters, as the amount of computation per frame, or the sensor size.

Summarizing, the proposed solution is robust, flexible, inexpensive and very effective, and we believe it may become an alternative to more sophisticated WFC schemes in many practical cases. Future development and spread of this kind of hybrid capture-processing techniques will depend on the availability of highly efficient implementations of advanced numerical deconvolution / denoising algorithms, especially when aiming to design real-time response cameras.



## 6. ACKNOWLEDGEMENTS

Work described in this paper was supported by the Spanish MoD COINCIDENTE program DN8644. JAGC is funded by AP2005-2969 FPU fellowship (M.E.C.). JP is funded by the "Ramón y Cajal" Program (M.E.C.). JAGC and JP are also funded by TEC2006/13845/TCM (M.E.C.) grant. Thanks are due J. M. Lazaro (INDRA) and to F. Márquez (Testing Laboratory, CIDA) for previous collaborations and technical support.

## 7. REFERENCES

1. E. R. Dowski and W. T. Cathey "Extended depth of field through wave-front coding" *Applied Optics* Vol. 34, N° 11, pp. 1859-1866 (1995)
2. W. T. Cathey and E. R. Dowski "New paradigm for imaging systems" *Applied Optics* Vol. 41, N°29, pp. 6080-6092 (2002)
3. S. S. Sherif, W. T. Cathey, and E. R. Dowski, "Phase Plate to Extend the Depth of Field of Incoherent Hybrid Imaging Systems," *Appl. Opt.* 43, 2709-2721 (2004)
4. S. Bosch, R. Tudela, M. C. de la Fuente, J. Ferré-Borrull, "Impact of the 2D structured noise in the post-processing of hybrid optical-digital imaging systems", *Proceedings of SPIE*, Vol 6342 (2006)
5. S. S. Sherif, E. R. Dowski, and W. T. Cathey, "Effect of detector noise in incoherent hybrid imaging systems," *Opt. Lett.* 30, 2566-2568 (2005)
6. S. Tucker, W. T. Cathey, and E. Dowski, Jr., "Extended depth of field and aberration control for inexpensive digital microscope systems," *Opt. Express* 4, 467-474 (1999)
7. K. Kubala and E. Dowski, "Reducing size, weight, and cost in a LWIR imaging system with Wavefront Coding" *Proceedings of SPIE* Vol. 5407 (2004)
8. G. Muyo, A. Singh, M. Andersson, D. Huckridge, A. Harvey "Optimized thermal imaging with a singlet and pupil plane encoding: experimental realization" *Proceedings of SPIE*, Vol 6395, (2006)
9. J A Guerrero-Colon and J Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *IEEE Int'l Conf. on Image Proc.* (2005)
10. J A Guerrero-Colon and J Portilla, "Deblurring-by-denoising using spatially adaptive gaussian scale mixtures in overcomplete pyramids," in *IEEE Int'l Conf. on Image Proc.* (2006)
11. J.A. Guerrero-Colon, J. Portilla "Image Restoration Using Space-Variant Gaussian Scale Mixtures in Overcomplete Pyramids". *IEEE Trans, on Image Processing*, in press, (2007)
12. J. Portilla, J. A. Guerrero-Colon and C. Dorransoro "Method for the restoration of images affected by imperfections and its applications". *OEPM Patent application P200602116* (2006)
13. M. C. de la Fuente, "GASIR 1: a promising material for dual waveband systems" *Proceedings of SPIE*, Vol 6342 (2006)
14. T. Hellmut, A. Bich, R. Borret, A. Holschbach, A. Kelm, "Variable phaseplates for focus invariant optical systems" *Proceedings of SPIE*, Vol 5962 (2005)
15. S. Bosch, J. Ferré-Borrull, R. Tudela, *Internal Progress Report for INDRA*, (2006)

## S.2.7. Realistic Bayesian camera identification based on demosaicing traces

- J. A. Guerrero-Colón, M. A. T. Figueiredo and J. Portilla, "Realistic Bayesian camera identification based on demosaicing traces", IEEE Int'l Conf. on Image Proc., 2008, Submitted.



# REALISTIC BAYESIAN CAMERA IDENTIFICATION BASED ON DEMOSAICING TRACES

Jose A. Guerrero-Colón\*,

Mário A. T. Figueiredo,

Javier Portilla†

Dept. of Computer Science and A.I.  
Universidad de Granada  
Granada (Spain)  
jaguerrero@decsai.ugr.es

Instituto de Telecomunicações  
Instituto Superior Técnico  
Lisboa (Portugal)  
mtf@lx.it.pt

Instituto de Óptica  
CSIC  
Madrid (Spain)  
portilla@io.cfmac.csic.es

*We present a Bayesian classifier to identify the digital camera model from a single acquired image. As in other works, we use the traces of the demosaicing algorithms as features, but we extract them in an *least-squares-optimal* way, coupling all color channels. We compare two simple statistical models to describe the cameras (Gaussian and Gaussian mixture), as well as two different combination rules (*ad-hoc* vs. *model based*). We train and test the proposed algorithm with unrestricted image sets (in scenery and/or capture conditions). This fact, coupled with its robustness and flexibility, allows using the algorithm in real situations, achieving promising results.*

**Index Terms**— Image forensics, camera identification, Bayesian classifier.

## 1. INTRODUCTION

Image forensics is a broad area including many applications sharing a common goal: to obtain information concerning the source or the authenticity of an image. Forgery detection, secret information detection (steganalysis), and image type identification are just a few examples. In this paper, we propose an identification method to detect the camera model that was used to capture a given image.

Most digital cameras include headers in their image files, with information about the device, as well as other data, such as aperture, shutter speed, time and date. However, this information is unreliable, since it can be easily edited and changed. Recently, watermarking has been proposed as a solution to the authentication problem (see [1, 2, 3], for overviews), namely by “*sign-stamping*” images in the acquisition device. This approach should be able to solve the camera identification problem; however, it requires dedicated hardware in the cameras, and is currently ignored by the manufactures.

To solve the identification problem, several authors have proposed methods exploiting the imperfections inherent to each particular device [4, 5, 6]. Unfortunately, all of them require the devices to be available for inspection, giving no answer to many real situations where this is not possible.

In 2004, Kharrazi *et al.* [7] proposed a set of image features (such as average pixel values, RGB pairs correlations and energy) to identify the capture device. The results achieved were very promising, but subject to certain restrictions; namely, the classifier (a support vector machine—SVM [8]), was mainly trained with images that shared the same scenery with the test images and that were captured

under controlled conditions<sup>1</sup>. One year later, Bayram *et al.* [9, 10], inspired in other forgery detection methods [11, 12], proposed an algorithm that used traces of the color filter array (CFA) interpolation schemes to distinguish among cameras. As Kharrazi *et al.* [7], they used an SVM classifier, and the training/testing sets suffered from the same constraints<sup>2</sup>. In parallel and independent work, Portilla *et al.* [13, 14] presented, among other things, a model and method for extracting CFA traces, claiming their potential use in camera identification.

In this paper, we present an identification method based on the traces of the CFA interpolation schemes. In contrast with other methods [9, 10], where the features are extracted independently for each color channel [12], we follow an LS-optimal methodology [13, 14] coupling them together. The algorithm is based on a Bayesian classifier that, unlike other methods [7, 9, 10], is trained and tested with totally unrestricted image sets, making it suitable to be used in real situations.

## 2. BACKGROUND

A central (and, in general, one of the most expensive) component of a digital camera is the image sensor (typically CCD or CMOS) which converts light into electric signals. These devices are monochromatic, so three of them are needed to capture a full RGB image, dramatically increasing the cost and complexity of cameras. To avoid this cost, most manufactures use only one sensor equipped with a color filter array (CFA) that captures different color components according to a certain mosaic. The most used mosaic is made of repetitions of the Bayer pattern [15], shown on the left side of Fig. 1. Hence, the captured (raw) image is incomplete, in the sense that at each location of the (CCD or CMOS) sensor, only captures one color component (R, G, or B). The full RGB image is obtained by locally interpolating the raw image, to recover the missing colors, a process known as *demosaicing*. In general, manufactures use their own demosaicing algorithms, using different filters (in shape and size) and a wide variety of interpolations functions. This fact implies that the demosaicing process is strongly related to the source camera model, and, hence, is potentially useful for identification. In the next section we describe an LS-optimal procedure to extract these features [14].

## 3. FEATURE EXTRACTION

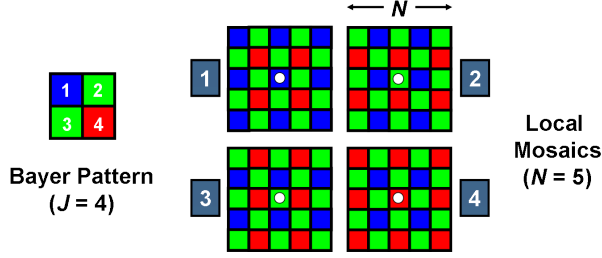
Images captured by CFA sensors can be modelled as

$$\mathbf{y} = \mathbf{S}\mathbf{z},$$

<sup>1</sup>In the best case, 2/5 of the sets were affected by these constraints.

<sup>2</sup>In the best case, 2/3 of the sets were affected by these constraints.

\*JAGC is supported by AP2005-2969 FPU contract and TEC2006/13845/TCM grant, both funded by the Ministerio de Educación y Ciencia (Spain). †JP is supported by TEC2006/13845/TCM grant and the Ramón y Cajal Program, both funded by the Ministerio de Educación y Ciencia (Spain).



**Fig. 1.** Local mosaics obtained from the classic Bayer pattern.

where  $\mathbf{S}$  is the color sampling matrix implemented by the CFA and  $\mathbf{z}$  is the complete color image with  $3P$  elements, where  $P$  is the number of acquired pixels in  $\mathbf{y}$ . Modelling the demosaicing procedure as linear, the observed interpolated image can be written as

$$\tilde{\mathbf{y}} \approx \mathbf{D}\mathbf{y}, \quad (1)$$

where  $\mathbf{D}$  is a  $3P \times P$  interpolation matrix, which contains the information (i.e., the features) needed to identify the camera. The huge size of  $\mathbf{D}$  precludes, in practice, solving the problem through pseudo-inversion. To overcome this, a local solution was proposed in [13, 14], by forcing the estimates to have a restricted spatial support. The main drawback of that solution is that it requires careful handling of local mosaics, which depend on sensor localization. A mosaic is made of repetitions of a basic pattern of  $J$  elements with  $N_c$  different colors ( $J > N_c$ ). So, there are  $J$  different local mosaics, one per each of the  $J$  element. This is illustrated in Fig. 1, for a typical RGB Bayer pattern ( $N_c = 3$ ;  $J = 4$ ).

At this point, assumed known the mosaic phase, we need an information reordering. We divide the image into  $J$  sets (with  $L_j$  elements each), according to the local mosaic type around each pixel. Let  $\mathbf{y}_i^j$  be the vector made of the unknown color samples of pixel  $(i, j)$ , where  $j$  indexes the  $J$  sets and  $i$  points to one among the  $L_j$  pixels of the set. We construct the matrices  $\mathbf{Y}_j = \{\mathbf{y}_1^j, \mathbf{y}_2^j, \dots, \mathbf{y}_{L_j}^j\}$ , each one sized  $(N_c - 1) \times L_j$ . Now, we make explicit the local estimation constraint by forcing the prediction to be made using only  $N \times N$  samples around each pixel. For that, we define  $\tilde{\mathbf{y}}_i^j$ , as the vector containing all the observed color samples in an  $N \times N$  neighborhood around pixel  $(i, j)$ . As before, we build a matrix with all neighborhoods:  $\tilde{\mathbf{Y}}_j = \{\tilde{\mathbf{y}}_1^j, \tilde{\mathbf{y}}_2^j, \dots, \tilde{\mathbf{y}}_{L_j}^j\}$ . According to this, we can predict the unknown color samples of a pixel as a linear combination of the observed neighbors<sup>3</sup> color samples:

$$\mathbf{Y}_j = \mathbf{D}_j \tilde{\mathbf{Y}}_j.$$

Applying pseudoinverse (involving this time a much smaller inverse,  $N^2 \times N^2$ ), we get the  $J$  desired interpolation filters:

$$\mathbf{D}_j = \mathbf{Y}_j (\tilde{\mathbf{Y}}_j)^\# = \mathbf{Y}_j (\tilde{\mathbf{Y}}_j \tilde{\mathbf{Y}}_j^T)^{-1} (\tilde{\mathbf{Y}}_j). \quad (2)$$

We use the phase estimation method described in [14]. It computes all the  $J$  interpolators for the  $J$  possible phases, choosing the one yielding the lowest quadratic error between observation and the resulting interpolated image.

The feature vectors, denoted  $\mathbf{x}$ , are extracted from the filters  $\mathbf{D}_j$ . We have  $J(N_c - 1)$  filters of  $N^2$  elements, one for each type of mosaic  $j$  and each unknown color  $(N_c - 1)$ . Each filter can be reduced to  $((N + 1)/2)^2$  coefficients, by imposing symmetry. Thus, the final feature vector has dimension  $(N_c - 1)J((N + 1)/2)^2$ .

<sup>3</sup>Note that, unlike in Eq.1 and for convenience, now we predict just the unknown color samples.

#### 4. BAYESIAN CLASSIFIER FOR SOURCE CAMERA MODEL IDENTIFICATION

Given a set of  $K$  camera classes,  $\{c_1, c_2, \dots, c_K\}$ , and a feature vector  $\mathbf{x}$  extracted from an observed image  $\mathbf{z}$ , our goal is to identify which of these cameras produced the image  $\mathbf{z}$ . To this end, we adopt the Bayesian paradigm, in particular the maximum a posteriori (MAP) classifier

$$\hat{c} = \arg \max_{c \in \{c_1, \dots, c_K\}} P(c|\mathbf{x}) = \arg \max_{c \in \{c_1, \dots, c_K\}} \frac{p(\mathbf{x}|c)P(c)}{p(\mathbf{x})}, \quad (3)$$

where  $\{P(c_1), \dots, P(c_K)\}$  are the a priori class probabilities and  $p(\mathbf{x}|c)$ , for  $c \in \{c_1, \dots, c_K\}$ , are the class-conditional densities (or likelihood functions), i.e., probabilistic models of the features, given each camera class, which we refer to as camera models.

##### 4.1. Camera Models and Parameter Estimation

We consider two simple models for representing the class-conditional densities. The first one is simply a Gaussian model  $p(\mathbf{x}|c_k) = \mathcal{N}(\mathbf{x}|\mu_k, \mathbf{C}_k)$ , for  $k = 1, \dots, K$ . The second one, assumes that each density  $p(\mathbf{x}|c_k)$  is a Gaussian mixture, that is,

$$p(\mathbf{x}|c_k) = \sum_{i=1}^{G_k} \alpha_{i,k} \mathcal{N}(\mathbf{x}|\mu_{i,k}, \mathbf{C}_{i,k}).$$

This second models requires, for each class density  $p(\mathbf{x}|c_k)$ ,  $G_k$  times more parameters, plus the associated  $G_k$  weights.

We estimate the model parameters from a training set containing  $M_k$  independent images,  $\{\mathbf{x}_1^k, \dots, \mathbf{x}_{M_k}^k\}$ , from class  $k$ , for  $k = 1, \dots, K$ . This is done by using the usual maximum likelihood (ML) criterion. For the single Gaussian case, the parameters to estimate simply are the mean and covariance of each class.

In Gaussian mixture models, the parameters are estimated via the standard expectation-maximization (EM) algorithm (see, for example, [16]). We assume, for now, that the number of components in each mixture,  $G_k$ , is known.

##### 4.2. Decision rule

For computational reasons, a given observed image  $\mathbf{z}$  is divided into  $M$  blocks and a set of  $M$  feature vectors  $\{\mathbf{x}_{(1)}, \dots, \mathbf{x}_{(M)}\}$  are obtained from these blocks, as described in Section 3. Thus, to classify a given image, we need to combine the evidence provided by the  $M$  blocks, for which we consider two strategies: *a*) a voting scheme and *b*) a model-based scheme, where the image regions are considered as independent observations of the same class, thus the MAP classifier has the form

$$\hat{c} = \arg \max_{c \in \{c_1, \dots, c_K\}} P(c) \prod_{m=1}^M p(\mathbf{x}_{(m)}|c). \quad (4)$$

##### 4.3. Implementation Details

We assume a Bayer pattern (Fig. 1), which implies  $N_c = 3$  (RGB) and  $J = 4$  (local mosaics). We chose a  $5 \times 5$  local estimation neighborhood ( $N = 5$ ). With this setup, and after symmetry reduction, the feature vectors have 72 dimensions.

Due to the high dimensionality of the parameter space, poorly conditioned covariance estimates may arise. To avoid this effect, we regularize these estimates by adding a small constant to the diagonal of these covariance estimates. Initialization of the EM algorithm is done as suggested in [16].

## 5. RESULTS AND DISCUSSION

### 5.1. Data source

To assess the performance of the proposed classifiers, we have selected four SLR cameras: *Nikon D70*, *Canon EOS 300D*, *Olimpus E300* and *Konica Minolta A2*. All of them share the same mosaic type (Bayer), with unknown phase [17]. We have no details about the interpolation methods. Images are taken from A. Sierra’s webpage<sup>4</sup>, 10 per camera, capturing very different scenes. Each image is split into  $M = 77$  non-overlapping patches (of size  $256 \times 256$ ) giving rise to 770 patches per camera. Samples will be independently classified and their outputs will be combined according to the decision rule to classify the whole image.

### 5.2. Training: Parameter Learning

We have estimated the camera model parameters from 20%, 30% and 40% of the samples, by randomly selecting 2, 3 and 4 whole images, respectively. This ensures that the training and test sets are totally independent. We verified that a totally random selection, allowing the same image to have (different) patches in the training and test sets, leads to a striking performance increase, making clear that: *a*) the features are highly dependent on the image content and *b*) unconstrained training/target sets are key for real conditions use. Another remarkable fact is that these features are almost perfectly separable, as we systematically obtained almost 100% accuracy on the training set.

For the Gaussian mixture models, estimation of the number of components in each mixture is required. By constraining the number of components to be no larger than  $G_{max} = 5$ , we need to choose among  $G_{max}^K$  possible configurations; this has been carried out by a standard 5-fold cross-validation criterion.

### 5.3. Classifier testing

Classification accuracy on the test sets for single patches, for all cameras ( $K = 4$ ) and for all binary classification problems ( $K = 2$ ), are presented in Table 1. The first column represents single Gaussian results, whereas the second shows those obtained using Gaussian mixtures, averaged over 100 trials (for 100 random initializations of the EM algorithm). The last column contains the standard deviations of the latter results. Results have been divided in three blocks, according to the training set size. Figure 2 shows a bar plot with data from the central block of the table. We observe that the single Gaussian classifiers achieve results close to those of the mixture models. One may think that this could be related to the number of parameters to estimate versus the number of samples, but we see that this behavior is homogeneous for all training set sizes. This fact suggests that the single Gaussian is a better option due to its simplicity.

In terms of patch combination methods for classifying whole images (see Section 4.2) we compare the performance of the two options (voting versus model-based) in Fig. 3 (both with 30% samples for training). These results suggest that the voting scheme achieves better performance, which may be explained by the presence of outlier regions, which can severely affect the model-based strategy, while the voting scheme remains almost unaffected. Regarding the behavior of the models, the mixture model systematically outperformed the single Gaussian model, when using the voting scheme,

<sup>4</sup>We are very thankful to Albert Sierra for letting us download and make use of his personal pictures (<http://www.albertsierra.fotki.com>).

Model	Gaussian	Gaussian mixture	
<i>Training set: 20% of the samples</i>			
Comparative	Hits	Hits (average)	$\sigma_p$
Nikon Vs Canon	67.94%	95.56%	1.23%
Nikon Vs Konica	76.70%	75.83%	1.33%
Nikon Vs Olimpus	69.97%	69.30%	1.07%
Canon Vs Konica	85.06%	84.14%	0.98%
Canon Vs Olimpus	67.94%	67.08%	2.51%
Konica Vs Olimpus	76.54%	73.13%	3.66%
All cameras	62.78%	63.28%	0.96%
<i>Training set: 30% of the samples</i>			
Comparative	Hits	Hits (average)	$\sigma_p$
Nikon Vs Canon	74.03%	72.87%	1.2%
Nikon Vs Konica	80.43%	80.01%	0.84%
Nikon Vs Olimpus	67.90%	69.35%	1.22%
Canon Vs Konica	86.83%	86.83%	0%
Canon Vs Olimpus	68.37%	68.58%	1.38%
Konica Vs Olimpus	73.47%	73.47%	0%
All cameras	65.82%	65.65%	1.34%
<i>Training set: 40% of the samples</i>			
Comparative	Hits	Hits (average)	$\sigma_p$
Nikon Vs Canon	70.56%	68.70%	2.09%
Nikon Vs Konica	80.95%	80.34%	1.58%
Nikon Vs Olimpus	79.55%	78.19%	2.55%
Canon Vs Konica	86.26%	85.81%	0.72%
Canon Vs Olimpus	84.74%	84.04%	1.49%
Konica Vs Olimpus	90.26%	89.42%	0.93%
All cameras	73.05%	73.30%	1.69%

**Table 1.** Test set accuracy (per image blocks); see text for details.

whereas when using the product rule, there is no significant difference.

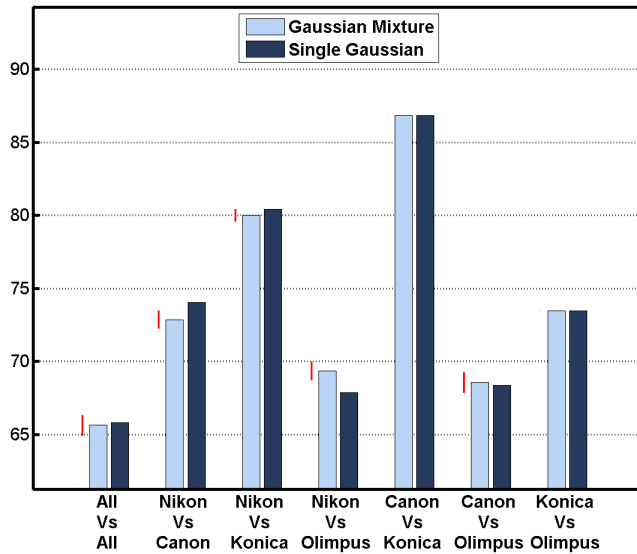
## 6. CONCLUSIONS

We have described a Bayesian classification approach for identifying the source camera model from acquired images. We have verified that the traces of CFA interpolation obtained through a least-squares linear fitting allow us to distinguish among source camera models, even if the images used for training and testing are not constrained by capture conditions or scenery. We empirically observed that the features are highly dependent on the image content and that, for the parameter setup used, they can be considered as perfectly separable. We compare two different camera models: single Gaussian vs. Gaussian mixture, concluding that the complexity of the latter does not pay off in terms of performance. Regarding the decision rule, in the absence of outlier detection, we conclude that the voting scheme (ad-hoc) is more robust and effective.

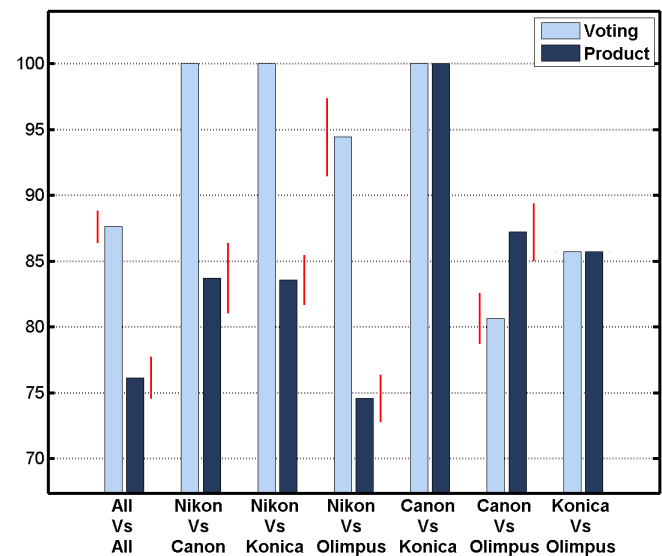
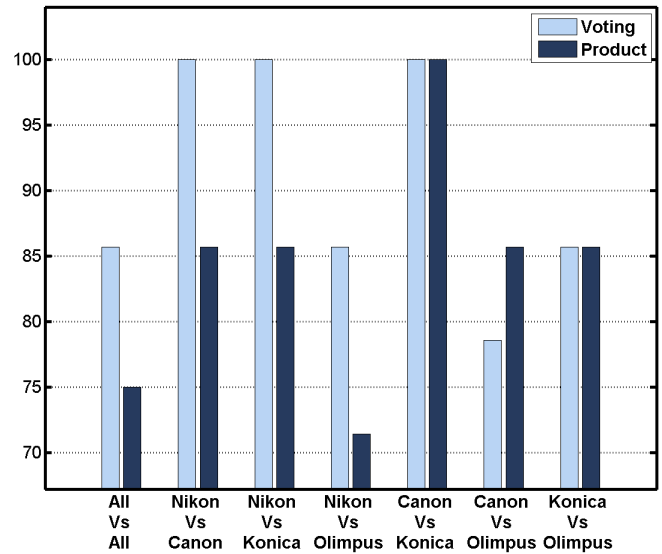
As future work, we believe that there is still potential for improvement: *a*) by developing an outlier detection procedure and *b*) by carefully inspecting and modelling the dependence features-image content.

## 7. REFERENCES

- [1] S. Katzenbeisser and F. Petitcolas, *Information techniques for Steganography and digital watermarking*, Artech House, 2000.
- [2] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking*, Morgan Kaufmann, 2002.



**Fig. 2.** Comparison of the test set accuracy (per image blocks) of the Gaussian and Gaussian mixture models. Data is taken from central block of Table 1. In the mixture case, the vertical red lines show the standard deviation over 100 repetitions.



**Fig. 3.** Comparison of the test set accuracy (per images) of the voting and model based combination rules described in Section 4.2. In the mixture case, the vertical red lines show the standard deviation over 100 repetitions.

[3] “Special issue on data hiding,” *IEEE Trans. Signal Proc.*, vol. 41, 2003.

[4] K. Kurosawa, K. Kuroki, and N. Saitoh, “CCD fingerprint method-identification of a video camera from videotaped images,” in *IEEE Int’l Conf. on Image Proc.*, 1999, vol. 3, pp. 537–540.

[5] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh, “Methods for identification of images acquired with digital cameras,” in *Proc. SPIE, Enabling Technologies for Law Enforcement and Security*, February 2001, vol. 4232, pp. 505–512.

[6] J. Lukas, J. Fridrich, and M. Goljan, “Digital camera identification from sensor pattern noise,” *IEEE Trans. Info. Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.

[7] M. Kharrazi, H. T. Sencar, and N. D. Memon, “Blind source camera identification,” in *IEEE Int’l Conf. on Image Proc.*, 2004, pp. 709–712.

[8] C. Chang and C. Lin, *LIBSVM: a library for support vector machines*, 2001, Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

[9] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, “Source camera identification based on CFA interpolation,” in *IEEE Int’l Conf. on Image Proc.* Genoa, Italy, September 2005, vol. 3, pp. 69–72.

[10] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, “Improvements on source camera-model identification based on CFA interpolation,” in *International Conference on Digital Forensics*, February 2006.

[11] A.C. Popescu and H. Farid, “Exposing digital forgeries by detecting traces of re-sampling,” *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 758–767, 2005.

[12] A.C. Popescu and H. Farid, “Exposing digital forgeries in color filter array interpolated images,” *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3948–3959, 2005.

[13] J. Portilla, D. Otaduy, and C. Dorransoro, “Low-complexity linear demosaicing using joint spatial-chromatic image statistics,” in *IEEE Int’l Conf. on Image Proc.*, 2005, pp. 61–64.

[14] J. Portilla, C. Dorransoro, and D. Otaduy, “Método de predicción lineal óptima para la reconstrucción de la imagen en cámaras digitales con sensor de mosaico,” 2005, Patent, P200502100.

[15] B. E. Bayer, “Color imaging array,” 1976, US Patent, 3971065.

[16] G. McLachlan and D. Peel, *Finite Mixture Models*, Wiley, 2000.

[17] DPreview, “Web page,” <http://www.dpreview.com>.

## S.3. Conclusions and future work

### S.3.1. Conclusions

One of the main goals of this Thesis was to develop new statistical image models to improve what was here our starting point, i.e., the original Gaussian scale mixture (GSM) local model as described in [10]. In this Thesis we have discussed about the inherent limitations of the latter model when trying to capture local spectral features with only one covariance matrix (up to scale factor) per subband. We have presented two new image models that make use of GSM but following very different approaches.

The first proposed model, based on the estimation of local statistics, characterizes the image subbands using spatially variant GSMs (SVGSM [11, 16]). This conceptually simple extension provides a significant improvement in terms of local features adaptation, which is strongly related to the covariance spatial variability. The presented denoising algorithm (based on SVGSM), which inherits the good features of its predecessor, provides a generalized improvement, definitely paying off a moderate increase of computational cost. We have tested this method with two different and complementary representations: Full Steerable Pyramid (FSP) [10] and Translation Invariant Haar Pyramid (TIHP) [16]. We conclude that no representation is better than the other in absolute terms, as the performance depends on the image content. Texture rich images should be processed using FSP, whereas those having more flat regions (especially if having strong vertical or horizontal content) are better processed with TIHP. However, from a computational cost point of view, TIHP is a better choice, because its significant smaller redundancy factor results in much fewer subbands to process. Finally, as the performance of this method is related to covariance spatial variability, we have shown that the relative improvement of SVGSM over classical GSM is higher for TIHP because, in contrast to FSP, it provides subbands with high local spectral variability.

The limitations of the original GSM model become more evident when dealing with more difficult tasks, such as deconvolution under additive Gaussian noise [19]. In this work, we have dealt with the severe constraints imposed by local models when trying to compensate for certain types of degradations, even if the models are used under multi-resolution representations. We have proposed a solution [12, 16] based on a two-step scheme: global linear deconvolution, first, and then non-linear local denoising (based on SVGSM). This strategy allows us to overcome some of the limitations inherent to local models while keeping their strengths. Despite we have coupled these two steps in an empirical way, the proposed procedure is flexible and simple, providing a solution that, in contrast to other state-of-the-art methods, is robust against errors in the power spectral density estimation and against different types of blurring kernels. All these facts, coupled with



the moderated (and image-content independent) computational cost, make this method suitable to restore images in real situations, as we have shown in [15, 14].

Despite its good properties, SVGSM does not exploit the non-local redundancy present in natural images. In addition it is severely affected by the sampling scheme (blocks). These facts motivated the development of the other image model presented in this Thesis [18]. As we said, the original local approach is still kept, but now using the support of a global statistical characterization for each subband. The model, based on a mixture of GSMs (MGSM), allows us to capture different local spectral features within each subband, significantly increasing the adaptation, whereas exploiting the non-local redundancy existing among non-adjacent patches of similar content. Our solution is conceptually clean and the parameter estimation algorithm iteratively searches for the maximum likelihood solution in a generalized Expectation-Maximization fashion. This procedure implies an important difference with respect to original GSM and SVGSM, as there is no longer need for priors on the hidden scaling variables, because now they are adaptively ML-estimated for each subband. The MGSM model is very effective at describing image statistics. This makes it extraordinarily powerful for denoising, where it provides a significant and generalized improvement over our previous GSM models. We believe that the MGSM model represents a potentially significant contribution to the state of the art, being the version presented here a first step towards an elegant and flexible model which is able to provide a new, more favorable, tradeoff between bias and adaptation capability.

The other goal of this Thesis was to develop a procedure to identify cameras using a single observed image. For that purpose we used a Bayesian classifier characterizing cameras with two simple statistical models: a single Gaussian and a Gaussian mixture, both modelling the distributions of some previously proposed feature vectors [20, 21]. The used features are based on demosaicing interpolation filters, which are different for each camera model. Here, in contrast to other state-of-the-art methods, we have emphasized in applying a very rigorous methodology, and, thus, our training and target sets have not been restricted at all. The reasonably high performance obtained, even under these unconstrained conditions, encourages us to consider the method as potentially applicable to real situations.

Summarizing, this work represents yet another proof of the flexibility and strength of the Bayesian paradigm and Gaussian mixtures (continuous and discrete) as universal tools for developing applications and models in the digital image processing field. In particular, we have directly and widely experienced that their combination provides excellent results, both for estimation and classification tasks.

### S.3.2. Future work

We believe that the most immediate source of future research comes from a further development of the new MGSM model. Despite its good behavior, the model is still at preliminar stages and deeper studies are required. The main limitation is the computational cost: the current version is severely affected by a strong computational load that is roughly one magnitude order above our local GSM reference. Another interesting aspect to improve is related to the number of mixture components (GSMs) used at each subband. By adaptively setting this parameter, which is strongly related to the subband content, we expect an improvement both in performance and efficiency. A similar gain may arise by adaptively estimating the mixing scale components of each GSM. Another critical issue is the way the neighborhood samples are used in the MGSM model. Currently, we use all possible (overlapping) neighborhoods within subbands. We would like to consider two different alternatives. The first one is to include an elaborated pruning procedure to significantly decrease the overlapping among neighborhoods. This approach, in addition to the significant reduction of the computational cost and the strong increase of the average likelihood (and, hence, of the description power of the model), will make the method sensitive to local phase. This would be at the price of losing a big part of the redundancy, a key advantage for the estimation accuracy. From an opposite point of view, the second approach consists of studying the potential redundancy of structures when applying to the neighborhoods simple operations such as rotations and/or scalings. This option will raise the redundancy at the price of significantly increasing the computational cost. Finally we would like to explore other estimation criteria alternative to Maximum Likelihood, such as "Score Matching" [22, 23].

On the application side, we would like to explore new image processing methods for the MGSM model, such as deconvolution under additive Gaussian noise or texture analysis. Concerning the forensic application, we want to increase the robustness, efficiency and hit rate of the presented camera identification algorithm by: (a) decoupling the detected dependence between the image content and the feature vectors, (b) reducing the dimensionality of the descriptors, and (c) using more elaborated classifiers and possible combinations among them.



## Parte II: Versión en español



# Resumen

El paradigma Bayesiano es un marco de trabajo de gran utilidad en la mayoría de las aplicaciones relacionadas con el procesamiento de imágenes. Particularmente, su utilización en el ámbito de la restauración (vía estimación) se encuentra fuertemente ligada a la existencia de modelos estadísticos que representen el conocimiento *a priori* que poseemos sobre las imágenes naturales. Precisamente, el estudio de las propiedades estadísticas de las imágenes ha motivado la proliferación de nuevas representaciones capaces de capturar dichas propiedades. Entre ellas, las representaciones piramidales sobre-completas han supuesto un gran impulso en el rendimiento de los modelos y sus aplicaciones de restauración. Por esta razón, una de las tendencias actuales es desarrollar modelos estadísticos en este dominio. Al comienzo de esta Tesis (Capítulo 2), se hace hincapié en la importancia que tienen dichos modelos en el rendimiento de las aplicaciones, describiendo una de las referencias actuales basada en una mezcla en escala de Gaussianas (GSM). Tomando como punto de partida la citada referencia, presentamos dos nuevos modelos estadísticos que son aplicados a la eliminación de ruido (Capítulos 3 y 4). Estos modelos, aun compartiendo una base común (GSM), atacan el problema desde perspectivas muy diferentes: caracterización local vs. caracterización global. Otra interesante aplicación de restauración es la compensación de emborronamiento en presencia de ruido aditivo Gaussiano, problema este, de mayor complejidad. En este trabajo (Capítulo 5) hemos presentado una solución basada en un esquema de dos pasos: (a) compensación lineal de emborronamiento (global), y (b) eliminación no lineal de ruido (local). De esta forma, el problema es descompuesto en un filtrado global y en una eliminación de ruido Gaussiano no blanco, reduciendo considerablemente la dificultad del problema. Además, esta estrategia nos permite aplicar y explotar los potentes métodos de eliminación de ruido actuales (como por ejemplo, los presentados en esta Tesis).

Las herramientas Bayesianas también son muy utilizadas para abordar problemas de clasificación. En el Capítulo 6 de esta Tesis proponemos dos modelos estadísticos de un descriptor de cámaras propuesto recientemente. Dichos modelos son utilizados, dentro del marco Bayesiano, para desarrollar un algoritmo de identificación de cámaras digitales.

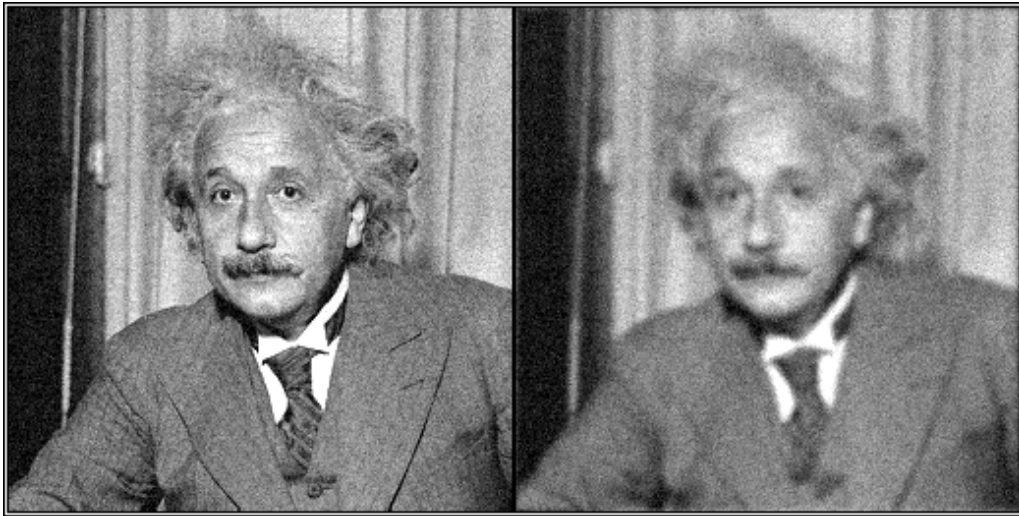


# Capítulo 1

## Introducción

Los nuevos avances de la era digital han convertido a las imágenes, incluso más que en el pasado, en uno de los vehículos de información más importantes. En un corto periodo de tiempo, los dispositivos de adquisición digital han dejado de ser solo accesibles para un número (relativamente) reducido de usuarios, profesionales principalmente, convirtiéndose en elementos tan comunes como el teléfono móvil. La industria ha inundado el mercado con estos dispositivos, compitiendo por proporcionar cada vez más aplicaciones y mejor calidad a un menor coste. Estos hechos han motivado que el procesamiento digital de imágenes sea uno de los campos más tratados hoy en día. La ubicuidad de las imágenes motiva la proliferación de aplicaciones en entornos muy dispares y con objetivos muy variados: codificación, compresión, seguimiento, identificación, restauración, fusión, clasificación... son solo un pequeño ejemplo en el extenso campo del procesamiento de las imágenes digitales [1]. En la literatura existen gran cantidad de soluciones a estos problemas. Aunque muchas de ellas siguen una aproximación ad-hoc, cada vez está más extendido el uso de modelos estadísticos que proporcionan soporte a dichas aplicaciones. Desde el punto de vista de la información involucrada, esta tendencia se explica por sí sola, ya que la información de entrada de la mayoría de las aplicaciones tiene como común denominador a las *imágenes naturales*, entendiendo por imágenes naturales a aquellas que provienen del mundo real en la forma en que habitualmente lo observamos. El frecuente uso de los modelos estadísticos también puede ser explicado considerando la forma de utilizar el conocimiento sobre las imágenes naturales. Desde este punto de vista, la razón podría estar en el extendido uso del paradigma Bayesiano, que hace uso explícito de dicho conocimiento a priori. Además, este paradigma puede ser utilizado en la gran mayoría de los problemas de inferencia estadística. Esta flexibilidad hace mucho más atractivo el uso de modelos estadísticos, que pueden ser utilizados en una gran variedad de aplicaciones.

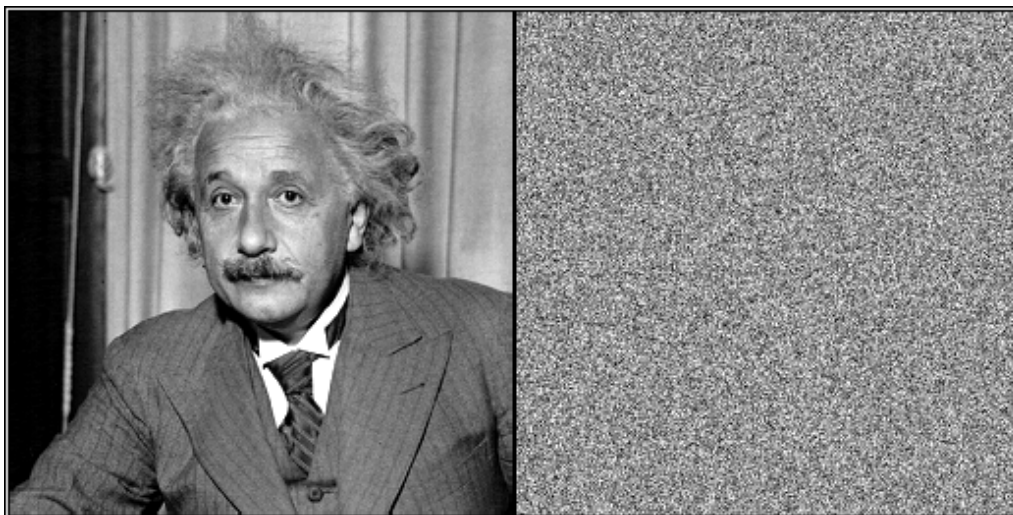




**Figura 1.1.** Degradaciones típicas en problemas de restauración. A la izquierda la imagen de *Einstein* afectada por ruido aditivo. A la derecha la misma imagen afectada por emborronamiento y ruido.

Particularmente, el paradigma Bayesiano es muy utilizado en el área de la restauración de imágenes. El objetivo de este tipo de aplicaciones es obtener una *estimación* de una imagen ideal a partir de una versión degradada de la misma. Existen una gran variedad de ámbitos (doméstico, científico-técnico, industrial, militar, policial, judicial, etc...) en los que el procesamiento de imágenes degradadas es útil, no solo para mejorar su aspecto visual, sino también para facilitar la interpretación de sus contenidos, ya sea por personas o por computadoras. La degradación a la que se ve sometida una imagen debe ser entendida en sentido amplio, pudiendo representar por ejemplo, pérdida de información (píxeles "muertos"), emborronamiento, contaminación por adición de ruido, emborronamiento en presencia de ruido, etc... En la Figura 1.1 se muestra una imagen natural afectada por las dos últimas degradaciones. A las aplicaciones que tratan de compensarlas se las conoce como eliminación de ruido (*denoising*, en inglés) y compensación de emborronamiento o deconvolución (*deblurring* o *deconvolution*, en inglés) respectivamente. Gran parte de esta Tesis está dedicada al desarrollo de estas aplicaciones. Como describiremos en detalle más adelante (Capítulo 2), el paradigma Bayesiano utiliza, a través de la conocida regla de Bayes, tanto la información sobre la degradación, como la información *a priori* disponible sobre las imágenes naturales. En este trabajo asumimos<sup>1</sup> que el proceso de degradación es perfectamente conocido y correctamente modelado. Bajo estas condiciones, el éxito de la restauración Bayesiana, cuando es medido a través de simulaciones que utilizan a las imágenes conocidas como referencias ideales, recae

<sup>1</sup>Esta asunción es habitualmente realizada en la literatura.



**Figura 1.2.** Comparación entre una imagen natural (*Einstein*) y una imagen generada aleatoriamente (ruido blanco Gaussiano).

principalmente en la existencia de un buen modelo estadístico de las imágenes naturales.

Para construir modelos de imagen que permitan distinguir a la señal (imagen) de la degradación (ruido, p.e.) cabe hacerse la siguiente pregunta: ¿Qué características diferencian a las imágenes naturales? La Figura 1.2 muestra una imagen natural (izquierda) junto a una imagen sintética de ruido Gaussiano (derecha). Una simple inspección visual nos permite extraer diferencias significativas, como por ejemplo la presencia de zonas estructuradas y suaves en las imágenes naturales, en contraste con la aleatoriedad no estructurada de las imágenes artificiales. Las imágenes naturales han sido ampliamente analizadas en la literatura, sacando a relucir muchas propiedades particulares. Tres de las más importantes se enumeran a continuación [2, 3]:

- Alta correlación entre los píxeles vecinos.
- Presencia de patrones locales orientados, como bordes, barras, esquinas, etc...
- Las propiedades estadísticas son, aproximadamente, invariantes a un cambio de escala.

Estas características mantienen una estrecha relación con los modelos biológicos visuales [4], motivada por la coincidencia entre las escenas adquiridas en las imágenes naturales y los estímulos del mundo real a los que el sistema visual de los seres vivos se ha adaptado a lo largo de millones de años. Las propiedades estadísticas de las imágenes, junto con las citadas relaciones, han motivado un

cambio en la representación, llevando a multitud de investigadores a proponer transformaciones inspiradas en el sistema visual humano (por ejemplo [5, 6]). Como veremos en el siguiente Capítulo, las wavelets críticamente muestreadas reducen extraordinariamente la alta correlación existente en el dominio píxel y proporcionan selectividad tanto a la escala como a la orientación, haciéndolas especialmente adecuadas para representar eficientemente a las imágenes naturales (JPEG2000 [7], p.e.). Sin embargo, en problemas de restauración, las representaciones piramidales sobre-completas son las que han proporcionado un mejor rendimiento (medido en términos de error cuadrático o MSE - del inglés *Mean Square Error*), añadiendo a la selectividad en escala/orientación dos ventajas determinantes: la invarianza a traslación y estimaciones más fiables gracias a la redundancia. Adicionalmente, Raphan *et al.* [8] han demostrado formalmente que la esperanza del error cuadrático medio en el dominio de la imagen proveniente de una representación redundante es menor o igual que el obtenido de una representación no redundante.

Lo comentado anteriormente explica que la tendencia actual sea elaborar modelos estadísticos de imagen en el dominio sobre-completo. De nuevo, volvemos a la pregunta que realizábamos anteriormente, pero con una importante modificación: ¿Qué características hacen diferentes a las imágenes naturales, pero ahora, en el dominio piramidal sobre-completo? La respuesta es obtenida, otra vez, a partir del estudio de las propiedades estadísticas de los coeficientes wavelet. Una de las referencias actuales es el modelo basado en mezclas en escala de Gaussianas [9] (GSM, del inglés *Gaussian Scale Mixtures*). Este modelo da soporte a aplicaciones de restauración muy exitosas, como por ejemplo la presentada en [10]. Esta última referencia representa el punto de partida de esta Tesis doctoral.

En otro orden de aplicaciones se encuentra el área del procesamiento forense de las imágenes digitales. Al igual que los avances científicos y técnicos han provocado un incremento considerable en las capacidades de los dispositivos de adquisición, el desarrollo de software de edición ha seguido un desarrollo similar. Por ello, en la actualidad, es relativamente sencillo (incluso para los no expertos) manipular el contenido de las imágenes digitales con fines muy dispares, como la eliminación/adición de contenido a la escena, introducción de información secreta (esteganografía), retoque, y un largo etcétera. En los últimos años, el procesamiento forense de las imágenes digitales ha despertado el interés tanto de la comunidad científica como del entorno de la seguridad y defensa, que aúnan esfuerzos para proporcionar aplicaciones de detección de manipulaciones, codificación/descodificación de mensajes ocultos, autenticación, etc...

La identificación de cámaras digitales representa otra interesante aplicación dentro del ámbito forense. El objetivo principal es determinar, a partir de una observación, el dispositivo de adquisición. De forma genérica, la clasificación Bayesiana proporciona una medida probabilística de la pertenencia de una

observación a una determinada clase. En los problemas de clasificación de imágenes la información de entrada es, de nuevo, una imagen natural. Sin embargo, en este caso, y en contraste con las aplicaciones de restauración, la información a *priori* no se corresponde con un modelo estadístico de la imagen, sino con un valor discreto que representa la probabilidad de cada cámara.

Ahora, el rendimiento de la aplicación recae principalmente en la selección de las características discriminantes y en el modelado estadístico de las características dada cada una de las clases (el equivalente al modelo de degradación en restauración) y no tanto en la (todavía relevante) información a priori. Siguiendo con el paralelismo, la pregunta a responder es ahora: ¿Qué características de las imágenes digitales minimizan el solapamiento entre las distribuciones de los descriptores para cada cámara? En esta Tesis utilizamos las trazas de los filtros de de-mosaico como características para describir a las cámaras. En el Capítulo 6 presentamos dos modelos estadísticos de las características (diferentes) que son usados para desarrollar un método Bayesiano de identificación de cámaras.

## 1.1. Objetivos y contribuciones de la Tesis

Los objetivos generales de esta Tesis están resumidos en el título de la misma: Se pretende utilizar el paradigma Bayesiano para obtener métodos de restauración de imágenes (vía estimación) y métodos de identificación de cámaras (vía clasificación). De forma un poco más concreta, los objetivos que deseamos cubrir son los siguientes:

1. Proponer nuevos modelos estadísticos de las imágenes naturales en dominios piramidales sobre-completos que superen algunas de las limitaciones de nuestro modelo de referencia (GSM local [10]), incrementando sus fortalezas.
2. Utilizar los nuevos modelos de imagen junto con las herramientas de estimación Bayesiana para construir aplicaciones de restauración, tal como eliminación de ruido y compensación de emborronamiento en presencia de ruido.
3. Obtener modelos estadísticos de descriptores de cámaras digitales que, junto con el uso de herramientas de clasificación Bayesiana, permitan la construcción de un algoritmo de identificación de cámaras a partir de sus imágenes.

La Figura 1.3 muestra una visión general de los contenidos presentados en esta Tesis, poniendo en contexto las contribuciones que hemos realizado (enmarcadas con doble línea roja), así como las relaciones existentes entre ellas. Como podemos observar, los objetivos quedan cubiertos por los Capítulos 3, 4, 5 y 6. A continuación se describe el contenido de cada Capítulo.

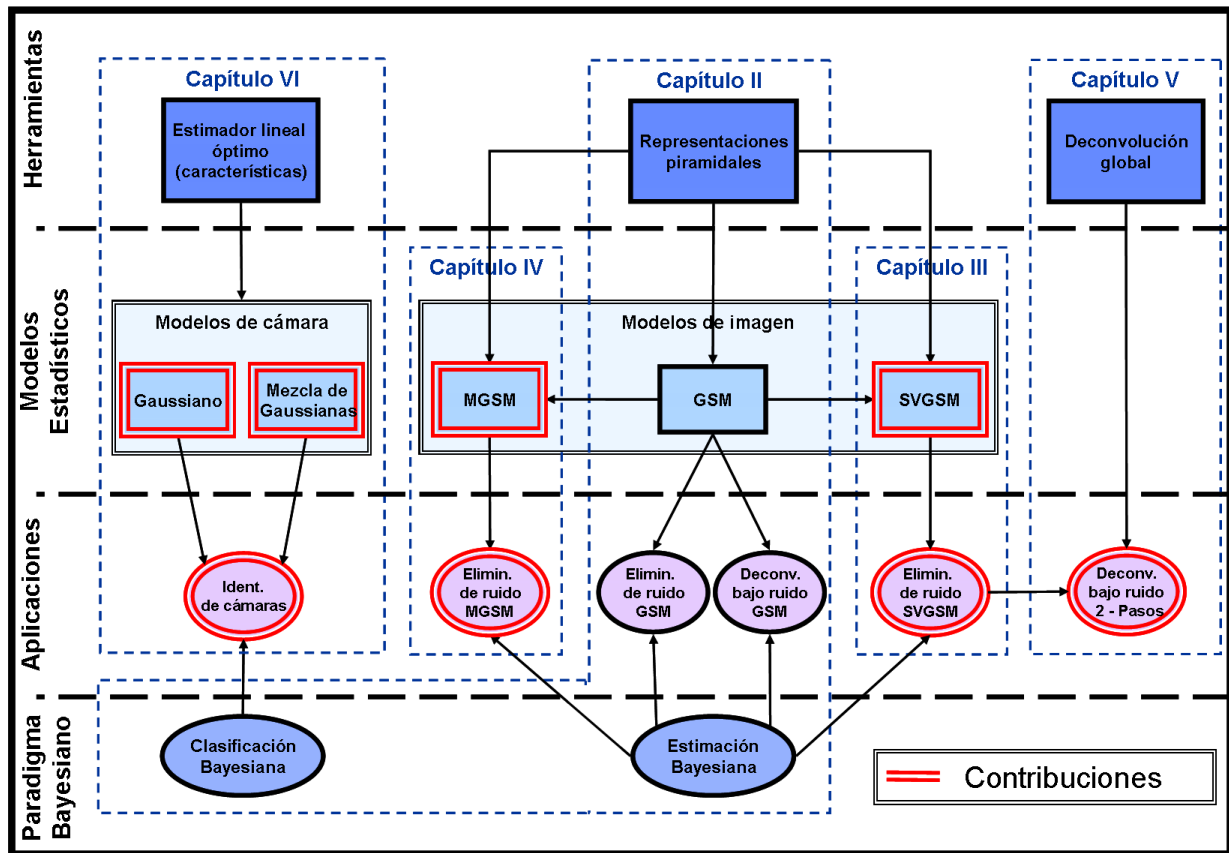


Figura 1.3: Esquema de contenidos.

El Capítulo 2 pretende poner en antecedentes al lector presentando una serie de herramientas y conceptos básicos importantes, así como una detallada descripción del modelo GSM, punto de partida de esta Tesis y de gran importancia para la comprensión de la mayoría de los contenidos tratados en la misma. Los Capítulos 3 y 4, núcleo central de este trabajo, muestran dos nuevos modelos estadísticos de imágenes que son utilizados para desarrollar dos nuevos algoritmos de eliminación de ruido. Si bien ambos modelos están basados en el modelo GSM local [10], las propuestas corresponden a dos puntos de vista muy diferentes, una de ellas basada en estadísticos locales (Capítulo 3), frente a otra basada en estadísticos no locales (Capítulo 4). El siguiente Capítulo presenta un algoritmo de restauración (compensación de emborronamiento bajo ruido) basado en un esquema de dos pasos: (a) compensación lineal de emborronamiento (global), y (b) eliminación no lineal de ruido (local). En el Capítulo 6 se presentan dos sencillos modelos de cámaras que, junto con el paradigma de clasificación Bayesiano, nos permite construir el algoritmo de identificación de cámaras que cubre el último de los

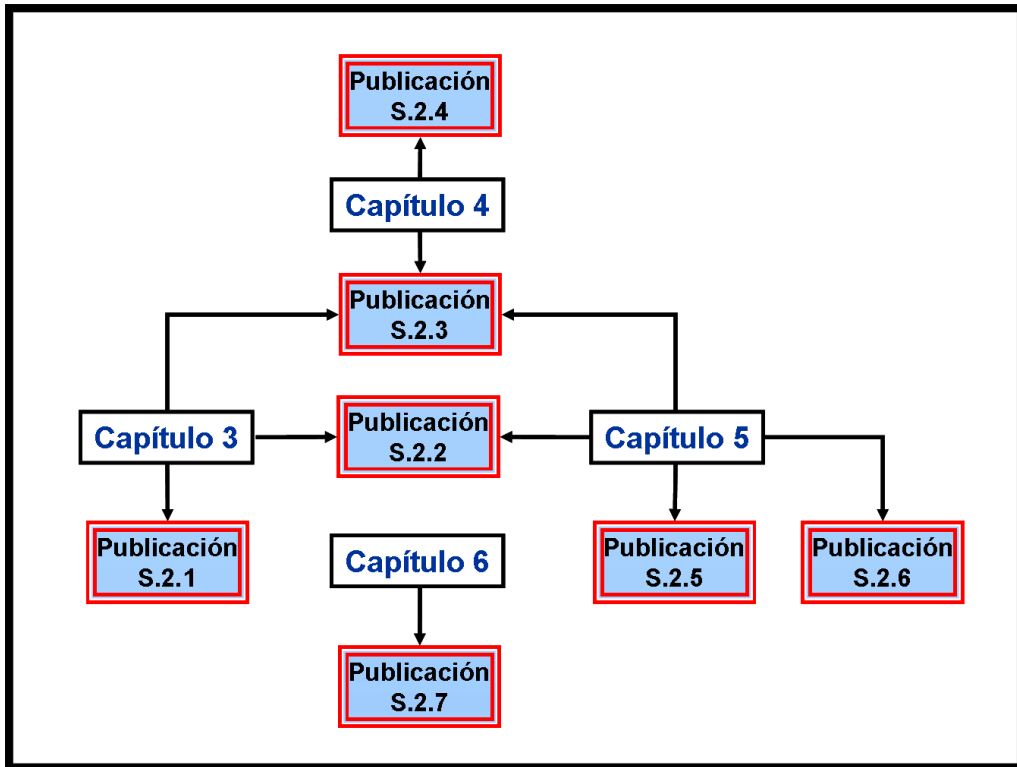


Figura 1.4: Relaciones entre publicaciones y Capítulos.

objetivos deseados. Finalmente, el Capítulo 7 cierra esta Tesis, presentando las conclusiones generales obtenidas y algunas de las líneas de trabajo que quedan abiertas para el futuro.

La consecución de los objetivos planteados en esta Tesis han dado su fruto en forma de publicaciones científico-técnicas. En el Apéndice B podemos encontrar la relación de publicaciones que dan soporte a esta Tesis doctoral. En la Sección S.2, perteneciente a la versión en inglés (resumida) de este trabajo, hemos incluido una copia de las mismas. Entre ellas se encuentran cuatro comunicaciones a congresos [11, 12, 13, 14], una patente [15] y un artículo de revista internacional [16], todas ellas publicadas. A estas últimas se suman dos comunicaciones a congreso que se encuentran en proceso de revisión [17, 18].

En la Figura 1.4 presentamos las relaciones entre las publicaciones y las contribuciones de esta Tesis agrupadas por Capítulos (3 al 6). Como podemos ver, para cada Capítulo, incluimos una comunicación a congreso que resume el contenido del mismo, con la excepción del Capítulo 5 que cuenta con una publicación adicional (disponible en la Sección S.2.6) describiendo una interesante aplicación óptica de nuestro algoritmo de restauración. El contenido de los Capítulos 3 y 5 está completamente cubierto por nuestra publicación de revista

internacional incluida en la Sección S.2.2. Además, estos contenidos son tratados de forma más resumida en la publicación de la Sección S.2.3, que también incluye una versión simplificada del nuevo modelo estadístico propuesto en el Capítulo 4 (MGSM).

## Capítulo 2

# Fundamentos

### 2.1. Introducción

Este Capítulo pretende describir al lector conceptos y herramientas que se utilizarán a lo largo de esta Tesis. Comenzamos el mismo abordando una serie de consideraciones estadísticas acerca de las imágenes naturales. Tras ello, se presenta el paradigma Bayesiano como marco fundamental en la resolución de una gran variedad de problemas relacionados con el tratamiento de imágenes, prestando especial atención a la restauración y la clasificación. La siguiente Sección analiza un conjunto de transformaciones lineales, que van desde la transformada de Fourier hasta representaciones piramidales sobre-completas. Por último, se describe en detalle el trabajo que representa el punto de partida de esta Tesis, un modelo local de señal basado en mezclas en escala de Gaussianas (GSM, del inglés, *Gaussian Scale Mixtures*) y dos aplicaciones de restauración basadas en el mismo.

### 2.2. Campos aleatorios homogéneos

Las imágenes naturales poseen características estadísticas que, en una primera aproximación, pueden considerarse independientes de las coordenadas espaciales en el plano de la imagen. Gracias a ello podemos modelarlas utilizando campos aleatorios homogéneos (estacionarios). La autocorrelación de estos modelos se define como:

$$R(\mathbf{p}_1, \mathbf{p}_2) = E\{x(\mathbf{p}_1) \cdot x(\mathbf{p}_2)\},$$

donde  $\mathbf{p}_1$  y  $\mathbf{p}_2$  son dos puntos en el plano de la imagen. En los campos estacionarios [24] los estadísticos son independientes de las coordenadas espaciales,



solo dependen de la posición relativa entre los dos puntos. Por ello:

$$\begin{aligned} R(\mathbf{p}_1, \mathbf{p}_2) &= r(\mathbf{p}_1 - \mathbf{p}_2) \\ \boldsymbol{\tau} &= \mathbf{p}_1 - \mathbf{p}_2 \\ R(\mathbf{p}_1, \mathbf{p}_2) &= r(\boldsymbol{\tau}) = r(-\boldsymbol{\tau}). \end{aligned}$$

Con lo que cada elemento de la matriz de correlaciones se puede calcular de la siguiente forma:

$$c_{i,j} = R(\mathbf{p}_i, \mathbf{p}_j) = r(\mathbf{p}_i - \mathbf{p}_j) = r_{|i-j|}.$$

La matriz resultante es simétrica. Si además consideramos que la imagen se extiende periódicamente entonces  $\mathbf{p}_{n+j} = \mathbf{p}_j$  y la autocorrelación  $r_{n-j} = r_{-j} = r_j$  con lo que la matriz resultante queda como sigue:

$$\begin{aligned} \mathbf{C}_x &= \begin{pmatrix} r_0 & r_1 & \dots & r_{n-1} \\ r_1 & r_0 & \dots & r_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n-1} & r_{n-2} & \dots & r_0 \end{pmatrix} \\ &= \begin{pmatrix} r_0 & r_1 & \dots & r_1 \\ r_1 & r_0 & \dots & r_2 \\ \vdots & \vdots & \ddots & \vdots \\ r_1 & r_2 & \dots & r_0 \end{pmatrix}. \end{aligned}$$

Esta matriz es una matriz Toeplitz circulante por bloques<sup>1</sup>. Este tipo de matrices (circulantes) aplicadas sobre un vector representan una convolución sobre la señal periódica. Sus autovectores son exponenciales complejas ( $\cos w_n + j \sin w_n$ ) y sus autovalores son la transformada discreta de Fourier de la función de autocorrelación  $r$ , es decir, la densidad espectral de potencia (PSD, del inglés Power Spectral Density) del campo aleatorio estacionario extendido de forma periódica.

### 2.3. Paradigma Bayesiano

El paradigma Bayesiano proporciona un marco estadístico de trabajo que es especialmente adecuado para el tratamiento de los problemas relacionados con el procesamiento de imágenes. La mayoría de las aplicaciones pueden plantearse como problemas de inferencia estadística, donde a partir de una observación se pretende inferir una determinada información. Podemos identificar esta inferencia con múltiples tareas: restauración, detección, seguimiento, clasificación, etc. Los trabajos realizados en esta Tesis se centran en las aplicaciones de restauración y clasificación, que siguiendo un enfoque Bayesiano pueden describirse como:

<sup>1</sup>Por simplicidad, se muestra una matriz circulante para el caso de una señal 1D.

- Restauración: A partir de una imagen observada  $\mathbf{y}$ , que es una versión degradada (ruidosa y/o emborronada, en nuestro caso) de una original  $\mathbf{x}$ , se desea obtener una estimación de esta última.
- Clasificación: Dada una observación  $\mathbf{y}$ , que corresponde, en general, a un vector que caracteriza una imagen, y un conjunto de  $K$  clases, se pretende inferir la clase a la que pertenece dicha observación con mayor probabilidad.

Para resolver estos problemas se utiliza:

- El conocimiento a priori:  $p(\mathbf{x})$ . En el caso de la restauración, proporciona información estadística de cómo son típicamente las imágenes naturales. En el caso de clasificación nos indica cómo de probable es cada clase considerada.
- El conocimiento del proceso observado:  $p(\mathbf{y}|\mathbf{x})$ . En el caso de la restauración proporciona información sobre el proceso de degradación. En el caso de la clasificación, nos proporciona una caracterización estadística de los elementos de cada clase por separado.

El paradigma Bayesiano utiliza ambas fuentes de información para obtener una inferencia a través de la regla de Bayes:

$$p(\mathbf{x} | \mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}. \quad (2.1)$$

Podemos obtener nuestra estimación/elección de acuerdo a varios criterios, entre ellos los más usados son:

- MAP: Maximizando la probabilidad a posteriori:

$$\hat{\mathbf{x}}_{MAP} = \arg \max_{\mathbf{x}} \{p(\mathbf{x}|\mathbf{y})\}. \quad (2.2)$$

- BLS: Obteniendo la solución de mínimo error cuadrático medio esperado (*Bayes Least Squares*), que coincide con la media del posterior:

$$\hat{\mathbf{x}}_{BLS} = E_{\mathbf{x}}\{\mathbf{x}|\mathbf{y}\}. \quad (2.3)$$

En el caso particular de modelos Gaussianos, los estimadores MAP y BLS coinciden, por coincidir la moda y la media de una distribución Gaussiana. En el siguiente apartado describimos dos ejemplos clásicos de aplicación Bayesiana: el filtrado de Wiener y el clasificador lineal.

### 2.3.1. Estimación Bayesiana del filtro de Wiener

Asumiendo que el modelo de la señal sigue una distribución Gaussiana de media nula y que el ruido, independiente de la señal, es Gaussiano, podemos obtener expresiones del filtrado de Wiener utilizando la estimación MAP (2.2). A continuación se muestran las soluciones para dos problemas de restauración<sup>2</sup>:

#### Eliminación de ruido

El objetivo de la eliminación de ruido es obtener una estimación del vector original,  $\hat{\mathbf{x}}$ , a partir de una observación ruidosa  $\mathbf{y}$ . Podemos describir el modelo de observación como sigue:

$$\mathbf{y} = \mathbf{x} + \mathbf{w},$$

donde  $\mathbf{x}$  representa a la señal y  $\mathbf{w}$  al ruido. Ambos son vectores Gaussianos  $N$ -dimensionales de media cero y con covarianzas  $\mathbf{C}_{\mathbf{x}}$  y  $\mathbf{C}_{\mathbf{w}}$  respectivamente. Maximizar  $p(\mathbf{x}|\mathbf{y})$  con respecto a  $\mathbf{x}$  es equivalente a maximizar  $p(\mathbf{y}|\mathbf{x})p(\mathbf{x})$ , el numerador de la Ecuación 2.1. En este caso, el proceso de degradación posee la siguiente distribución:

$$p(\mathbf{y}|\mathbf{x}) = \frac{\exp(-(\mathbf{y} - \mathbf{x})^T \mathbf{C}_{\mathbf{w}}^{-1} (\mathbf{y} - \mathbf{x})/2)}{((2\pi)^N |\mathbf{C}_{\mathbf{w}}|)^{1/2}},$$

mientras que el modelo de la señal es simplemente:

$$p(\mathbf{x}) = \frac{\exp(-\mathbf{x}^T \mathbf{C}_{\mathbf{x}}^{-1} \mathbf{x}/2)}{((2\pi)^N |\mathbf{C}_{\mathbf{x}}|)^{1/2}}.$$

De esta forma, maximizar el producto de ambos modelos es equivalente, al aplicar logaritmos, a minimizar la siguiente expresión:

$$\hat{\mathbf{x}}_{MAP} = \arg \min_{\mathbf{x}} \{C(\mathbf{x})\} = \arg \min_{\mathbf{x}} \{(\mathbf{y} - \mathbf{x})^T \mathbf{C}_{\mathbf{w}}^{-1} (\mathbf{y} - \mathbf{x}) + \mathbf{x}^T \mathbf{C}_{\mathbf{x}}^{-1} \mathbf{x}\}.$$

Para ello derivamos con respecto a  $\mathbf{x}$  e igualamos a cero:

$$\begin{aligned} \frac{d}{d\mathbf{x}} C(\mathbf{x}) &= \frac{d}{d\mathbf{x}} (\mathbf{x}^T (\mathbf{C}_{\mathbf{w}}^{-1} + \mathbf{C}_{\mathbf{x}}^{-1}) \mathbf{x} - 2\mathbf{y}^T \mathbf{C}_{\mathbf{w}}^{-1} \mathbf{x} + \mathbf{y}^T \mathbf{C}_{\mathbf{w}}^{-1} \mathbf{y}) \\ &= 2(\mathbf{C}_{\mathbf{w}}^{-1} + \mathbf{C}_{\mathbf{x}}^{-1}) \mathbf{x} - 2\mathbf{C}_{\mathbf{w}}^{-1} \mathbf{y} = 0, \end{aligned} \quad (2.4)$$

que, tras despejar, obtenemos:

$$\hat{\mathbf{x}} = (\mathbf{C}_{\mathbf{w}}^{-1} + \mathbf{C}_{\mathbf{x}}^{-1})^{-1} \mathbf{C}_{\mathbf{w}}^{-1} \mathbf{y}. \quad (2.5)$$

Operando en la expresión anterior (2.5) resulta:

$$\hat{\mathbf{x}} = \mathbf{C}_{\mathbf{x}} (\mathbf{C}_{\mathbf{x}} + \mathbf{C}_{\mathbf{w}})^{-1} \mathbf{y}, \quad (2.6)$$

<sup>2</sup>Todos los vectores implicados (1-D) se han obtenido aplicando un orden lexicográfico a los píxeles de las imágenes (2-D), como es habitual.

que es la expresión del filtro de Wiener en forma vectorial. Al ser las matrices de covarianza Toeplitz circulantes, podemos aplicar la transformada discreta de Fourier (DFT, del inglés *Discrete Fourier Transform*), obteniendo el filtro de Wiener en el dominio de la frecuencia:

$$\hat{X}(\mathbf{f}) = \frac{P_x(\mathbf{f})}{P_x(\mathbf{f}) + P_w(\mathbf{f})} Y(\mathbf{f}), \quad (2.7)$$

siendo  $P_x(\mathbf{f})$  y  $P_w(\mathbf{f})$  la densidad espectral de potencia de la señal y el ruido, respectivamente.

### Eliminación de ruido y emborronamiento

En este caso la señal es degradada en primer lugar por un filtro, típicamente paso bajo, y, tras el mismo, se añade ruido a la señal emborronada. El modelo de observación en este caso es:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w}, \quad (2.8)$$

donde  $\mathbf{H}$  es una matriz circulante que realiza la convolución con el filtro lineal  $h$  (también conocido como respuesta al impulso o PSF, del inglés *Point Spread Function*). El objetivo, de nuevo, es obtener una estimación de la señal a partir de la observación y del conocimiento, por un lado, de los parámetros de degradación, y, por otro, de la distribución a priori de la señal. De forma análoga a la estimación MAP para el caso de eliminación de ruido, la estimación deseada tiene ahora la siguiente expresión:

$$\hat{\mathbf{x}}_{MAP} = \arg \min_{\mathbf{x}} \{C(\mathbf{x})\} = \arg \min_{\mathbf{x}} \{(\mathbf{y} - \mathbf{H}\mathbf{x})^T \mathbf{C}_w^{-1} (\mathbf{y} - \mathbf{H}\mathbf{x}) + \mathbf{x}^T \mathbf{C}_x^{-1} \mathbf{x}\}.$$

Derivando respecto a  $\mathbf{x}$  e igualando a cero, obtenemos:

$$\begin{aligned} \frac{d}{d\mathbf{x}} C(\mathbf{x}) &= \frac{d}{d\mathbf{x}} (\mathbf{x}^T (\mathbf{H}^T \mathbf{C}_w^{-1} \mathbf{H} + \mathbf{C}_x^{-1}) \mathbf{x} - 2\mathbf{y}^T \mathbf{C}_w^{-1} \mathbf{H}\mathbf{x} + \mathbf{y}^T \mathbf{C}_w^{-1} \mathbf{y}) \\ &= 2(\mathbf{H}^T \mathbf{C}_w^{-1} \mathbf{H} + \mathbf{C}_x^{-1}) \mathbf{x} - 2\mathbf{H}^T \mathbf{C}_w^{-1} \mathbf{y} = 0, \end{aligned} \quad (2.9)$$

de donde despejamos:

$$\hat{\mathbf{x}} = (\mathbf{H}^T \mathbf{C}_w^{-1} \mathbf{H} + \mathbf{C}_x^{-1})^{-1} \mathbf{H}^T \mathbf{C}_w^{-1} \mathbf{y}, \quad (2.10)$$

que operando sobre el óptimo resulta:

$$\hat{\mathbf{x}} = \mathbf{C}_x \mathbf{H}^T (\mathbf{H} \mathbf{C}_x \mathbf{H}^T + \mathbf{C}_w)^{-1} \mathbf{y}. \quad (2.11)$$

Esta solución representa la expresión del filtro de Wiener en forma vectorial para la eliminación de emborronamiento en presencia de ruido. Aplicando la transformada discreta de Fourier obtenemos la expresión del filtro en el dominio de la frecuencia:

$$\hat{X}(\mathbf{f}) = \frac{P_x(\mathbf{f}) H(\mathbf{f})^*}{|H(\mathbf{f})|^2 P_x(\mathbf{f}) + P_w(\mathbf{f})} Y(\mathbf{f}). \quad (2.12)$$

### 2.3.2. Clasificador Bayesiano Gaussiano

La teoría de decisión Bayesiana es un potente conjunto de herramientas que permite describir el problema de la clasificación en términos probabilísticos. Para introducir esta herramienta, nos apoyaremos en un ejemplo práctico: la identificación de cámaras. Imaginemos que disponemos de un conjunto  $\{c_1, c_2, \dots, c_K\}$  formado por  $K$  cámaras diferentes y deseamos determinar con cual de ellas se ha obtenido una determinada imagen. Para ello tenemos acceso, por ejemplo, al índice de ventas de cada cámara. Este índice puede identificarse con la probabilidad a priori de cada cámara,  $P(c_k)$ . En ausencia de otra información, la cámara seleccionada sería aquella con mayor índice de ventas. Supongamos ahora que podemos extraer de cada imagen un vector de características  $\mathbf{x}$ , perteneciente a un espacio  $\mathbb{R}^d$  (espacio de características). Idealmente, imágenes de cámaras diferentes proporcionarán vectores de características con valores sensiblemente distintos. Esta variabilidad puede ser expresada en términos probabilísticos, considerando que  $\mathbf{x}$  es un vector aleatorio de dimensión  $d$ , cuya distribución está en función de la cámara  $c_k$ . Esta distribución condicional,  $p(\mathbf{x}|c_k)$ , representa la verosimilitud, es decir, la probabilidad de la característica observada para una determinada cámara  $c_k$ . Dicha información puede ser combinada con la probabilidad a priori de cada cámara mediante la regla de Bayes para obtener la probabilidad a posteriori:

$$P(c_k|\mathbf{x}) = \frac{p(\mathbf{x}|c_k)P(c_k)}{p(\mathbf{x})}. \quad (2.13)$$

Es importante destacar en este caso que, a efectos de decisión sobre la clase  $c_k$ , la probabilidad  $p(\mathbf{x})$  es completamente irrelevante, puesto que solo representa una normalización. Para determinar la clase a la que pertenece la observación es necesario establecer una *regla de decisión*. El sentido común nos llevaría a elegir a la clase  $c_k$  cuya probabilidad a posteriori  $P(c_k|\mathbf{x})$  fuese mayor. Efectivamente, un análisis más detallado [25] demuestra que este criterio minimiza la probabilidad del error. El paradigma Bayesiano permite el uso de reglas de decisión más elaboradas, basadas en minimizar una determinada función de coste. Para una descripción más detallada, remitimos al lector a [25].

Basándonos en el marco descrito, el clasificador Bayesiano Gaussiano modela las densidades condicionales como distribuciones Gaussianas de dimensión  $d$ :

$$p(\mathbf{x}|c_k) = \frac{\exp(-(\mathbf{x} - \boldsymbol{\mu}_k)^T \mathbf{C}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)/2)}{((2\pi)^N |\mathbf{C}_k|)^{1/2}},$$

que quedan definidas mediante los  $K$  vectores de medias ( $\boldsymbol{\mu}_k$ ) y las  $K$  matrices de covarianza ( $\mathbf{C}_k$ ). En problemas de aprendizaje supervisado, en los cuales se cuenta con un conjunto de entrenamiento, el ajuste de estos parámetros puede realizarse siguiendo técnicas de estimación estándar, como máxima verosimilitud (ML, del inglés *Maximum Likelihood*).

## 2.4. Representación de las imágenes naturales

En esta Sección tratamos de motivar la conveniencia de utilizar representaciones diferentes de la original ("píxeles", o dominio de la imagen) para las imágenes naturales. Para ello, primero describiremos las herramientas de las que partimos, expondremos las limitaciones que presentan y finalmente propondremos posibles soluciones. Por simplicidad, las herramientas descritas inicialmente en esta Sección son de aplicación a señales unidimensionales, pero pueden ser extendidas para su uso con señales bidimensionales (imágenes, en nuestro caso). Finalizaremos la Sección motivando la idoneidad de las pirámides orientadas para el tratamiento de las imágenes y describiendo las representaciones que serán utilizadas en esta Tesis.

### 2.4.1. Las limitaciones de la transformada de Fourier

La transformada de Fourier es una de las herramientas más utilizadas en el procesamiento de señales espacial o temporalmente invariantes. Su extendido uso viene motivado por el carácter de los operadores lineales invariantes (convoluciones), que, como vimos en la Sección anterior, poseen como autovectores la base de Fourier [26].

La transformada de Fourier de una función  $f(t)$  se define como:

$$F(w) = \int f(t)e^{-j\omega t} dt. \quad (2.14)$$

Cada coeficiente de la transformada de Fourier (una frecuencia determinada,  $w$ ) se calcula como la correlación de la función  $f$  y la senoide  $e^{-j\omega t}$ . Cuanto más parecido sea  $f$  a esa senoide (léase, cuanto más componente de esa frecuencia posea  $f$ ) mayor será en magnitud el correspondiente coeficiente de Fourier.

La principal limitación de la transformada de Fourier es su carácter global. Cada coeficiente  $F(w)$  calculado en la expresión 2.14 depende de todo instante de tiempo  $t$ , es decir, es una mezcla de información de todos los instantes temporales. Esta mezcla hace que sea imposible analizar, a partir de los coeficientes de Fourier, qué está ocurriendo localmente en un determinado instante de tiempo  $t$ . El problema es extensivo a transformadas de Fourier de cualquier dimensión. Particularmente, en el caso de las imágenes, la transformada de Fourier no proporciona información local en una determinada posición espacial  $(n, m)$ .

### 2.4.2. Transformada de Fourier enventanada y sus limitaciones

Una posible alternativa al comportamiento global de la transformada de Fourier es aplicarla, con un soporte reducido, de forma local. Para ello dividimos la señal en segmentos iguales (multiplicando la señal por una ventana) y aplicamos separadamente la transformada de Fourier. Esta solución es conocida como la

transformada de Fourier enventanada (*Short Time Fourier Transform*, STFT), que posee la siguiente expresión:

$$STFT_{f(t)}^w(\tau, w) = \int [f(t)W(t - \tau)]e^{-jwt} dt, \quad (2.15)$$

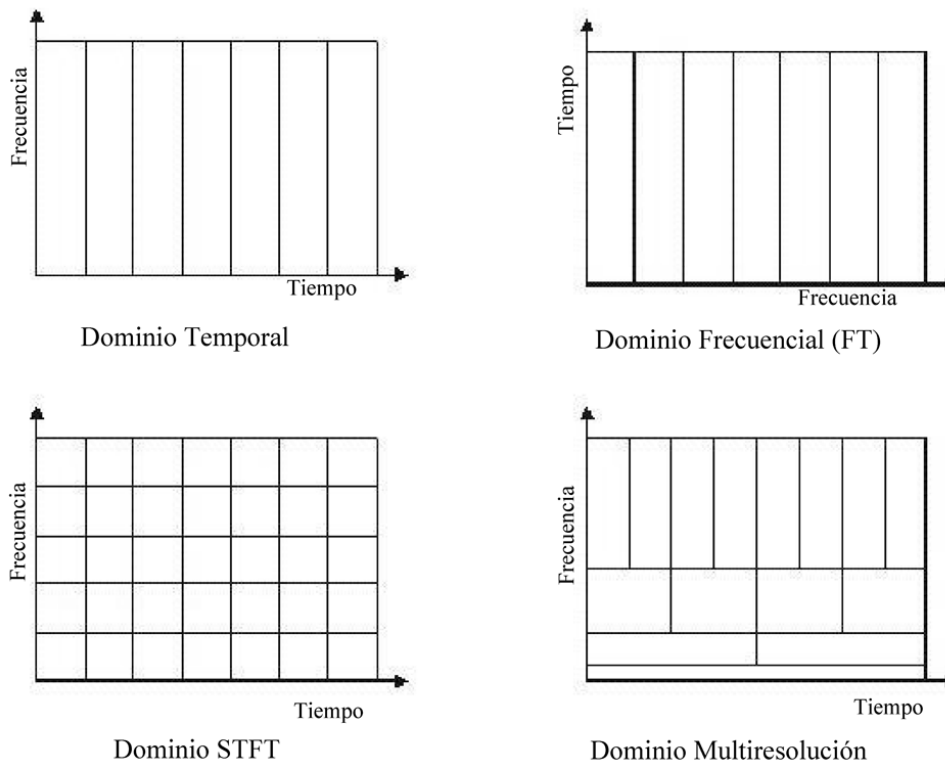
donde  $W(t - \tau)$  es la función ventana centrada en  $t = \tau$ .

El principio de incertidumbre de Heisenberg (1927), postula que es imposible conocer conjuntamente con exactitud la posición y la velocidad de una partícula en el caso de la mecánica cuántica. Cuanto mayor sea la precisión con que determinamos la posición, menor será la de su velocidad y viceversa. Exactamente lo mismo ocurre con la frecuencia y el tiempo (o el espacio), al ser ambas variables conjugadas, como en el caso de la posición y la velocidad de una partícula. No podemos calcular con precisión qué frecuencias ocurren en un determinado instante de tiempo, debemos conformarnos con saber qué bandas de frecuencias dominan en un cierto intervalo de tiempo. Vemos que el comportamiento de la transformada de Fourier enventanada depende críticamente de la elección de la ventana. Si deseamos una buena resolución en frecuencia, el tamaño de la ventana debe ser grande. Por contra, si queremos buena resolución temporal, la ventana ha de ser pequeña.

### 2.4.3. Análisis multi-escala

Cuando hablamos de escala nos referimos a un concepto equivalente al usado en cartografía. La escala es la relación entre las dimensiones del mapa y las dimensiones reales de la superficie que se representa. Un mapa a escala grande proporciona una vista general de una superficie extensa, mientras que un mapa a escala pequeña nos proporciona una vista particular de una superficie menor. Podemos pues establecer que la frecuencia es inversa a la escala utilizada, en tanto que escalas altas nos dan información sobre las frecuencias bajas y escalas bajas nos dan información sobre frecuencias altas.

La motivación del análisis multi-escala es encontrar un buen compromiso de resolución tiempo-frecuencia. Es decir, deseamos un mapa a escala variable, dependiendo de la información que queramos obtener. Para ello, el análisis multi-escala opta por dar una buena resolución temporal (y mala espectral) para las altas frecuencias y una buena resolución espectral (y mala temporal) para las bajas frecuencias. Obviamente esta elección no es casual, ya que en las señales que son de nuestro interés, las variaciones en los contenidos de alta frecuencia tienden a ocurrir en cortos periodos de tiempo, mientras que las bajas frecuencias suelen cambiar en periodos más largos de tiempo. En la Figura 2.1 se muestran los planos tiempo-frecuencia para los diferentes dominios que hemos descrito aquí. Como se puede observar, el dominio temporal no posee información acerca de las frecuencias, así como el dominio de Fourier carece de información temporal. La solución dada por la STFT divide al plano en regiones iguales, proporcionando



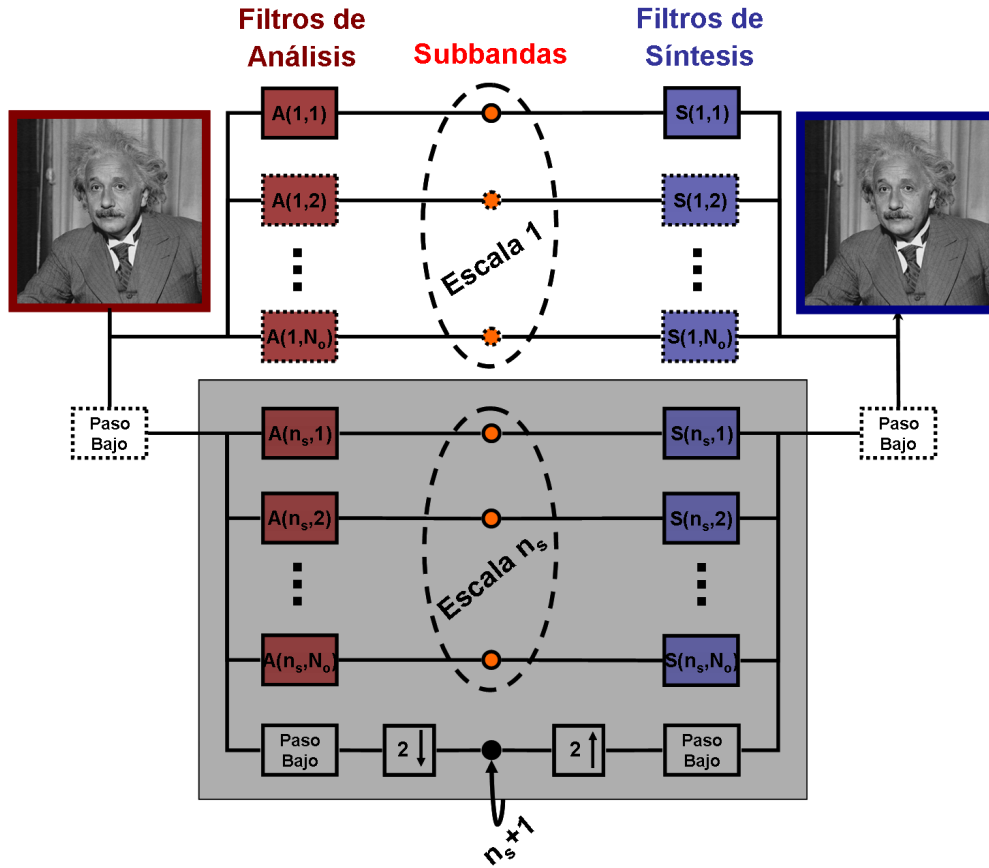
**Figura 2.1:** Planos tiempo-frecuencia para diferentes dominios.

la misma resolución con independencia de la escala. Por el contrario, la solución multi-escala divide al plano en regiones iguales en área, pero proporcionando distintas resoluciones a distintos niveles de tiempo y frecuencia. En definitiva, la diferencia entre el análisis multi-escala y la STFT está en el uso de una ventana variable para las distintas escalas. Para disponer de una transformación de este estilo es necesaria una función ventana ajustable, como la función ondícula (en inglés, *wavelet* [26]):

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi \left( \frac{t-b}{a} \right). \quad (2.16)$$

Esta función varía su aspecto y posición para distintos valores de  $a$  y  $b$ . El parámetro  $a$  controla la escala (dilata o comprime la ventana) y el parámetro  $b$  controla dónde se centra la ventana (traslación, centrada en  $t = b$ ). El término  $\frac{1}{\sqrt{a}}$  es un factor de normalización para asegurar que todas las ventanas poseen la misma energía. En la literatura, a esta función se le conoce como ondícula madre (en inglés, *mother wavelet*). Haciendo uso de estas funciones, la transformada ondicular continua (en inglés, *Continuous Wavelet Transform - CWT*) se expresa





**Figura 2.2.** Esquema genérico multi-resolución orientado: Análisis y Síntesis. La imagen de entrada es filtrada con las funciones base de análisis, obteniendo las subbandas asociadas a cada escala. Típicamente, se suele aplicar un esquema iterativo, que consiste en introducir el cuadro sombreado en el punto sólido del diagrama (que incluye filtrado paso bajo y submuestreo). La reconstrucción se obtiene de forma similar, pero aplicando los filtros de síntesis y sumando los resultados. Los componentes punteados son opcionales.

matemáticamente como sigue:

$$CWT_{f(t)}^{\Psi}(a, b) = \frac{1}{\sqrt{a}} \int f(t) \Psi \left( \frac{t-b}{a} \right) dt. \quad (2.17)$$

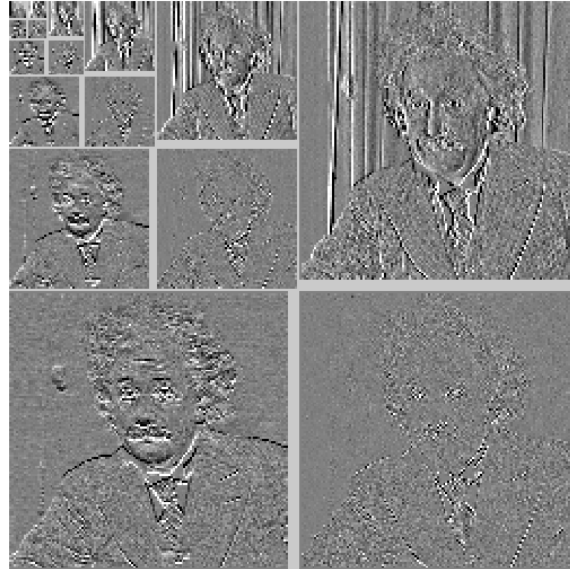
Existen muchas transformaciones multi-escala, tantas como funciones ondículas madre válidas. De las más citadas y antiguas es la ondícula de Haar [27].

#### 2.4.4. Análisis multi-resolución de imágenes digitales

Hasta ahora hemos descrito las transformaciones y las limitaciones asociadas a las mismas considerando señales unidimensionales. Como comentamos al

comenzar la Sección, las características de estas transformaciones son de aplicación a señales bidimensionales. La diferencia más notable respecto a las transformaciones unidimensionales es la posibilidad de incluir un nuevo grado de libertad en las ondículas madre: la orientación. Si bien las ondículas podrían no favorecer ninguna orientación (usando funciones isotrópicas), es bastante común el uso de ondículas con orientación dominante, que permiten la extracción de características en una determinada orientación. Además, las señales de nuestro interés, las imágenes digitales, son señales discretas. El análisis multi-resolución surge a partir de la discretización del análisis multi-escala. Esta discretización se realiza de forma diferente en las distintas escalas, lo que da lugar a las representaciones piramidales. En una pirámide multi-resolución, las escalas más bajas (base de la pirámide) poseen información de alta frecuencia, lo cual requiere un muestreo denso para ser representadas. A medida que se va subiendo de escala, se va aumentando el contenido de baja frecuencia y reduciendo el de alta, lo que permite (gracias al teorema de Nyquist) realizar un muestreo menos denso sin pérdida de información. En la Figura 2.2 se muestra un esquema multi-resolución genérico de la construcción y reconstrucción de una pirámide orientada (*Análisis-Síntesis*). Como podemos observar, la imagen es inicialmente filtrada por  $N_o$  funciones base orientadas, siendo  $N_o$  el número de orientaciones. Como resultado obtendríamos las subbandas de la primera escala. Las subbandas orientadas del resto de escalas se obtienen de forma iterativa, introduciendo el esquema sombreado en el punto sólido del diagrama. El proceso de reconstrucción tomaría las subbandas y procedería de forma inversa, aplicando los filtros de síntesis y sumando los resultados para obtener de nuevo la imagen. En la literatura existen gran variedad de descomposiciones lineales multi-resolución. Podemos hacer una clasificación atendiendo a la relación existente entre las diferentes funciones base generadas por la ondícula madre:

- Críticamente muestreadas: El espacio transformado posee la misma dimensión que el espacio original.
  - Ortogonales: Son aquellas cuyas funciones base son ortonormales, todas las funciones son ortogonales entre sí y además poseen norma unidad. Como ejemplo, en la Figura 2.3 podemos observar la descomposición de la imagen de *Einstein* usando la ondícula de Haar [27] aplicada de forma separable a las filas y columnas de la imagen.
  - Biortogonales: Son aquellas cuyas funciones de análisis son ortogonales a las de síntesis. Sin embargo las funciones de análisis, al igual que las de síntesis, no son ortogonales entre sí.
- Sobre-completas: Representaciones cuyas funciones base son redundantes, es decir, el número de coeficientes de la representación es mayor que el número de píxeles de la imagen. En la Figura 2.4 se muestra un ejemplo



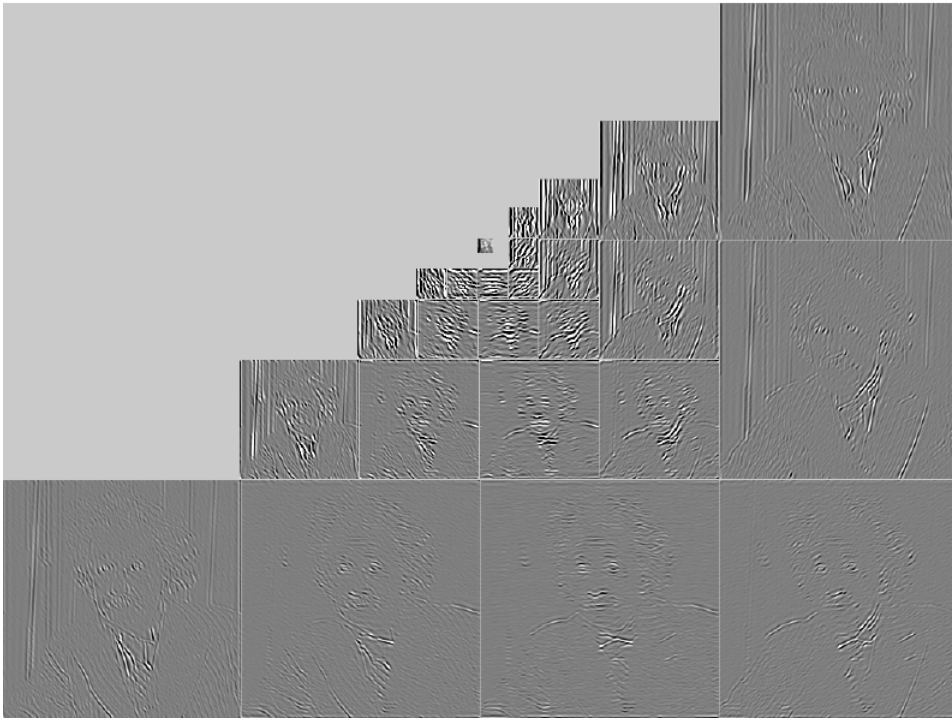
**Figura 2.3.** Transformación wavelet (Haar) de la imagen de *Einstein*  $256 \times 256$ .

de descomposición piramidal sobre-completa de la imagen de *Einstein* en 6 orientaciones y 4 escalas usando la *Steerable pyramid* [6].

La elección de una familia de representaciones u otras depende principalmente del objeto de la aplicación. Para restauración de imágenes en general, está ampliamente aceptado que las representaciones sobre-completas son más potentes que las de muestreo crítico. Los inconvenientes asociados al incremento de dimensión (aumento de procesamiento, principalmente) son claramente compensados por las ventajas que poseen:

- No se introducen artefactos por sub-muestreo.
- Pueden diseñarse de forma selectiva en orientación (no se mezclan orientaciones, como ocurre con las representaciones ortogonales y biortogonales).
- Más similar al procesamiento en el sistema visual humano temprano. Los filtros utilizados son cualitativamente parecidos a los campos receptivos de las neuronas simples en el área V1 de la corteza visual (filtrado paso banda, selectivo en orientación, localizado espacialmente)<sup>3</sup>.

<sup>3</sup>Esta propiedad, aunque no se puede calificar como una ventaja directamente, parece deseable desde el punto de vista de que el sistema visual humano ha optimizado sus recursos durante millones de años de evolución para permitir un procesamiento eficiente de la información visual.



**Figura 2.4.** Transformación en 6 orientaciones y 4 escalas de la imagen de *Einstein*  $256 \times 256$ , usando *Steerable pyramid*.

#### 2.4.5. Idoneidad del análisis multi-resolución en el procesamiento de imágenes

Una vez que hemos analizado las propiedades de una descomposición multi-resolución, podemos entender mejor la razón de usarla en el procesamiento de imágenes. Desde el punto de vista estadístico, el uso de estas representaciones está justificado por la excelente descripción de las características estadísticas de las imágenes naturales. La capacidad de proporcionar análisis a distintas resoluciones permite representar de forma eficiente objetos de diferente tamaño. Adicionalmente, las imágenes presentan patrones locales orientados tales como bordes, barras, etc, por lo que una representación selectiva en orientación es fundamental.

Además del punto de vista estadístico, podemos justificar el uso de estas representaciones desde el punto de vista de la visión. Los sistemas visuales biológicos han evolucionado hacia un óptimo aprovechamiento de sus recursos, tanto de almacenamiento como de procesamiento, de forma que la representación visual biológica está adaptada a las características estadísticas de los estímulos de entrada (imágenes naturales) [2, 4]. Esto implica que los sistemas artificiales que maximicen la cantidad de información relevante representable por un número

dado de muestras, tendrán coincidencias básicas con los sistemas de representación biológicos.

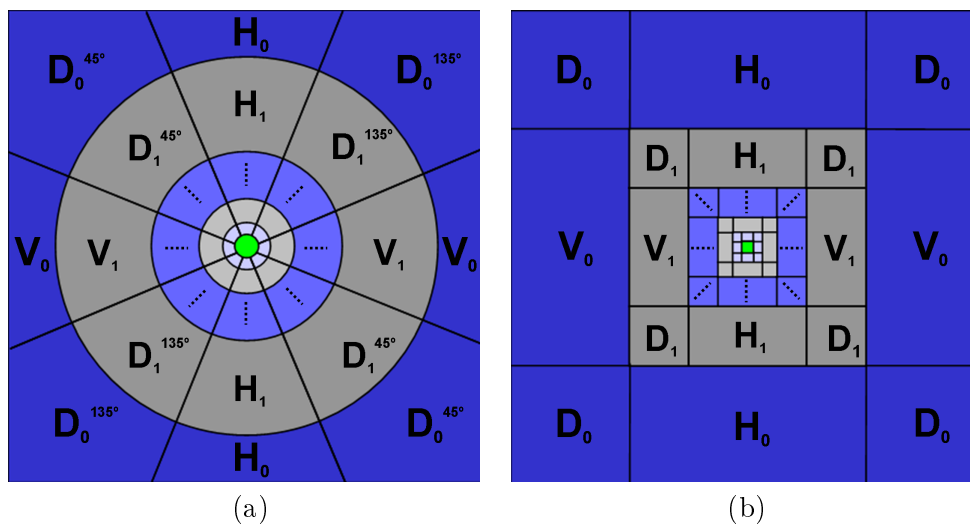
Como comentamos en la Sección anterior, las pirámides orientables sobre-completas, (*Steerable pyramid* [6], *curvelets* [28] o wavelets complejas [29], entre otras muchas) han demostrado ser mucho más potentes para aplicaciones de restauración que las críticamente muestreadas. Sin embargo, no existe respuesta para la pregunta de qué representación es la que proporciona mejores características, ya que en general, éstas dependen del contenido de las imágenes. Las representaciones que usaremos en esta Tesis se describen a continuación.

### Full Steerable Pyramid (FSP)

La FSP [10] es una extensión de la pirámide orientable de Simoncelli y Freeman [6], que, a diferencia de esta última, incluye selectividad en orientación en el residuo paso alto. Las principales ventajas de la FSP son: (a) sus filtros pueden ser muy selectivos en frecuencia, y (b) es flexible en el número de orientaciones. Los inconvenientes están en relación con estas ventajas: (a) debido a la alta selectividad de los filtros en frecuencia, éstos poseen un gran soporte espacial, y (b) debido al gran número de subbandas, su uso implica un alto coste computacional (por ejemplo, una descomposición con 8 orientaciones deriva en un factor de redundancia cercano a 19, es decir, un ratio de 19 coeficientes por cada píxel de la imagen). Esta representación es especialmente adecuada para representar imágenes ricas en texturas. En la parte izquierda de la Figura 2.5 podemos observar un ejemplo de descomposición en frecuencia con esta representación. Para una descripción más detallada, remitimos al lector al apéndice de [10].

### Translation Invariant Haar Pyramid (TIHP)

La TIHP es una versión sobre-completa de la ondícula de Haar [27] bidimensional. Se trata de una representación invariante a traslación y con reconstrucción casi perfecta. De forma complementaria a FSP, esta representación posee una excelente localización espacial, gracias al reducido tamaño de sus filtros. La pirámide tiene limitada el número de orientaciones (3: vertical, horizontal y las dos diagonales mezcladas), lo que proporciona un bajo factor de redundancia (7). Los principales inconvenientes son la baja localización espectral y el bajo rendimiento en el procesado de orientaciones oblicuas. TIHP es pues adecuada para procesar imágenes con zonas homogéneas con bordes muy definidos, sobre todo en la dirección vertical y horizontal. En la parte derecha de la Figura 2.5 se muestra en detalle la descomposición en frecuencia de esta pirámide. Remitimos al lector al apéndice de [16] para una lectura más detallada acerca de esta representación.



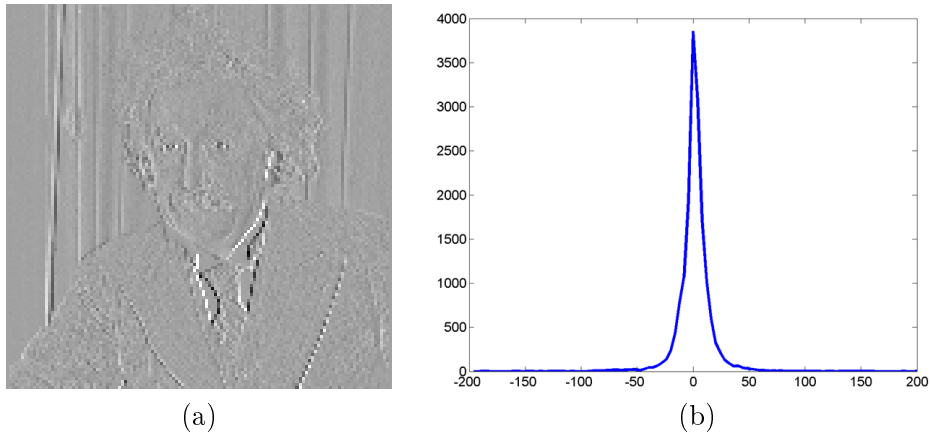
**Figura 2.5.** División frecuencial de las representaciones utilizadas en esta Tesis. (a) *Full Steerable Pyramid* (FSP). (b) *Translation Invariant Haar Pyramid* (TIHP). Las regiones marcadas con una misma etiqueta representan el contenido frecuencial de una determinada subbanda.

## 2.5. Introducción a las mezclas en escala de Gaussianas (GSM)

En el ámbito del procesamiento de imágenes es de capital importancia poseer un modelo estadístico de las mismas. Para la construcción de estos modelos es fundamental un riguroso estudio de las propiedades estadísticas de las imágenes. En esta Sección describiremos de forma detallada las propiedades de las imágenes en el dominio ondicular y presentaremos a GSM como una herramienta que modela, de forma natural, el comportamiento estadístico de vecindades o agrupaciones de coeficientes wavelet.

### 2.5.1. Estadística de las imágenes en el dominio ondicular

La estadística de las imágenes fue descrita en un principio en base a sus propiedades espectrales. Mediante estudios empíricos [30, 31] se demostró que el comportamiento típico de la densidad espectral de potencia seguía una distribución  $f^{\nu-2}$ , siendo  $\nu$  un pequeño factor de corrección. De esta forma, la tendencia más común consistía en modelar las imágenes como muestras de campos aleatorios Gaussianos fuertemente correlacionados. Field [4] en 1987 y dos años más tarde Mallat [32], demostraron con datos empíricos que las respuestas de las imágenes naturales a filtrados paso-banda (subbandas) estaban lejos de ser Gaussianas. Los coeficientes de representaciones ondulares presentaban una función densidad de probabilidad marginal con muy alta kurtosis: un pico



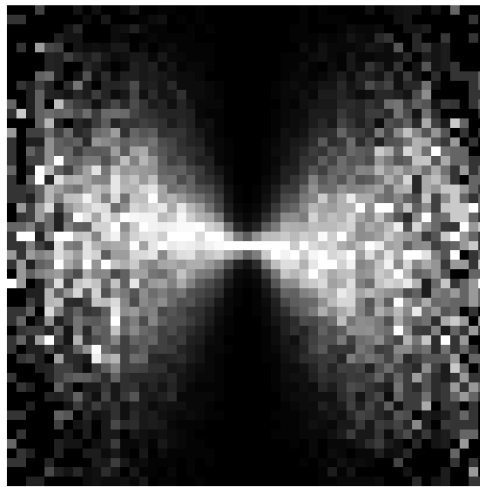
**Figura 2.6.** Subbanda vertical de alta frecuencia de la imagen de *Einstein* y su histograma. En (a) se observa que los valores bajos (en amplitud) son muy numerosos en comparación con los altos. En (b) se muestra el histograma, con la mayoría de los coeficientes concentrados en valores de baja amplitud, resultando un histograma muy no Gaussiano.

pronunciado en cero y colas que caen rápidamente. Así pues, una pequeña proporción de coeficientes concentraban la mayoría de la energía. Como podemos observar en la Figura 2.6, esto no solo ocurría con los histogramas de todos los coeficientes de la representación, también sucedía en los coeficientes de las subbandas por separado. Podemos encontrar la razón de este comportamiento en la alta localización de bordes (alto contraste) frente al dominio en área de texturas de bajo contraste. En definitiva, la heterogeneidad espacial de contenidos de la imagen es la fuente principal que aleja a la estadística de las imágenes de distribuciones Gaussianas, ya que provoca cambios abruptos y respuestas altas en amplitud en los bordes de los objetos.

En 1997, Simoncelli [33] mostró una importante observación acerca de la estadística en el dominio ondicular: Existe un fuerte acoplamiento en amplitud entre coeficientes vecinos, tanto espaciales como vecinos en escala o en orientación. Este descubrimiento revela explícitamente un comportamiento conjunto no Gaussiano, que es coherente con las características de las densidades marginales comentadas previamente. En la Figura 2.7 podemos observar los coeficientes (en valor absoluto) de la transformación wavelet de Haar. Como vemos claramente, existe una correlación significativa a nivel de cada subbanda (vecinos espaciales) al igual que en escalas y orientaciones contiguas. La Figura 2.8 muestra la función densidad de probabilidad de la amplitud de un coeficiente dada la amplitud del coeficiente vecino. Teniendo en cuenta que negro representa probabilidad 0 y blanco alta probabilidad, es claro que si el coeficiente de referencia posee baja amplitud, hay una muy alta probabilidad de que el vecino posea baja amplitud.



**Figura 2.7.** Coeficientes en amplitud de la transformación wavelet (Haar) de la imagen de *Einstein*  $256 \times 256$ .



**Figura 2.8.** Función densidad de probabilidad condicional empírica de un coeficiente dado un vecino, en escala, orientación o espacio. La incertidumbre acerca de la amplitud del coeficiente vecino aumenta a medida que el coeficiente de referencia tiene mayor amplitud. Blanco indica alta probabilidad y negro probabilidad 0. Las columnas han sido normalizadas independientemente en amplitud (en lugar de en su integral) para facilitar la visualización. Figura tomada de [10].

En cambio si la referencia posee un valor alto, existe mucha incertidumbre acerca del valor que poseerá el vecino.



El descubrimiento de estas características fundamentales ha sido explotado en áreas como la compresión de imágenes [34, 35], el modelado y síntesis de texturas [36] y la restauración de imágenes, lo que deja claro la importancia de que los modelos sean capaces de reproducir este comportamiento.

### 2.5.2. Modelado de las propiedades estadísticas no Gaussianas

En 1999, Wainwright y Simoncelli [9] descubrieron que las mezclas en escala de Gaussianas [37] (GSM, del inglés *Gaussian Scale Mixtures*) proporcionaban una vía adecuada para modelar y reproducir las dos propiedades estadísticas presentadas en el apartado anterior: la alta kurtosis y el acoplamiento en amplitud de los coeficientes wavelet vecinos en el espacio, escala u orientación.

Un vector aleatorio  $\mathbf{x}$  es una mezcla en escala de Gaussianas [37] si y sólo si lo podemos expresar como el producto de dos variables aleatorias independientes:

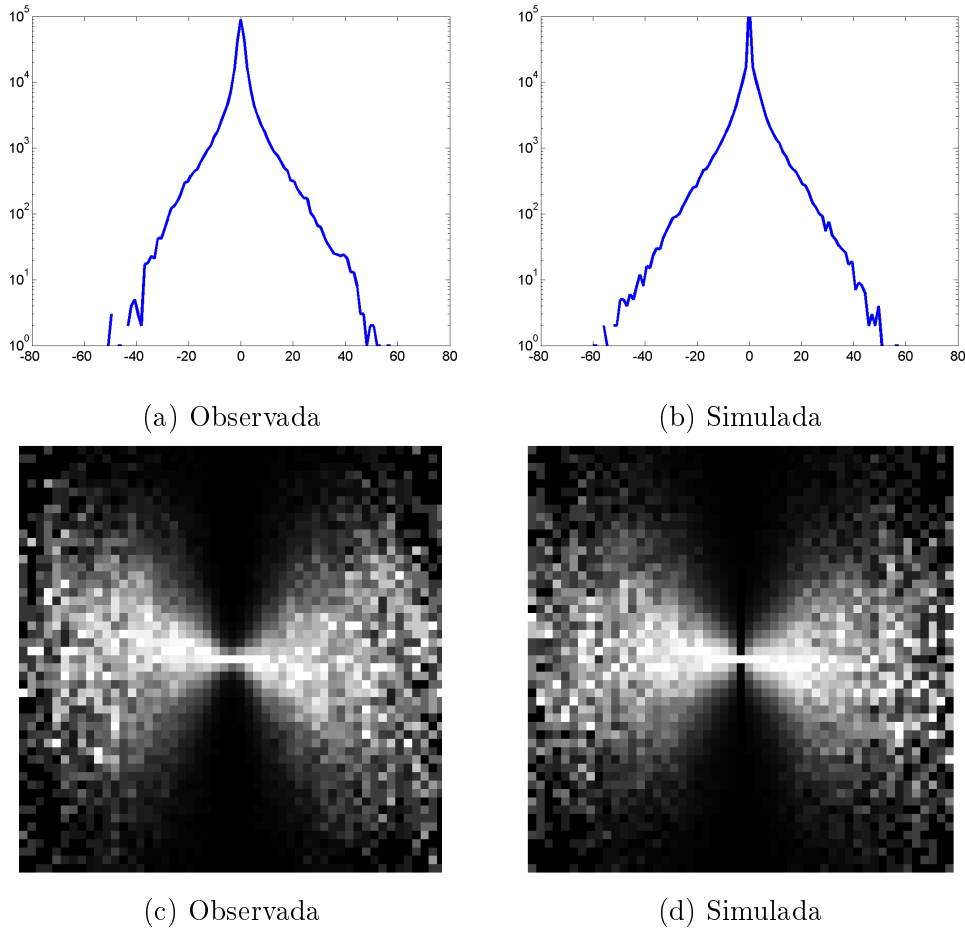
$$\mathbf{x} = \sqrt{z}\mathbf{u}, \quad (2.18)$$

siendo  $\mathbf{u}$ , un vector Gaussiano de media cero y  $\sqrt{z}$  un escalar positivo. A esta última variable se le conoce como multiplicador oculto y su rol se asocia con la modulación de la varianza local de los coeficientes. De esta forma, el vector  $\mathbf{x}$  es una mezcla infinita de vectores Gaussianos, cuya densidad viene determinada por la matriz de covarianza  $\mathbf{C}_{\mathbf{u}}$ , asociada al vector  $\mathbf{u}$ , y por la densidad del multiplicador,  $p(z)$ :

$$\begin{aligned} p(\mathbf{x}) &= \int p(\mathbf{x}|z)p(z) dz \\ &= \int \frac{\exp(-\mathbf{x}^T(z\mathbf{C}_{\mathbf{u}})^{-1}\mathbf{x}/2)}{(2\pi)^{N/2}|z\mathbf{C}_{\mathbf{u}}|^{1/2}} p(z) dz, \end{aligned} \quad (2.19)$$

donde  $N$  es la dimensión de  $\mathbf{x}$  y  $\mathbf{u}$ , que en nuestro caso representa el número de coeficientes de la vecindad considerada. Como avanzábamos, las características de las GSM son muy adecuadas para modelar las propiedades no Gaussianas de los coeficientes wavelet. Son distribuciones simétricas centradas en cero con densidades marginales leptokurtóticas (kurtosis  $\leq 3$ ). Pero sin duda, una de las características más ventajosas de los modelos GSM es que la densidad de  $\mathbf{x}$  condicionada a  $z$  (fijado un  $z$ ) es Gaussiana. Incluso el vector normalizado  $\mathbf{x}/\sqrt{z}$  es Gaussiano. Esta importante propiedad permite que los modelos sigan siendo Gaussianos de forma condicional, con la flexibilidad que ello lleva asociado.

Como muestra la Figura 2.9, se consigue una simulación bastante fiel de las propiedades estadísticas de la imagen transformada, tanto de las densidades marginales como de las condicionadas.



**Figura 2.9.** Coeficientes estadísticos de una subbanda vertical de la imagen *Boats* y de una simulación usando GSM. (a) y (b): Histogramas marginales en escala logarítmica. (c) y (d): Histogramas condicionales de dos coeficientes vecinos (espaciales). Blanco representa alta probabilidad, negro probabilidad 0. Figura tomada de [10].

## 2.6. Restauración de imágenes usando GSM

En esta Sección describiremos en detalle BLS-GSM [10] (del inglés, *Bayes Least Squares-Gaussian Scale Mixtures*), un método de eliminación de ruido en imágenes que ha tenido gran difusión en los últimos años, y que constituye el punto de partida de las aplicaciones desarrolladas en esta Tesis. Además, hablaremos sobre la posibilidad de utilizar BLS-GSM para la restauración de imágenes afectadas por ruido y emborronamiento. La comprensión de este modelo permitirá al lector entender la motivación de varias de las aportaciones que se presentan en este trabajo.

### 2.6.1. Eliminación de ruido

El método de eliminación de ruido BLS-GSM, al igual que la mayoría de aproximaciones multi-escala/multi-orientación, realiza la restauración en tres pasos: (a) descompone la imagen en subbandas utilizando una representación piramidal, (b) procesa todas las subbandas excepto el residuo paso bajo, y (c) recompone la pirámide para obtener la estimación.

#### Modelo

El modelo de imagen GSM original [10] es un modelo local que permite caracterizar las fluctuaciones de la varianza local dentro de las subbandas de representaciones piramidales. Para ello, se definen vecindades de tamaño  $L \times L$  (típicamente  $3 \times 3$  [10]) que son modeladas, previa vectorización, como vectores GSM cuya distribución viene dada por la Ecuación 2.19. La elección de la densidad del multiplicador,  $p(z)$ , es importante y repercute de forma crucial en el rendimiento del modelo. Si bien en [10] se presentan varias posibilidades, como distribuciones Gaussianas generalizadas o estimaciones de máxima verosimilitud (ML, del inglés *Maximum Likelihood*), se opta por el uso de una distribución uniforme en el logaritmo, que, en este caso, se corresponde con el prior de Jeffrey o prior no informativo.

Aplicando la definición de GSM (ver Ecuación 2.18), el modelo de observación,  $\mathbf{y}$ , es:

$$\mathbf{y} = \mathbf{x} + \mathbf{w} = \sqrt{z}\mathbf{u} + \mathbf{w}, \quad (2.20)$$

donde  $\mathbf{u}$  y  $\mathbf{w}$  son vectores Gaussianos de dimensión  $N$  ( $N = L^2$ ) de media cero, que representan respectivamente a la señal normalizada y al ruido. Se asume la independencia de ambos vectores y el escalar  $z$ . La matriz de covarianza de la observación, fijado un  $z$ , puede expresarse entonces en función de las covarianzas de  $\mathbf{u}$  y  $\mathbf{w}$  como  $\mathbf{C}_{\mathbf{y}|z} = z\mathbf{C}_{\mathbf{u}} + \mathbf{C}_{\mathbf{w}}$ , lo que permite expresar la densidad de la vecindad observada condicionada al escalar  $z$  como sigue:

$$p(\mathbf{y}|z) = \frac{\exp(-\mathbf{y}^T(z\mathbf{C}_{\mathbf{u}} + \mathbf{C}_{\mathbf{w}})^{-1}\mathbf{y}/2)}{\sqrt{(2\pi)^N |z\mathbf{C}_{\mathbf{u}} + \mathbf{C}_{\mathbf{w}}|}}. \quad (2.21)$$

Gracias a la independencia de  $z$ ,  $\mathbf{u}$  y  $\mathbf{w}$ , y tomando esperanzas sobre  $z$ , se obtiene:

$$\mathbf{C}_{\mathbf{y}} = \mathbb{E}\{\mathbf{C}_{\mathbf{y}|z}\} = \mathbb{E}\{z\}\mathbf{C}_{\mathbf{u}} + \mathbf{C}_{\mathbf{w}}, \quad (2.22)$$

que al forzar, sin pérdida de generalidad,  $\mathbb{E}\{z\} = 1$ , se convierte finalmente en:

$$\mathbf{C}_{\mathbf{y}} = \mathbf{C}_{\mathbf{u}} + \mathbf{C}_{\mathbf{w}}. \quad (2.23)$$

Una vez elegida la densidad del multiplicador oculto, los parámetros a estimar (para cada subbanda) son las covarianzas de la señal,  $\mathbf{C}_u$ , y ruido,  $\mathbf{C}_w$ . Para calcular esta última, primero se aplica la misma descomposición piramidal a una señal determinista<sup>4</sup> con la misma autocorrelación (conocida) que el ruido. Tras ello, se obtienen muestralmente las covarianzas asociadas a cada una de las subbandas obtenidas. Para calcular la covarianza de la señal,  $\mathbf{C}_u$ , se resta a la covarianza de la observación  $\mathbf{C}_y$ , estimada de forma muestral a partir de las vecindades observadas en cada subbanda, la covarianza del ruido, con la precaución de asegurar autovalores no negativos. En este método, tanto la covarianza del ruido como la covarianza de la señal,  $\mathbf{C}_u$ , se asumen constantes para todas las vecindades GSM dentro de una misma subbanda, por lo que solo es necesario estimarlas una vez por subbanda.

### Estimación de coeficientes

Para cada vecindad  $\mathbf{y}$  de coeficientes observados, se estima  $x_c$ , el coeficiente de referencia situado en el centro de la vecindad. Usando mínimos cuadrados de Bayes (BLS), la estimación queda como sigue:

$$\begin{aligned}
 \mathbb{E}\{x_c|\mathbf{y}\} &= \int x_c p(x_c|\mathbf{y}) dx_c \\
 &= \int \int_0^\infty x_c p(x_c, z|\mathbf{y}) dz dx_c \\
 &= \int \int_0^\infty x_c p(x_c|\mathbf{y}, z) p(z|\mathbf{y}) dz dx_c \\
 &= \int_0^\infty p(z|\mathbf{y}) \mathbb{E}\{x_c|\mathbf{y}, z\} dz.
 \end{aligned} \tag{2.24}$$

La solución se escribe como una media de estimaciones BLS del coeficiente central condicionado por un  $z$  y ponderado por la densidad a posteriori de ese  $z$  dado la vecindad observada,  $p(z|\mathbf{y})$ . Una gran ventaja de este modelo es que el vector  $\mathbf{x}$  es Gaussiano dado un  $z$ . Gracias a ello, y a que el ruido también es Gaussiano, el valor esperado dentro de la integral (2.24) es la estimación de Wiener:

$$\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\} = z\mathbf{C}_u(z\mathbf{C}_u + \mathbf{C}_w)^{-1}\mathbf{y}. \tag{2.25}$$

Para aumentar la eficiencia del método, se realiza una doble diagonalización que simplifica la dependencia con  $z$ , lo que permite una importante reducción del coste computacional. Referimos al lector a [10] para una descripción más detallada de los detalles de implementación.

<sup>4</sup>Al usar una señal determinista se eliminan fluctuaciones aleatorias.

### 2.6.2. Eliminación de ruido y emborronamiento

El modelo GSM es muy flexible y permite considerar que, además de ruido, existe una distorsión lineal (filtrado) en el proceso de degradación:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w} = \sqrt{z}\mathbf{H}\mathbf{u} + \mathbf{w}. \quad (2.26)$$

La propiedad fundamental a conservar es que la observación local pueda seguir siendo descrita como Gaussiana cuando se condiciona en el multiplicador oculto  $z$ . Afortunadamente, el filtrado lineal no destruye esta propiedad, y el método de restauración [19] se describe de forma análoga a la solución BLS-GSM. Teniendo en cuenta ahora que:

$$\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\} = z\mathbf{C}_u\mathbf{H}^T(z\mathbf{H}\mathbf{C}_u\mathbf{H}^T + \mathbf{C}_w)^{-1}\mathbf{y},$$

siendo la obtención de la estimación BLS análoga a la que mostramos en la Ecuación 2.24 (ver [19] para más detalles).

## Capítulo 3

# Modelado con mezclas en escala de Gaussianas espacialmente adaptativas

### 3.1. Introducción

El modelado estadístico es un tema ampliamente tratado en el área del procesamiento de imágenes. Esto se debe, principalmente, a la importante repercusión que tienen estos modelos en el rendimiento de las aplicaciones finales. BLS-GSM [10], descrito en el Capítulo anterior, es un claro ejemplo en este sentido: el potente (y flexible) modelado estadístico de la imagen basado en GSM permitió el desarrollo de uno de los métodos de eliminación de ruido más exitosos de los últimos años. En este Capítulo presentamos una mejora del modelo GSM original y describimos un nuevo método de eliminación de ruido que proporciona una mejora generalizada respecto a dicha referencia.

La solución clásica del filtrado de Wiener, descrita en la Sección 2.3, es uno de los métodos de restauración más simples que existen en la literatura. A pesar de ello, proporciona un resultado óptimo (en sentido mínimo cuadrático) cuando la señal y el ruido son Gaussianos. Si bien el modelado del ruido como Gaussiano representa una aproximación razonable en muchos casos, no ocurre lo mismo con el modelado de la señal: en la Sección 2.5.2 pudimos comprobar que la estadística de las imágenes es totalmente diferente de una distribución Gaussiana [4, 32]. Para solventar este problema, muchos autores han optado por caracterizar a las imágenes como *localmente* Gaussianas. En 1980, Lee [38] propuso una versión local del filtrado de Wiener escalar. El método, que se aplicaba en el dominio de la imagen, estaba basado en la estimación de la media y la varianza local en distintas posiciones espaciales de la imagen. Con la aparición de las representaciones piramidales, se generalizó el uso de modelos que representaban el comportamiento de las imágenes en el dominio transformado, aumentando así la

### **34 Modelado con mezclas en escala de Gaussianas espacialmente adaptativas**

potencia de los modelos espacialmente adaptativos. Mientras que algunos autores exploraron aproximaciones que dieron lugar a diferentes funciones de umbralizado de los coeficientes (por ejemplo, [39, 40, 41, 42]), otros siguieron con el desarrollo de Wiener local en el dominio ondicular. Mihçak *et al.* [43], continuando con la idea de Lee, propuso un prior de la varianza local de los coeficientes que estimó siguiendo un esquema máximo a posteriori (MAP). Una aproximación Bayesiana similar fue propuesta en [44]. En 2001, Portilla *et al.* [45], presentaron una mejora sustancial del modelo Bayesiano, incluyendo en el mismo la matriz de covarianza de las vecindades de la señal (Wiener vectorial) y simplificando la formulación con el uso de mezclas en escala de Gaussianas [9]. Todos estos métodos adaptativos basados en Wiener, seguían una aproximación de Bayes empírico: (a) estimar los parámetros (varianza o covarianza, según el caso), y (b) aplicar el filtrado de Wiener usando la estimación. Dos años después, Portilla *et al.* [10] presentaron el método BLS-GSM. Este método proporcionaba una mejora significativa de su trabajo anterior, al eliminar la separación de la estimación previa de los parámetros, integrando toda la estimación en un solo paso óptimo en un sentido de mínimos cuadrados BLS (del inglés, *Bayes Least Squares*). Hasta ese momento, los métodos de Wiener adaptativos o bien tan solo se adaptaban a la varianza local (Wiener escalar) sin tener en cuenta la estructura de la covarianza en las subbandas [43, 44], o bien consideraban que dicha covarianza posee una estructura fija que solo varía de acuerdo a un factor de escala [45, 10], al cual se adaptaban localmente. En ambos casos podemos pensar que se trata de una adaptación *fin*a, en el sentido de que se estima la varianza local usando pequeñas vecindades. Por contra, existe una amplia literatura acerca de la adaptación local en covarianza y sus aplicaciones (por ejemplo, [46]). La estimación de la covarianza permite una caracterización selectiva de las propiedades espectrales locales. Sin embargo, al necesitar una vecindad mucho mayor para su estimación, no es capaz de capturar las variaciones rápidas en regiones pequeñas. Como consecuencia, para las imágenes naturales, la complejidad de incluir un modelo adaptativo a la covarianza no compensa en términos de rendimiento si se compara con una estimación simple de la varianza local. En este Capítulo presentamos una nueva aproximación que proporciona ambos tipos de adaptación en un mismo modelo [11, 16]. Un trabajo relacionado fue posteriormente propuesto por Hammond y Simoncelli [47], en el cual se describía otra interesante mejora de BLS-GSM basada en la adaptación local a la orientación dominante (OAGSM, del inglés, *Oriented Adaptive Gaussian Scale Mixtures*).

Como venimos comentando, BLS-GSM [10] ha marcado el estado del arte en eliminación de ruido Gaussiano en los últimos años. Sin embargo, recientemente han aparecido una serie de métodos con resultados impresionantes, algunos de ellos superando claramente a BLS-GSM, tanto visualmente como en términos de norma  $L2$ . La mayoría de estos métodos se basan en explotar una propiedad importante de las imágenes: la existencia de una gran redundancia espacial, en el

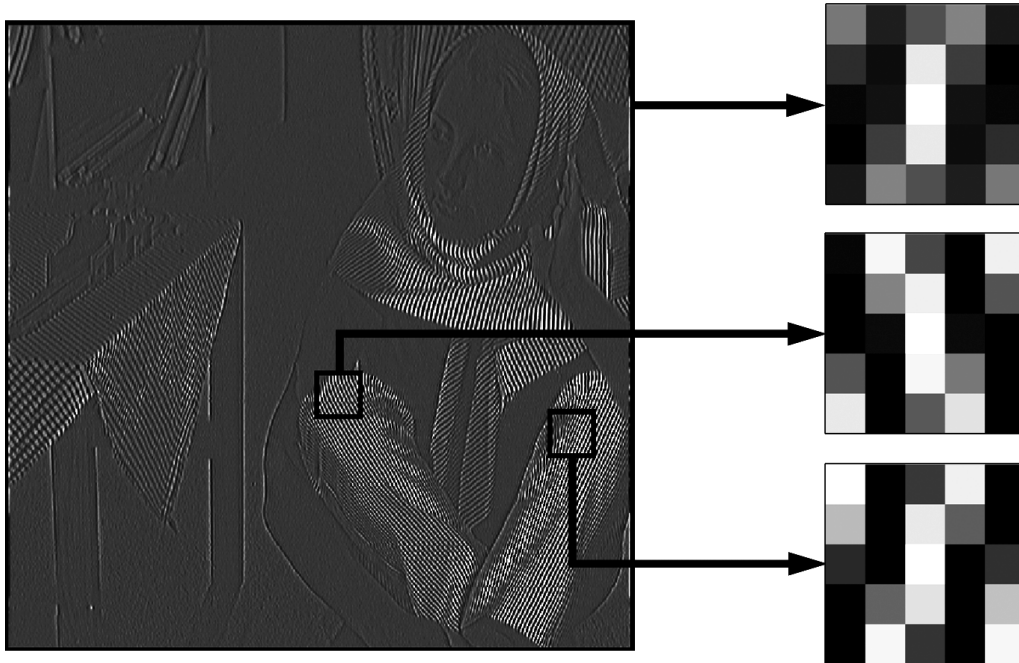
sentido de que una característica local está muy probablemente presente en varias regiones de la imagen (posiblemente no adyacentes). Podemos pensar, por ejemplo, en la fachada de un edificio, con sus ladrillos repetidos, en las esquinas de una ventana o simplemente en porciones de una línea orientada. Esta propiedad puede ser explotada para la eliminación de ruido, asumiendo (razonablemente) que la mayor parte de las diferencias entre estas regiones se deben a la presencia de ruido. La aproximación común de estos nuevos métodos consiste en considerar que cada vecindad de píxeles puede ser expresada como una suma ponderada (combinación lineal) de otras vecindades. Buades *et al.* [48] presentaron un método, "*medias no locales*", que calculaba los pesos a partir de la similitud contextual de las vecindades [49]. Kervrann y Boulanger [50] mejoraron este método introduciendo una vecindad adaptativa junto con un modelo estadístico de estimación de los parámetros. Hirakawa y Parks [51] también hicieron uso de esta propiedad y propusieron un algoritmo basado en mínimos cuadrados totales (TLS, del inglés *Total Least Squares*). Por otro lado, Dabov *et al.* [52] agruparon vecindades en estructuras 3D utilizando medidas de similitud  $L2$  (correlación). Estas estructuras eran entonces procesadas mediante una combinación de dos métodos: umbralizado y filtrado de Wiener adaptativo. A nuestro leal entender, estos autores poseen el estado del arte actual en términos de rendimiento. Sin embargo, el método no está basado en un modelo de estimación simple, al aplicar re-estimaciones. Hablamos de re-estimación al hecho de utilizar el resultado de una estimación como parámetro de otra estimación diferente, compartiendo ambos el mismo objetivo<sup>1</sup>. Aunque posiblemente sea una técnica efectiva y conveniente en algunas situaciones prácticas, la re-estimación no es una técnica conceptualmente limpia, ya que implica el solapamiento de dos modelos mutuamente incoherentes. Otra tendencia de éxito en el campo de la eliminación de ruido en imágenes se basa en la adaptación a la geometría de la imagen. Foi *et al.* [53] presentaron un método muy potente basado en una transformada coseno discreta adaptativa a la forma (SA-DCT, del inglés *Shape Adaptive Discrete Fourier Transform*), que consiste en adaptar el soporte local de los datos a procesar en función del contenido de la imagen. Al igual que en [52] también hacen uso de re-estimaciones. Elad *et al.* [54] también han obtenido resultados impresionantes mediante el uso de una representación adaptativa que se basa en el uso del algoritmo KSVD (del inglés, *K-Singular Value Decomposition*) para obtener un diccionario que describe de forma efectiva el contenido de la imagen. Gran parte del buen rendimiento de este método se debe al uso de una representación no lineal adaptativa, que requiere un cómputo intensivo.

El método que describimos en este Capítulo, a diferencia de los descritos anteriormente, puede ser utilizado en cualquier descomposición multi-resolución.

---

<sup>1</sup>Cuando ambos métodos son el mismo, se habla de estimación iterativa, que es una práctica conceptualmente limpia, como es el caso del algoritmo EM (del inglés, *Expectation-Maximization*).





**Figura 3.1.** Variabilidad espacial de la autocovarianza dentro de una subbanda vertical (TIHP) de *Barbara*. Las muestras de autocovarianza corresponden a una vecindad GSM  $3 \times 3$ . Cada una de ellas se calcula con las muestras de la región marcada en la imagen.

El modelo subyacente es conceptualmente sencillo, robusto y eficiente. Su flexibilidad y eficiencia proporciona un método de estimación no iterativo y libre de re-estimaciones que hereda y extiende las buenas propiedades del modelo BLS-GSM original, como la capacidad de manejar ruido Gaussiano de densidad espectral arbitraria.

### 3.2. Motivación

El modelo GSM en el dominio ondicular, que presentamos en el Capítulo 2, basa su potencia en la capacidad de representar las fuertes fluctuaciones de la varianza local existente entre vecindades de coeficientes wavelet situadas en distintas posiciones espaciales. Esta heterogeneidad de la varianza está presente en las imágenes naturales y explica la mayoría de las dependencias no lineales observadas, como la alta kurtosis y el acoplamiento en amplitud de coeficientes vecinos. Como describimos anteriormente (ver Sección 2.5.2), el modelo GSM original asume una covarianza fija (normalizada) para todas las vecindades dentro de una subbanda. Sin embargo, las imágenes naturales poseen diferentes patrones de correlación en distintas localizaciones espaciales. Esto ocurre incluso

en el dominio transformado, donde, a pesar de que el filtrado asociado a cada subbanda reduce estas diferencias, sigue existiendo cierta variabilidad. En la Figura 3.1 se ilustra este comportamiento mostrando una subbanda vertical de *Barbara* en la representación TIHP. Asumiendo vecindades GSM  $3 \times 3$ , hemos comparado la autocovarianza local estimada para toda la subbanda (modelo GSM original), con dos autocovarianzas estimadas usando dos regiones diferentes de contraste similar. Como podemos comprobar, la autocovarianza global extrae el comportamiento promedio dentro de la subbanda, mientras que las calculadas a partir de regiones localizadas extraen diferencias significativas (por ejemplo, en orientación, como en este caso), aumentando el potencial de adaptación espacial a la señal. En este Capítulo presentamos una mejora del modelo GSM original que, manteniendo la capacidad de adaptación ( *fina* ) a la varianza, introduce un segundo nivel de adaptación ( *grueso* ), donde una vecindad mayor es utilizada para proporcionar adaptación a la covarianza local. A lo largo del Capítulo describiremos el nuevo modelo, SVGSM (del inglés, *Spatially Variant Gaussian Scale Mixture*), y presentaremos un método de eliminación de ruido, que al igual que el modelo original, se basa en un estimador BLS [11, 16].

### 3.3. GSM espacialmente adaptativas: SVGSM

Considerando una imagen descompuesta en subbandas orientadas, definimos  $x_c^j(n, m)$  como el coeficiente de *referencia* perteneciente a la subbanda  $j$ , centrado en la posición espacial  $(n, m)$ . El vector  $\mathbf{x}^j(n, m)$  se define como una *vecindad* de coeficientes agrupados en torno al coeficiente de referencia. Modelamos estas vecindades como mezcla en escala de Gaussianas (GSMs). Para dotar al modelo de adaptación espacial, definimos  $\mathbf{C}_u^{j,(n,m)}$  como la covarianza (espacialmente variante) de  $\mathbf{u}^j(n, m)$ . La densidad de  $\mathbf{x}$  viene determinada por  $p(z)$  y  $\mathbf{C}_u^{j,(n,m)}$  como sigue:

$$\begin{aligned} p^{j,(n,m)}(\mathbf{x}) &= \int p^{j,(n,m)}(\mathbf{x}|z) p(z) dz \\ &= \int \frac{\exp\left(-\mathbf{x}^T (z\mathbf{C}_u^{j,(n,m)})^{-1} \mathbf{x} / 2\right)}{(2\pi)^{N/2} |z\mathbf{C}_u^{j,(n,m)}|^{1/2}} p(z) dz, \end{aligned}$$

donde  $N$  (el tamaño de la vecindad) es la dimensión de  $\mathbf{x}$  y  $\mathbf{u}$ . Nótese que  $\mathbf{x}^j(n, m)$  es condicionalmente Gaussiano para un  $z$  dado. Al igual que en [10], se asume la densidad de  $z$ ,  $p(z)$ , constante en  $\log(z)$  para todas las subbandas en un determinado rango.

El modelo de imagen presentado, SVGSM, sigue explotando la capacidad de adaptación local a la varianza (adaptación  *fina* ), gracias al uso de GSM. De igual forma, sigue conservando la propiedad de que los vectores  $\mathbf{x}$  sean condicionalmente Gaussianos (dado un  $z$ ). La principal diferencia se encuentra en la inclusión de

otro nivel de adaptación (*grueso*) gracias al uso de una covarianza variable que depende de la posición espacial.

### 3.4. Eliminación de ruido usando SVGSM

En esta Sección presentamos un método de eliminación de ruido basado en el modelo anteriormente descrito. El método, al igual que su predecesor presentado en la Sección 2.6, se basa en un estimador de mínimos cuadrados de Bayes (BLS). Por ello, para aligerar la nomenclatura, prescindiremos en lo sucesivo del prefijo BLS, refiriéndonos indistintamente a los modelos o algoritmos de eliminación de ruido como GSM o SVGSM<sup>2</sup>.

De la misma forma que el GSM original, el modelo de observación SVGSM expresa una vecindad  $L \times L$  formada por  $N (= L^2)$  coeficientes ruidosos de una determinada subbanda  $j$  como:

$$\mathbf{y}^j(n, m) = \sqrt{z}\mathbf{u}^j(n, m) + \mathbf{w},$$

donde  $\mathbf{w}$  es un vector de ruido Gaussiano  $N$ -dimensional con matriz de covarianza  $\mathbf{C}_{\mathbf{w}}$ . Nótese que, en contraste con la covarianza de la señal  $\mathbf{C}_{\mathbf{u}}^{j,(n,m)}$  (dependiente de la posición espacial),  $\mathbf{C}_{\mathbf{w}}$  se asume constante para todas las vecindades de una subbanda (pero diferente para cada subbanda)<sup>3</sup>. Asumida la independencia del ruido y aplicando la definición de GSM, resulta que la densidad del vector observado condicionado en  $z$ ,  $p(\mathbf{y}|z)$  sigue una distribución Gaussiana,  $\mathcal{N}(0, z\mathbf{C}_{\mathbf{u}} + \mathbf{C}_{\mathbf{w}})$ .

#### 3.4.1. Estimación de parámetros

Los parámetros del modelo son la covarianza del ruido,  $\mathbf{C}_{\mathbf{w}}$ , la covarianza de la señal,  $\mathbf{C}_{\mathbf{u}}$ , y la densidad del multiplicador oculto,  $p(z)$ . El modelo SVGSM asume que la estadística del ruido no cambia con la posición espacial dentro de la imagen, por lo que puede ser estimada de la misma forma que en el modelo GSM original (ver Sección 2.6.1). La diferencia fundamental con el método GSM original se encuentra en la estimación de la matriz de covarianza  $\mathbf{C}_{\mathbf{u}}$  en las distintas posiciones espaciales. Para ello asumimos que  $\mathbf{C}_{\mathbf{u}}$  cambia suavemente dentro de la subbanda, con lo que puede ser muestreada mediante una estimación por regiones no solapadas<sup>4</sup>. Dividimos cada subbanda en bloques de tamaño  $B \times B$ ,

<sup>2</sup>En los casos en los que exista ambigüedad, utilizaremos la palabra "método" o "modelo" para referirnos al algoritmo de restauración o al modelo como tal, respectivamente.

<sup>3</sup>Para simplificar la notación, de aquí en adelante prescindiremos del superíndice  $j$  y los subíndices  $(n, m)$ . Es importante destacar que con esta simplificación las ecuaciones poseen la misma expresión que en el caso de GSM original, pero el lector no debe olvidar que  $\mathbf{C}_{\mathbf{u}}$  representa una covarianza local, al depender de la posición espacial.

<sup>4</sup>Hemos comprobado empíricamente que la ligera mejora conseguida usando bloques solapados no compensa el ingente aumento de computación que esta técnica implica.

que se asumen uniformes en covarianza. En cada bloque, la forma de proceder es equivalente al modelo GSM original, obteniendo  $\mathbf{C}_u$  mediante la resta de la covarianza del ruido  $\mathbf{C}_w$  a la covarianza estimada del bloque observado,  $\hat{\mathbf{C}}_y$ , forzando que sus autovalores no sean negativos.

### 3.4.2. Estimación de coeficientes

La estimación de los coeficientes se hace de forma idéntica al método GSM, esto es, cada coeficiente central  $x_c$  de una vecindad observada  $\mathbf{y}$  es estimado mediante mínimos cuadrados de Bayes (BLS). La solución se planteaba como una suma ponderada de soluciones de Wiener:

$$\mathbb{E}\{x_c|\mathbf{y}\} = \int_0^\infty p(z|\mathbf{y}) [\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\}]_{x_c} dz,$$

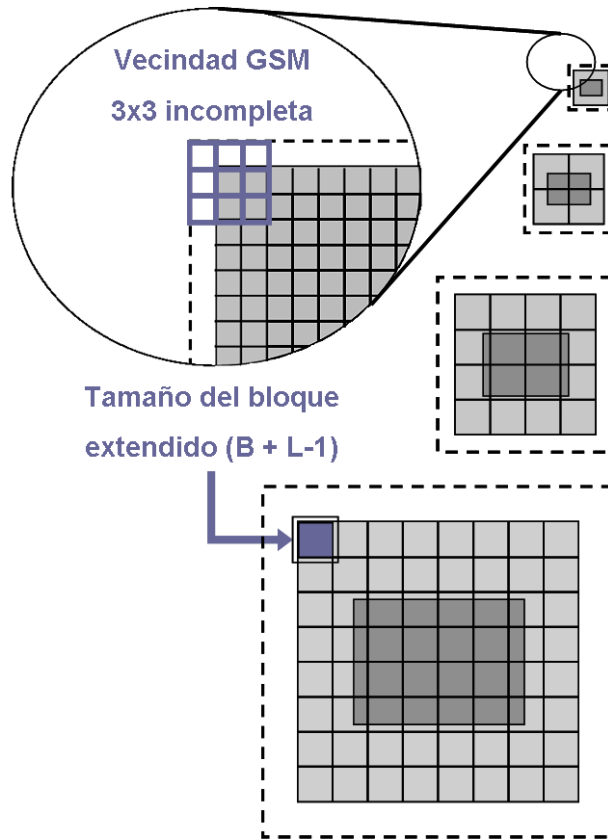
con  $\mathbb{E}\{\mathbf{x}|\mathbf{y}, z\} = z\mathbf{C}_u(z\mathbf{C}_u + \mathbf{C}_w)^{-1}\mathbf{y}$ . Recordemos que en el método GSM se realizaba una doble diagonalización de esta expresión, para simplificar la dependencia de  $z$ , y disminuir así el coste computacional. En el caso de SVGSM esta diagonalización tiene que computarse una vez por cada bloque (al depender  $\mathbf{C}_u$  de la posición espacial) en lugar de una vez por subbanda. Obviamente esto, junto con el tratamiento espacialmente variante por bloques, supone un cierto aumento del coste computacional con respecto al método GSM original.

## 3.5. Implementación

### 3.5.1. Tratamiento de bordes y organización por bloques

Como se demuestra en [10], un cuidadoso tratamiento de bordes es fundamental para obtener buenos resultados de eliminación de ruido. En el caso de SVGSM este tratamiento es aún más crítico, ya que las subbandas son divididas en bloques. Para ilustrar el problema, consideremos una vecindad GSM  $L \times L$  en una descomposición piramidal de  $N_s$  escalas. En la parte de arriba de la Figura 3.2, podemos observar una esquina de la subbanda perteneciente a la escala más alta de la pirámide (consideramos aquí  $L = 3$  y  $N_s = 4$ ). Todas las vecindades GSM cuyos coeficientes de referencia  $x_c$  se encuentran en las fronteras de la subbanda están incompletas, ya que no incluyen los coeficientes marcados con línea discontinua. Para solventar este problema, la imagen es extendida mediante reflejado (sin repetición) un total de  $2^{(N_s-1)}(L-1)/2$  píxeles por cada lado, lo que proporciona los coeficientes extra para la subbandas más altas de la pirámide (ver los cuadrados de líneas discontinuas en la Figura 3.2).

En los métodos basados en bloques, los artefactos por efectos de bordes son potencialmente mayores, por tener mayor proporción de coeficientes frontera. Al igual que previamente, cada bloque debe extenderse un total de  $(L-1)/2$  coeficientes por cada lado. Los bloques internos son ampliados con



**Figura 3.2.** Tratamiento de bordes y distribución de los bloques en un esquema multi-resolución.

coeficientes reales, mientras que los bloques frontera usan los coeficientes que fueron extendidos por reflexión para toda la subbanda. La cantidad aumentada es la mínima necesaria y resulta ser adecuada, ya que hemos comprobado empíricamente que un aumento de la extensión de cada bloque no proporciona una mejora significativa.

Otra cuestión práctica importante es la distribución de los bloques dentro de las subbandas. Para tener bloques completos en todas las subbandas, debemos asegurar que las mismas tengan una dimensión que sea múltiplo entero del tamaño del bloque, es decir,  $\mathbf{I}_{h,v}/(2^s B)$  debe ser un vector de enteros para  $s = 0 \dots N_s - 1$ , siendo  $\mathbf{I}_{h,v}$  un vector formado por las dimensiones de la imagen. En general, las imágenes no cumplirán este requerimiento, por lo que será necesario realizar una extensión, de nuevo usando reflejado sin repetición, para que satisfagan dicha restricción (ver el área de sombreado oscuro en la Figura 3.2). En la práctica aplicamos una versión relajada de este criterio: solo aquellas subbandas cuyo tamaño sea mayor que el tamaño de bloque elegido deben satisfacer esta

Representación	<i>FSP (8 Orientaciones)</i>		<i>TIHP (3 Orientaciones)</i>	
Tamaño $\Rightarrow$	$256 \times 256$	$512 \times 512$	$256 \times 256$	$512 \times 512$
$B = 16$	108.51	1090.07	46.18	451.29
$B = 32$	39.85	312.55	16.40	132.25
$B = 64$	22.65	119.36	8.76	52.19
<i>Original GSM</i>	16.14	62.92	7.22	32.44

**Tabla 3.1.** Tiempo de computación promedio en segundos (10 repeticiones) comparando diferentes tamaños de imágenes, bloques y representaciones.

restricción. El resto, aquellas menores o iguales que un bloque, son procesadas completas.

### 3.5.2. Experimentos

En aras de realizar una comparación en condiciones equitativas, hemos prestado especial atención a la configuración de los parámetros, siguiendo prácticamente las mismas pautas presentadas en [10].

Para la realización de los experimentos hemos utilizado dos representaciones diferentes (descritas en la Sección 2.4): *Translation-Invariant Haar Pyramid* (TIHP) y *Full Steerable Pyramid* (FSP). El número de escalas depende del tamaño de la imagen, siendo 4 para imágenes de  $256 \times 256$  y 5 para imágenes de  $512 \times 512$ . El número de orientaciones para TIHP es de 3, mientras que para FSP hemos utilizado 8, al igual que en [10]. La vecindad GSM elegida ha sido de  $3 \times 3$  alrededor de cada coeficiente. En el caso de FSP hemos incluido un coeficiente adicional (*padre*), aquel en la misma posición de la escala superior, al igual que en [10]. Se han probado varios tamaños de bloque para estimar la covarianza de la señal:  $B = 16, 32, 64, 128, F$ , donde  $F$  denota el tamaño de la imagen completa. Tanto el tamaño de bloque  $B$ , como la estructura de vecindad, se mantienen constantes en todos los niveles de la pirámide, excepto por los niveles altos de la pirámide (que no tienen *padre*) y aquellas subbandas con tamaño menor que el tamaño de bloque seleccionado.

### 3.5.3. Coste computacional

Como guía, la Tabla 3.1 muestra el tiempo de computación (en segundos) de nuestro algoritmo implementado en código *Matlab*<sup>®</sup> 7.0 (no optimizado). Los experimentos han sido ejecutados sobre *Windows XP* en un procesador *Intel Pentium IV* a 3.0 GHz con 1Gb de RAM. La tabla muestra los tiempos para las dos representaciones utilizadas, con diferentes tamaños de bloque y diferentes tamaños de imagen. En la parte baja de la Tabla hemos incluido, como referencia, los tiempos de ejecución de GSM. Podemos extraer tres patrones básicos: (a) el

coste escala inversamente con  $B$  en aproximadamente un factor de 3 cuando disminuimos el tamaño del bloque a la mitad en cada dimensión, (b) el coste escala aproximadamente con un factor 8 al doblar en cada dimensión el tamaño de la imagen, y (c) el tiempo de computación es significativamente mayor, pero todavía comparable, al de GSM original. La principal diferencia, desde el punto de vista del coste, es el aumento de las llamadas a funciones. Mientras que SVGSM hace una llamada por bloque, GSM hace tan solo una por subbanda. Además, cada división del bloque entre 2 multiplica por 4 el número de llamadas, lo que tiene un gran impacto cuando se usa código interpretado no optimizado, como es el caso de nuestra implementación.

### 3.6. Resultados y discusión

En esta Sección presentamos los resultados obtenidos tras aplicar nuestro método, SVGSM, a un conjunto de imágenes test standard: *Barbara*, *Lena* (ambas de  $512 \times 512$ ), *House* y *Peppers* (ambas de  $256 \times 256$ ). Con objeto de evaluar el rendimiento del método en un amplio rango de niveles de ruido, hemos añadido a las imágenes ruido blanco Gaussiano con diferentes desviaciones típicas:  $\sigma_w = 5, 10, 15, 20, 25, 50$ . Para ilustrar los resultados visuales hemos seleccionado las imágenes de *Barbara* y *House*, por ser dos ejemplos complementarios en cuanto a contenido.

La discusión que se presenta está dividida, a grandes rasgos, en dos apartados. El primero analiza el comportamiento del método, estudiando sus características y comparándolo con el método GSM original. En el segundo, comparamos con los métodos que representan el estado del arte actual.

#### 3.6.1. Características del modelo

En la Tabla 3.2 se muestran los resultados de nuestro algoritmo, en términos de relación señal ruido pico (PSNR, del inglés, *Peak Signal-to-Noise Ratio*),  $10 \log_{10}(255^2/\sigma_e^2)$  en dB, donde  $\sigma_e^2$  representa el error cuadrático medio (MSE, del inglés, *Mean Square Error*). Incluimos los resultados para las dos representaciones, TIHP y FSP, y comparamos nuestro método (columnas etiquetadas como SVGSM) con el método GSM [10] aplicado sobre las mismas representaciones (etiquetadas como GSM). En la última columna se incluye el tamaño de bloque  $B$  que proporciona los mejores resultados. La Figura 3.3 muestra los resultados visuales obtenidos con las imágenes de *Barbara* y *House*. Para facilitar la visualización de los artefactos, se muestran recortes de  $80 \times 80$  píxeles. La segunda y tercera fila muestran los resultados para TIHP y FSP respectivamente, mientras que por columnas hemos incluido una comparación con GSM original. Presentamos a continuación una discusión de los siguientes aspectos:

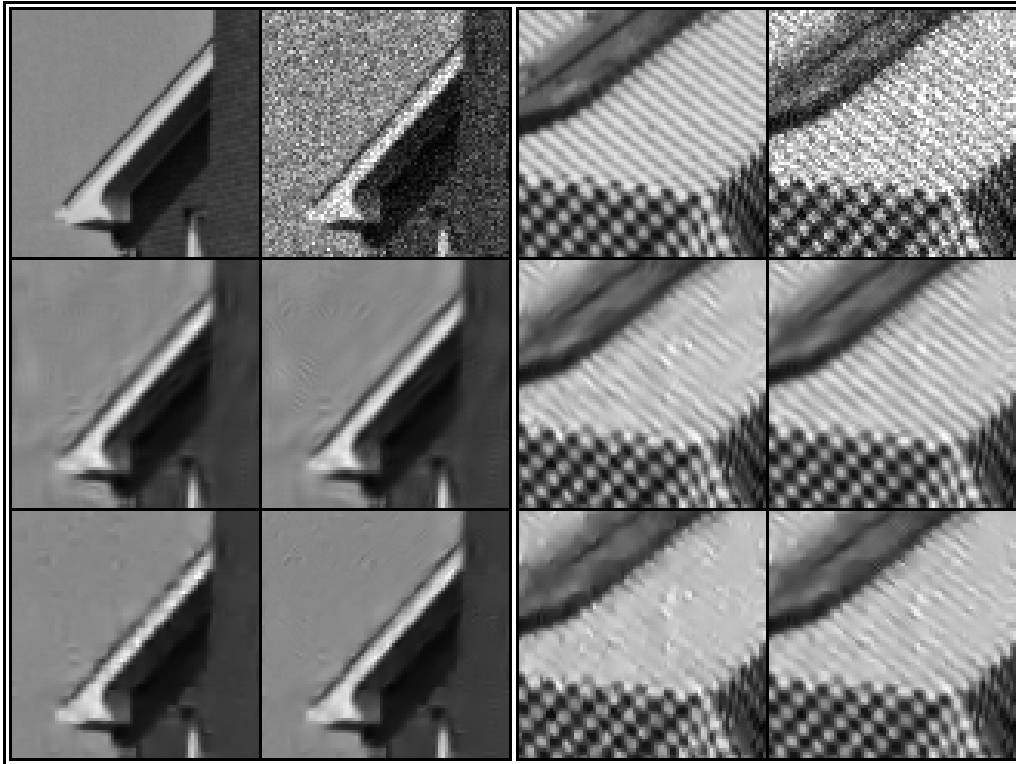
$\sigma_w$ / PSNR	FSP		TIHP		Bloque (Tamaño)
	GSM	SVGSM	GSM	SVGSM	
<i>Lena</i>					
5 / 34.15	38.49	<b>38.55</b>	38.19	38.34	32
10 / 28.13	35.60	<b>35.66</b>	35.23	35.38	64
15 / 24.61	33.90	<b>33.96</b>	33.50	33.65	64
20 / 22.11	32.67	<b>32.71</b>	32.25	32.40	64
25 / 20.17	31.69	<b>31.72</b>	31.26	31.40	64
50 / 14.15	<b>28.61</b>	<b>28.61</b>	28.21	28.21	<i>F</i>
<i>Barbara</i>					
5 / 34.15	37.78	<b>38.19</b>	37.19	37.84	16
10 / 28.13	34.02	<b>34.60</b>	33.13	34.13	16
15 / 24.61	31.83	<b>32.49</b>	30.76	31.85	16
20 / 22.11	30.27	<b>30.94</b>	29.08	30.19	16
25 / 20.17	29.07	<b>29.76</b>	27.81	28.81	32
50 / 14.15	25.42	<b>26.04</b>	24.33	25.03	32
<i>House</i>					
5 / 34.15	38.65	<b>38.88</b>	38.23	38.55	16
10 / 28.13	35.29	35.44	35.32	<b>35.56</b>	16
15 / 24.61	33.54	33.64	33.73	<b>33.94</b>	32
20 / 22.11	32.30	32.34	32.54	<b>32.71</b>	32
25 / 20.17	31.32	31.30	31.59	<b>31.71</b>	32
50 / 14.15	28.22	28.18	28.36	<b>28.38</b>	128
<i>Peppers</i>					
5 / 34.15	37.30	37.43	37.57	<b>37.72</b>	16
10 / 28.13	33.74	33.81	34.06	<b>34.24</b>	16
15 / 24.61	31.7	31.72	32.02	<b>32.18</b>	16
20 / 22.11	30.27	30.24	30.57	<b>30.67</b>	16
25 / 20.17	29.17	29.10	29.45	<b>29.50</b>	16
50 / 14.15	25.94	25.94	<b>26.11</b>	<b>26.11</b>	<i>F</i>

**Tabla 3.2.** Resultados de eliminación de ruido medido como relación señal ruido pico, en dB. La primera columna muestra la desviación típica del ruido y el valor PSNR inicial. Las siguientes cuatro columnas corresponden a los resultados del método GSM [10] (etiquetado como GSM) y a nuestra aportación (etiquetada como SVGSM) utilizando las representaciones TIHP y FSP (se resaltan en negrita los mejores resultados). En la última columna se muestra el tamaño de bloque  $B$  utilizado para SVGSM, siendo  $F$  el tamaño de la imagen completa.

### *Comparación con el método GSM*

La mejora del modelo presentada en la Sección 3.3 proporciona un incremento generalizado del rendimiento respecto al modelo GSM de referencia, tanto visualmente como en términos de error cuadrático medio (MSE). Comparando las





**Figura 3.3.** Comparación visual de resultados para las imágenes de *House* y *Barbara*. Se muestran regiones de  $80 \times 80$  píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). El tamaño de bloque  $B$  utilizado es de  $32 \times 32$  para *House* y  $16 \times 16$  para *Barbara*. De izquierda a derecha y de arriba a abajo se muestra (*House/Barbara* PSNR en dB): original; ruidosa utilizando  $\sigma_w = 20$  (22.11/22.11); método GSM [10] usando FSP (32.30/30.27); nuestro método con FSP (32.34/30.94); método GSM [10] usando TIHP (32.54/29.08); nuestro método con TIHP (32.71/30.19).

columnas de la Tabla 3.2 podemos observar que la mejora es significativa y generalizada para ambas representaciones (TIHP-FSP) y niveles de ruido. Es importante destacar que para imágenes con mucho contenido en textura, como por ejemplo *Barbara*, la mejora es comparativamente mayor que para el resto de imágenes. Este comportamiento está en relación directa con la variabilidad espacial en covarianza existente en diferentes regiones de la imagen (ver la Figura 3.1). Podemos comparar las columnas de la Figura 3.3 para observar las diferencias visuales de nuestro método con el modelo original. Como vemos, muchos de los artefactos de la línea del tejado en *House* desaparecen con el nuevo método (segunda columna). Para *Barbara* se puede observar una mejora importante de la calidad, tanto en disminución de artefactos como en recuperación de la textura.

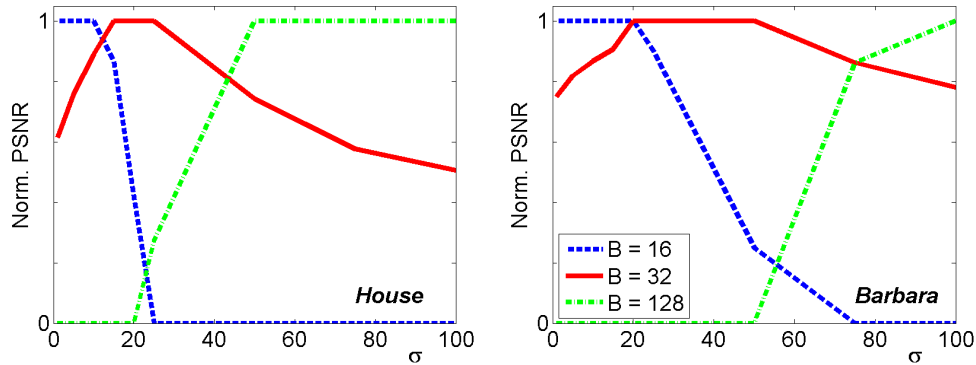
### *Comparación de representaciones: FSP vs. TIHP*

Como podemos ver en la Tabla 3.2, los resultados confirman lo que avanzábamos en la Sección 2.4. La idoneidad de una representación depende del contenido de la imagen. En particular podemos ver que *House* y *Peppers*, dos imágenes simples con poco contenido en textura, son mejor procesadas utilizando TIHP, mientras que para imágenes como *Lena* y *Barbara*, con mucha más riqueza en contenido, la representación más adecuada es FSP. También podemos apreciar diferencias importantes en términos visuales, observando las imágenes de *House* y *Barbara* en la segunda (FSP) y tercera fila (TIHP) de la Figura 3.3. En el caso de *House*, aun proporcionando un resultado con menos artefactos, la diferencia es modesta. Pero en el caso de *Barbara*, la diferencia es asombrosa: FSP proporciona un resultado mucho mejor que el obtenido con TIHP. En definitiva, ante el dilema de elegir una representación y pensando solo en términos de calidad (sin considerar el coste computacional), FSP sería la representación más adecuada. De manera adicional, y al igual que otros autores [55], hemos analizado la mejora obtenida al promediar los resultados de ambas representaciones, obteniendo una mejora media de 0.15 dB sobre el mejor resultado.

Otro dato a destacar acerca de las representaciones, son las diferencias de ganancias relativas respecto al método GSM. Comparando en la Tabla 3.2 las diferencias (por filas) de SVGSM con respecto al método de referencia, podemos observar que la mejora es mayor cuando utilizamos TIHP. La principal explicación a este comportamiento se encuentra en las características de las representaciones. Los filtros de FSP son muy selectivos en frecuencia, todo lo contrario a TIHP. Debido a ello, en esta última, existe una mayor variabilidad espacial de la covarianza dentro de cada subbanda. Esto implica una mayor diferencia, en promedio, entre las covarianzas locales (SVGSM) y la covarianza global usada por el modelo original (ver Figura 3.1).

### *Tamaño del bloque*

El tamaño de bloque óptimo depende tanto del contenido de la imagen como del nivel de ruido. Cuanto menor es la relación señal ruido, mayor es el tamaño del bloque óptimo. La explicación a este comportamiento se encuentra en la calidad de las estimaciones: a medida que el nivel de ruido sube, se hace necesario un mayor número de muestras para obtener una estimación razonable. En la Figura 3.4 mostramos gráficamente este comportamiento, tomando como ejemplo las imágenes de *Barbara* y *House*. Podemos observar una gráfica de PSNR (ajustada entre 0 y 1) para distintos tamaños de bloque ( $B = 16, 32$  y  $128$ ) y niveles de ruido. El nivel 1 representa el mejor resultado para el conjunto de bloques analizados, mientras que 0 representa el peor. Como esperábamos, el mejor resultado para niveles de ruido bajo se obtiene con el tamaño de bloque menor,  $B = 16$ , mientras que  $B = 128$  proporciona el mejor rendimiento para niveles altos de ruido. En general, parece que una adaptación muy *fina*

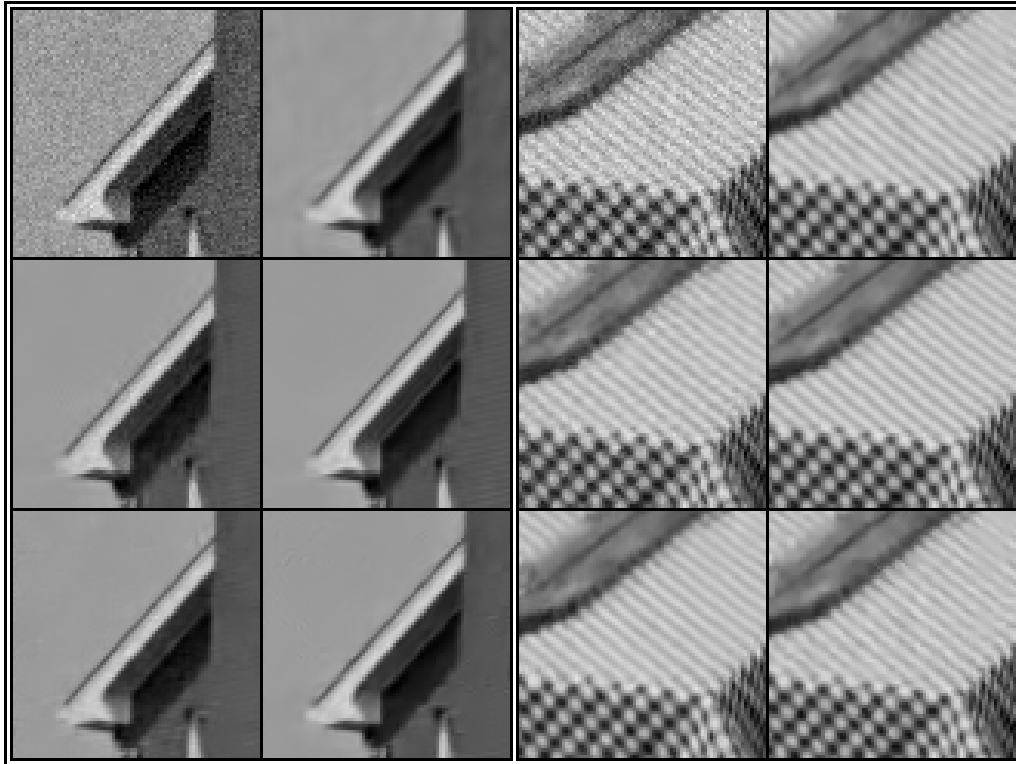


**Figura 3.4.** Gráficas PSNR vs. desviación típica del ruido para las imágenes de *House* y *Barbara* (ajustadas entre 0 y 1) utilizando diferentes tamaños de bloque  $B$ . Línea discontinua azul para  $B = 16 \times 16$ , continua roja para  $B = 32 \times 32$  y línea verde punteada para  $B = 128 \times 128$ . Nótese como el tamaño de bloque óptimo se incrementa con el nivel de ruido.

( $B = 16$ ) proporciona mejores resultados que una adaptación más *gruesa* (léase, tamaño de bloque mayor). Sin embargo, las diferencias en términos de rendimiento no son muy grandes, todo lo contrario que el coste computacional. Por ello, analizando las curvas de la Figura 3.4, podemos establecer  $B = 32$  como un buen compromiso práctico entre calidad y coste computacional, proporcionando así el mejor rendimiento para niveles de ruido moderados y una caída poco pronunciada en los niveles extremos.

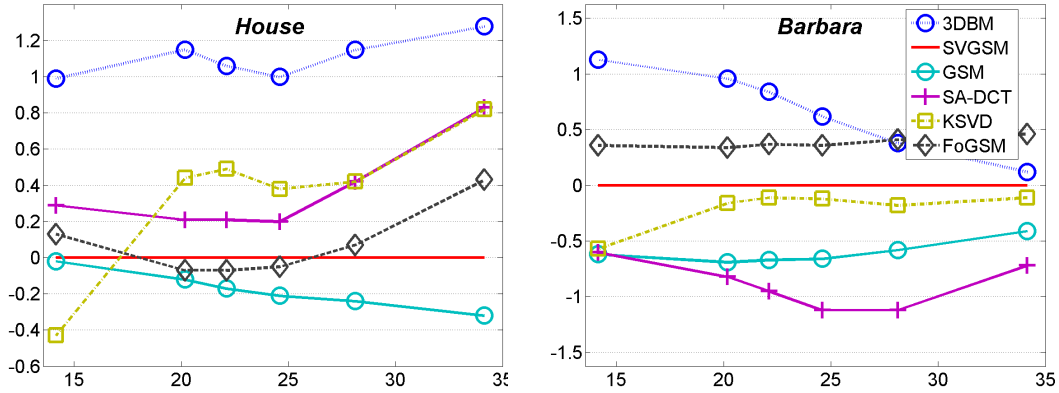
### 3.6.2. Comparación con el estado del arte

En la Figura 3.5 mostramos una comparación visual de nuestro algoritmo con cuatro métodos que, a nuestro leal conocer, proporcionan los mejores resultados de la literatura (en términos PSNR). Dos de ellos [53, 54] (SA-DCT y KSVD) se basan en la utilización de representaciones que se adaptan al contenido de la imagen, mientras que el propuesto en [52] (3DBM) representa una solución ad-hoc que explota la alta redundancia existente en las imágenes. Por último, el método presentado en [56] (FoGSM) utiliza como soporte un potente modelo global basado en campos de mezclas en escala de Gaussianas. En el caso de la imagen de *House* nuestro método se comporta peor en términos de error cuadrático, principalmente porque nuestros competidores consiguen recuperar la textura presente en la fachada. Sin embargo, la apariencia visual es bastante convincente. Por ejemplo, en la línea del tejado, podemos observar que es más nítida y tiene menos artefactos que el resultado proporcionado por [53]. En el caso de *Barbara* nuestro método se comporta de forma similar al resto, tanto visualmente como en términos MSE. Por último, en la Figura 3.6 podemos ver una comparación gráfica en términos



**Figura 3.5.** Comparación visual de resultados para las imágenes de *House* y *Barbara*. Se muestran regiones de  $80 \times 80$  píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo se muestra (*House/Barbara* PSNR en dB): observaciones ruidosas utilizando  $\sigma_w = 10$  (28.13/28.13); resultados de FoGSM [56] (35.63/35.01); resultados de SA-DCT [53] (35.98/33.48); resultados de 3DBM [52] (36.71/34.98); resultados de KSVd [54] (35.94/34.42); nuestros resultados usando  $B = 16 \times 16$  y las representaciones TIHP/FSP respectivamente (35.56/34.60).

de PSNR inicial vs. incremento de PSNR, tomando como referencia el método presentado (SVGSM, línea continua roja centrada en cero). Se incluyen en la comparativa cuatro de los mejores métodos del estado del arte [52, 53, 54, 56] y el método predecesor basado en GSM [10] (línea continua turquesa con círculos). Como podemos ver, tanto en *House* (usando TIHP) como en *Barbara* (usando FSP), el comportamiento de SVGSM es bastante competitivo con el estado del arte actual [52], considerando especialmente el moderado coste computacional. Remitimos al lector a [57] para una reciente comparación perceptual de algunos de los métodos aquí comparados, en la cual este método fue clasificado entre los tres mejores.



**Figura 3.6.** Incremento PSNR vs. PSNR de entrada, en dB, tomando como referencia el método presentado, SVGSM. Utilizamos TIHP para *House* y FSP con *Barbara*. Nuestros resultados, para  $\sigma_w = \{5, 10, 15, 20, 25, 50\}$ , se encuentran en la Tabla 3.2. Comparamos con cuatro métodos del estado del arte: línea punteada con círculos (azul) para 3DBM [52]; línea continua con cruces (lila) para SA-DCT [53]; línea discontinua con cuadrados (amarilla) para KSVD [54]; línea continua con círculos (turquesa) para GSM [10]; línea discontinua con rombos (gris) para FoGSM [56]; línea continua (roja) para nuestra aportación.

### 3.7. Conclusiones

En este Capítulo se ha presentado una mejora del modelo de imagen GSM que permite, mediante el uso de GSMs espacialmente adaptativas, representar las características espectrales locales de la señal. El modelo presentado, SVGSM, que hereda toda la potencia del modelo GSM original, ha demostrado ser especialmente efectivo para adaptarse a la variabilidad espacial en covarianza existente dentro de las subbandas. El algoritmo de eliminación de ruido propuesto, de aplicación a cualquier representación piramidal (tipo wavelet), muestra una mejora significativa y generalizada de los resultados, por la que se paga un moderado precio en incremento de coste computacional. Hemos analizado el comportamiento del método con dos representaciones complementarias, TIHP y FSP, concluyendo que FSP resulta más adecuada para imágenes texturadas mientras que TIHP se comporta mejor con imágenes simples. Cabe destacar también, que el método presentado tiene un mayor potencial de mejora respecto al GSM original cuando se aplica a representaciones cuyos filtros sean poco localizados en frecuencia (como es el caso de TIHP), al poseer dichas representaciones subbandas con mayor variabilidad en covarianza. En resumen, el método está basado en un modelo consistente, que pese a su simplicidad proporciona resultados competitivos con el estado del arte.

## Capítulo 4

# Modelado con múltiples mezclas en escala de Gaussianas

### 4.1. Introducción

En la introducción del Capítulo anterior mostramos un amplio repaso de algunos de los modelos de imagen de la literatura actual, incluyendo también una descripción de los métodos de eliminación de ruido. De la misma, pudimos extraer que aquellos métodos que explotan la redundancia espacial en contenidos existente en las imágenes naturales, son los que mejores resultados obtienen, incluso siendo algoritmos ad-hoc sin modelos estadísticos que les den soporte. Este hecho nos hace pensar que los modelos de imagen que aprovechen, de alguna forma, dicha redundancia, serán buenos candidatos a mejorar el rendimiento de las aplicaciones que los utilicen. Por un lado, es un hecho que, ni el uso de un modelo GSM simple por subbanda [10], ni la mejora basada en GSMs simples espacialmente variantes (SVGSM [11, 16]) son aproximaciones adecuadas para caracterizar la redundancia no local de las imágenes. Por contra, el modelo GSM tiene magníficas cualidades de adaptación a la energía local, superiores a muchos otros métodos. Lo ideal sería poder construir un modelo que contara con las ventajas de GSM y que, además, permitiera la existencia de múltiples matrices de covarianza (múltiples GSMs), sin estar restringidas a una determinada región espacial.

En este Capítulo proponemos un modelo basado en una "mezcla de mezclas en escala de Gaussianas" (MGSM, del inglés, *Mixture of Gaussian Scale Mixtures*) que añade a la potencia y flexibilidad de los modelos GSM, dos principales mejoras: (a) la capacidad de adaptación local a estructuras diferentes, proporcionando un único modelo local para todas las vecindades de cada subbanda, y (b) la capacidad de explotar la redundancia no local existente en las imágenes naturales. Estas mejoras proporcionan una ventaja sustancial, ya que a través de una sola covarianza podemos capturar características espaciales similares que se encuentren en posiciones diferentes de la imagen (en general,

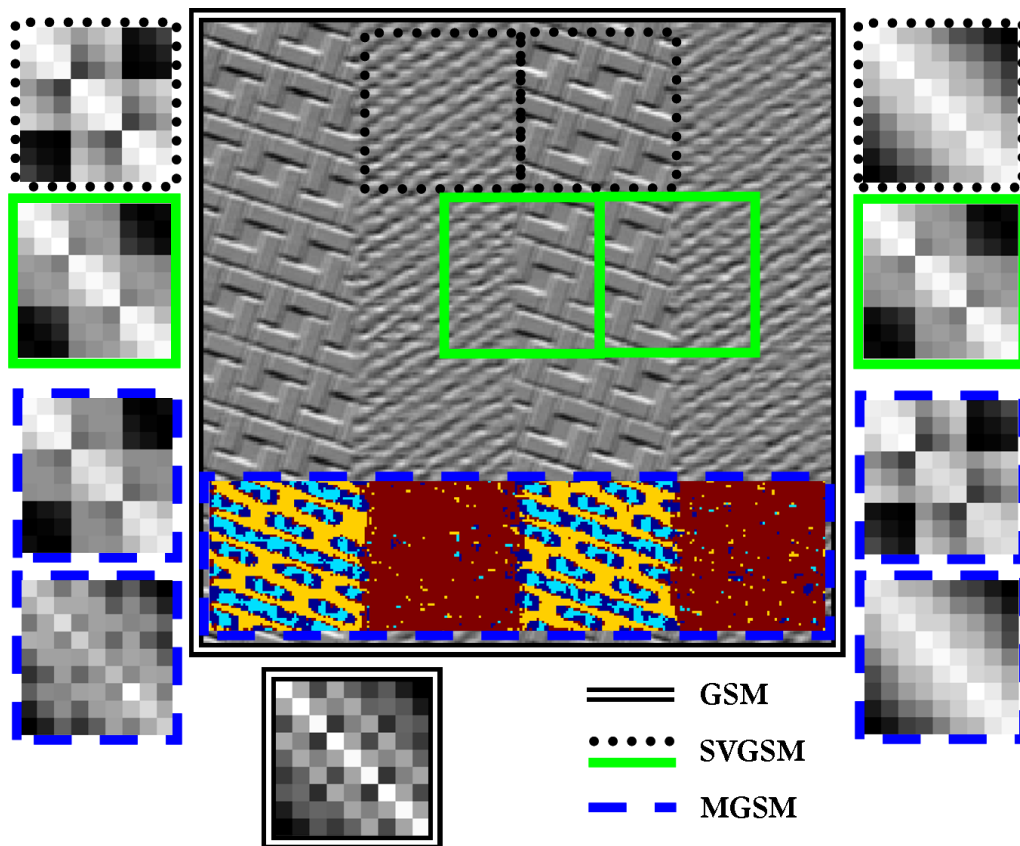
no adyacentes). Como consecuencia, el aprovechamiento de la redundancia no local proporciona una caracterización mucho más precisa de las propiedades de la imagen, aportando un plus al potencial de adaptación del modelo.

El modelo presentado podría ser empleado en gran variedad de aplicaciones, entre ellas, restauración, separación de texturas, clasificación, compresión, etc... Particularmente, en este trabajo hemos utilizado el modelo MGSM para eliminación de ruido [18]. Como veremos a lo largo del Capítulo, los resultados obtenidos mejoran de forma sistemática los proporcionados por nuestro modelo previo (SVGSM) y se comportan de forma comparable al estado del arte.

## 4.2. Motivación

En el Capítulo 3 describimos las limitaciones que presentaba el modelo GSM original, el cual no era capaz de representar de forma precisa los distintos patrones de correlación espacial presentes en una subbanda. Aunque la solución presentada en ese Capítulo, basada en un modelo GSM espacialmente variante (SVGSM), supuso una mejora significativa y generalizada sobre el modelo GSM, aún sigue existiendo un gran margen de mejora. En la Figura 4.1 se muestra una subbanda vertical de una imagen sintética formada por cuatro barras de dos texturas diferentes. Considerando una vecindad GSM  $3 \times 3$ , hemos estimado la covarianza<sup>1</sup> asociada ( $9 \times 9$ ) de tres formas distintas. En la primera de ellas, el modelo GSM original utiliza todas las posibles vecindades de la subbanda para calcular la covarianza, mezclando muestras GSM con estructura muy diversa. La covarianza resultante (enmarcada en la figura con una línea doble negra) posee la correlación "promedio" de toda la subbanda. En contraste, como describimos en el Capítulo anterior, la mejora espacialmente variante (SVGSM) trata de solventar el problema estimando covarianzas a partir de regiones no solapadas, realizando un muestreo por bloques de la subbanda. De esta forma, las covarianzas extraídas (enmarcadas en la figura con un cuadro de líneas punteadas negras) capturan correlaciones muy diferentes, incrementando así la capacidad de adaptación a la señal. Sin embargo, como hemos comentado, la adaptación local reduce las limitaciones de GSM original, pero no las supera por completo, ya que: (a) incluso dentro de las regiones puede existir variabilidad espacial y, lo que es aún más importante, (b) las características dentro de una subbanda no están localizadas espacialmente, sino que en general estarán distribuidas por toda la subbanda en regiones no adyacentes. Este hecho fundamental está siendo ignorado por el modelo SVGSM, que desperdicia completamente la valiosa redundancia no local existente en las subbandas. Otro aspecto importante a destacar es la importancia del muestreo realizado (bloques). Las covarianzas de la Figura 4.1 enmarcadas

<sup>1</sup>Las covarianzas son estimadas mediante el promedio de los productos externos de todas las vecindades (vectorizadas) de una determinada región, para un soporte dado.



**Figura 4.1.** Variabilidad de la covarianza en una subbanda vertical de TIHP. La imagen utilizada está formada por cuatro barras de dos texturas diferentes. Asumiendo una vecindad GSM  $3 \times 3$ , cada covarianza es estimada utilizando las muestras de la región asociada (cuadros de color). Se muestran las estimaciones utilizadas por el modelo GSM original (toda la subbanda, enmarcada con doble línea negra), modelo SVGSM (cuadros enmarcados con líneas punteadas negras y continuas verdes) y nuestra propuesta utilizando MGSM, usando 4 GSMs (cuadros enmarcados con línea azul discontinua).

con un cuadro verde representan un muestreo de bloques diferente (desplazado) al anterior. Como podemos observar, las estructuras de correlación que están capturando difieren de las obtenidas anteriormente, con lo que, fijado un muestreo, la capacidad de adaptación del modelo depende de forma arbitraria del contenido de la imagen. Además, el procesamiento local por bloques implica la existencia de un compromiso entre adaptación y fiabilidad en la estimación. A medida que se aumenta el tamaño de bloque, el número de muestras dentro del mismo es mayor, lo que implica una disminución de la varianza de la estimación. Por otro lado, cuanto mayor sea el bloque, menor será la capacidad de adaptación local,



llegando en el extremo al modelo GSM original (un solo bloque para toda la imagen).

El modelo que aquí presentamos considera las vecindades de una subbanda como muestras de una mezcla de mezclas en escala de Gaussianas (MGSM). Este modelo permite representar tantas estructuras como GSMs formen la mezcla, eliminando las restricciones de localidad impuestas por el modelo de estimación local SVGSM y explotando la redundancia no local para una mejor estimación de los parámetros.

### 4.3. Mezcla de mezclas en escala de Gaussianas: MGSM

A diferencia de la solución local presentada en el Capítulo anterior, donde la adaptación a los distintos patrones de correlación se llevaba a cabo mediante el uso de GSMs espacialmente variantes (SVGSM), en esta Sección presentamos un modelo local que proporciona una caracterización global para cada subbanda. Para ello, las vecindades GSM de las mismas son modeladas como muestras de una mezcla de mezclas en escala de Gaussianas, que, al igual que en el modelo GSM simple, se definen como:

$$\mathbf{x} = \sqrt{z}\mathbf{u},$$

con la salvedad de que en esta ocasión asumimos la existencia de una variable discreta oculta  $k \sim P_k$  que selecciona y determina la densidad del multiplicador escalar oculto  $p(z|k) = p_k(z)$ , así como la matriz de covarianza  $\mathbf{C}_k$  del vector Gaussiano de media cero  $\mathbf{u}$ . Teniendo esto en cuenta, la función densidad de probabilidad de la vecindad vectorizada es:

$$\begin{aligned} p(\mathbf{x}) &= \sum_{k=1}^K P_k p(\mathbf{x}|k) \\ &= \sum_{k=1}^K P_k \int_z p(\mathbf{x}|k, z) p_k(z) dz, \end{aligned} \quad (4.1)$$

donde  $p(\mathbf{x}|k, z)$  es una distribución Gaussiana de media cero y matriz de covarianza  $z\mathbf{C}_k$ :

$$p(\mathbf{x}|k, z) = \frac{\exp(-\mathbf{x}^T (z\mathbf{C}_k)^{-1} \mathbf{x} / 2)}{(2\pi)^{N/2} |z\mathbf{C}_k|^{1/2}}. \quad (4.2)$$

De esta forma, el conjunto de parámetros del modelo MGSM, para una subbanda determinada, está formado por:

- Los valores de probabilidad  $P_k$ .

- El conjunto de las  $K$  densidades que gobiernan la mezcla en escala  $p_k(z) = p(z|k)$ .
- El conjunto de las  $K$  matrices de covarianza  $\mathbf{C}_k$ , parámetros a los que se añaden dos más: el número de componentes de la mezcla ( $K$ ) y la forma y tamaño ( $N$ ) de la vecindad considerada, que en este trabajo asumiremos conocidos.

Es importante destacar que, a diferencia de los modelos GSM original y SVGSM, donde se utiliza como densidad del factor de escala de la mezcla un prior (Jeffrey's prior), este modelo siempre estima las distribuciones a partir de la observación, proporcionando un conjunto de densidades de mezcla en escala adaptativas.

#### 4.3.1. Estimación de parámetros

Para calcular los parámetros del modelo proponemos utilizar un criterio de máxima verosimilitud (ML). Las muestras observadas vienen representadas por las  $M$  vecindades (asumidas independientes) de una determinada subbanda. El objetivo es obtener los parámetros que maximizan el logaritmo de la verosimilitud global,  $L$ , dadas las muestras extraídas:

$$L = \log \left[ \prod_{m=1}^M p(\mathbf{x}_m) \right] = \sum_{m=1}^M \log p(\mathbf{x}_m) = \sum_{m=1}^M \log \left[ \sum_{i=1}^K P_{k_i} \int_z p(\mathbf{x}_m|k_i, z) p_{k_i}(z) dz \right].$$

Para cada parámetro por separado, obtenemos expresiones que maximizan marginalmente  $L$ . En aras de simplificar el problema de optimización, reducimos las densidades de los multiplicadores ocultos  $p_k(z)$  a un número finito de dimensiones. Esto permite derivar secuencialmente la verosimilitud global con respecto a cada componente para obtener el óptimo marginal, e ir mejorando el resultado a cada iteración. Las expresiones recursivas que aseguran, en la convergencia, una solución de máxima verosimilitud son<sup>2</sup>:

*Probabilidad de cada componente de la mezcla,  $P_k$ :*

$$P_k^{(n+1)} = P_k^{(n)} \frac{1}{M} \sum_{m=1}^M \frac{p(\mathbf{x}_m|k)^{(n)}}{\sum_{j=1}^K p(\mathbf{x}_m|j)^{(n)} P_j^{(n)}}, \forall k. \quad (4.3)$$

*Funciones densidad de probabilidad de los factores de escala,  $p_k(z)$ , de cada componente:*

$$p_k(z)^{(n+1)} = \frac{p_k(z)^{(n)}}{M P_k^{(n)}} \sum_{m=1}^M \frac{p(\mathbf{x}_m|k, z)^{(n)} P(k|\mathbf{x}_m)^{(n)}}{p(\mathbf{x}_m|k)^{(n)}}, \forall k. \quad (4.4)$$

<sup>2</sup>Para un análisis detallado del cálculo de estas expresiones, referimos al lector al Apéndice A.

Matrices de covarianza,  $\mathbf{C}_k$ , de cada componente:

$$\mathbf{C}_k^{(n+1)} = \frac{\int_z \sum_{m=1}^M p(z|\mathbf{x}_m, k)^{(n)} P(k|\mathbf{x}_m)^{(n)} \mathbf{x}_m \mathbf{x}_m^T z^{-1} dz}{\int_z \sum_{m=1}^M p(z|\mathbf{x}_m, k)^{(n)} P(k|\mathbf{x}_m)^{(n)} z^{-1} dz}, \forall k. \quad (4.5)$$

Siendo todos los parámetros desconocidos (salvo  $K$  y  $N$ ), que asumimos conocidos), proponemos el uso de un algoritmo iterativo de tipo (G)EM (del inglés *Generalized EM*). Al ser las expresiones anteriores recursivas y a su vez depender de otras expresiones, es necesario proporcionar una estimación inicial de los parámetros así como una forma de calcular las expresiones intermedias.

### *Estimaciones iniciales*

Para  $P_k$  asumimos una distribución inicial uniforme. En el caso de las densidades de las mezclas en escala,  $p_k(z)$ , utilizamos un prior no informativo en el intervalo considerado (en los mismos términos que en [10]). La asignación inicial para las covarianzas es un poco más elaborada: las  $\mathbf{C}_k$ ,  $k = 1..K$  covarianzas iniciales son calculadas utilizando regiones de la subbanda que poseen alta energía, que son seleccionadas de forma heurística. A continuación se detallan los pasos del procedimiento:

1. Obtener el valor cuadrático de cada coeficiente de la subbanda (*mapa de energía*).
2. Obtener la posición espacial del valor máximo del mapa de energía (*posición de referencia*).
3. Seleccionar alrededor del coeficiente de referencia una región de la subbanda, en general centrada<sup>3</sup>, de tamaño<sup>4</sup>  $\sqrt{M/K} \times \sqrt{M/K}$ .
4. Asignar a  $\mathbf{C}_k$  la covarianza muestral de la región seleccionada.
5. Actualizar el mapa de energía, poniendo a cero los coeficientes de energía de la región utilizada en 4.
6. Volver al paso (2) hasta que  $k = K$ .

### *Expresiones intermedias*

Para actualizar los parámetros adecuadamente necesitamos calcular, entre otras, la expresión  $p(z|\mathbf{x}_m, k)$ . Usando la fórmula de Bayes:

$$p(z|\mathbf{x}_m, k) = \frac{p(\mathbf{x}_m|k, z)p_k(z)}{p(\mathbf{x}_m|k)}. \quad (4.6)$$

<sup>3</sup>Si la referencia se encuentra cercana a las fronteras, se realiza el mínimo desplazamiento de la referencia que asegure una región del tamaño deseado.

<sup>4</sup>La región seleccionada será siempre de tamaño impar, forzando a ello mediante resta de un coeficiente cuando sea necesario.

Esto implica evaluar  $p(\mathbf{x}_m|k, z)$ , que es una distribución Gaussiana de media cero y covarianza  $z\mathbf{C}_k$  (ver la Ecuación 4.2), y  $p(\mathbf{x}_m|k)$ , que es obtenida a partir de la integración numérica:

$$p(\mathbf{x}_m|k) = \int_z p(\mathbf{x}_m|k, z)p_k(z)dz. \quad (4.7)$$

Las distribuciones anteriores están involucradas en las expresiones de actualización de los parámetros, al igual que  $P(k|\mathbf{x}_m)$ , que también se obtiene aplicando la fórmula de Bayes:

$$P(k|\mathbf{x}_m) = \frac{p(\mathbf{x}_m|k)P_k}{p(\mathbf{x}_m)} = \frac{p(\mathbf{x}_m|k)P_k}{\sum_{j=1}^K p(\mathbf{x}_m|j)P_j}. \quad (4.8)$$

#### *Algoritmo de estimación*

En resumen, el algoritmo comprende los siguientes pasos:

1. Inicializar los parámetros:  $P_k, p_k(z), \mathbf{C}_k$ .
2. Calcular  $p(\mathbf{x}_m|k, z)$  mediante la Ecuación 4.2.
3. Calcular  $p(\mathbf{x}_m|k)$  mediante la Ecuación 4.7.
4. Calcular  $P(k|\mathbf{x}_m)$  mediante la Ecuación 4.8.
5. Calcular  $p(z|\mathbf{x}_m, k)$  mediante la Ecuación 4.6.
6. Actualizar los parámetros mediante las Ecuaciones 4.3, 4.4 y 4.5.
7. Ir a (2) hasta convergencia.

## 4.4. Eliminación de ruido usando MGSM

En esta Sección describimos un algoritmo de eliminación de ruido basado en el modelo presentado anteriormente. El objetivo es estimar una imagen original degradada por ruido aditivo Gaussiano (no blanco, en general) de densidad espectral de potencia conocida. Como todos los métodos presentados en este trabajo, la restauración se realiza en el dominio de las pirámides orientadas sobre-completas y el estimador se basa en mínimos cuadrados de Bayes (BLS). Al igual que en el Capítulo anterior, prescindiremos del prefijo BLS al hablar del método de eliminación de ruido, refiriéndonos indistintamente al modelo o al método de eliminación de ruido como MGSM<sup>5</sup>.

<sup>5</sup>En los casos en los que exista ambigüedad, utilizaremos la palabra "método" o "modelo" para referirnos al algoritmo de restauración o al modelo como tal, respectivamente.

#### 4.4.1. Modelo de observación

Siguiendo el modelo MGSM presentado en la Sección 4.3, modelamos las vecindades ruidosas de cada subbanda como una mezcla de mezclas en escala de Gaussianas, a la que se ha sumado ruido Gaussiano:

$$\mathbf{y} = \sqrt{z}\mathbf{u} + \mathbf{w},$$

donde  $\mathbf{w}$  es un vector de ruido Gaussiano de media cero y matriz de covarianza  $\mathbf{C}_w$ . La función densidad de probabilidad del vector formado por los coeficientes de la vecindad observada condicionada en  $z$  y  $k$  es también Gaussiana de media cero y con matriz de covarianza  $\mathbf{C}_{\mathbf{y}|k,z} = \mathbb{E}\{\mathbf{y}\mathbf{y}^T|k, z\} = z\mathbf{C}_k + \mathbf{C}_w$ . De forma análoga al modelo de señal, ahora la densidad del vector observado  $\mathbf{y}$  es:

$$p(\mathbf{y}) = \sum_{k=1}^K P_k \int_z \frac{\exp(-\mathbf{y}^T(z\mathbf{C}_k + \mathbf{C}_w)^{-1}\mathbf{y}/2)}{(2\pi)^{N/2}|z\mathbf{C}_k + \mathbf{C}_w|^{1/2}} p_k(z) dz.$$

#### 4.4.2. Estimación BLS de coeficientes

La estimación de mínimos cuadrados de Bayes (BLS) para una determinada observación  $\mathbf{y}$  es  $\hat{\mathbf{x}}_{BLS}(\mathbf{y}) = \mathbb{E}\{\mathbf{x}|\mathbf{y}\}$ . Para el cálculo de esta esperanza, es necesario conocer la densidad del posterior,  $p(\mathbf{x}|\mathbf{y})$ , que en este caso puede escribirse como sigue:

$$\begin{aligned} p(\mathbf{x}|\mathbf{y}) &= \sum_{k=1}^K P(k|\mathbf{y}) p(\mathbf{x}|\mathbf{y}, k) \\ &= \sum_{k=1}^K P(k|\mathbf{y}) \int_z p(\mathbf{x}, z|\mathbf{y}, k) dz \\ &= \sum_{k=1}^K P(k|\mathbf{y}) \int_z p(\mathbf{x}|\mathbf{y}, k, z) p_k(z|\mathbf{y}) dz. \end{aligned} \quad (4.9)$$

De este modo, el estimador BLS posee la siguiente expresión:

$$\begin{aligned} \hat{\mathbf{x}}_{BLS}(\mathbf{y}) &= \int_{\mathbf{x}} \mathbf{x} p(\mathbf{x}|\mathbf{y}) d\mathbf{x} \\ &= \int_{\mathbf{x}} \mathbf{x} \sum_{k=1}^K P(k|\mathbf{y}) \int_z p(\mathbf{x}|\mathbf{y}, k, z) p_k(z|\mathbf{y}) dz d\mathbf{x} \\ &= \sum_{k=1}^K P(k|\mathbf{y}) \int_z \int_{\mathbf{x}} \mathbf{x} p(\mathbf{x}|\mathbf{y}, k, z) d\mathbf{x} p_k(z|\mathbf{y}) dz \\ &= \sum_{k=1}^K P(k|\mathbf{y}) \int_z \mathbb{E}\{\mathbf{x}|\mathbf{y}, k, z\} p_k(z|\mathbf{y}) dz, \end{aligned} \quad (4.10)$$

donde:

$$\mathbb{E}\{\mathbf{x}|\mathbf{y}, k, z\} = z\mathbf{C}_k(z\mathbf{C}_k + \mathbf{C}_w)^{-1}\mathbf{y},$$

que es una solución de Wiener (ver Sección 2.3.1), ya que las densidades condicionales involucradas son Gaussianas. Finalmente, vemos que la integral en  $z$  de la Ecuación 4.10 corresponde a la estimación BLS para una GSM, con lo que podemos escribir la estimación final como una media ponderada de soluciones BLS-GSM, pesadas de acuerdo a la probabilidad a posteriori de los componentes de la mezcla:

$$\hat{\mathbf{x}}_{BLS}(\mathbf{y}) = \sum_{k=1}^K P(k|\mathbf{y}) \hat{\mathbf{x}}_{BLS-GSM}(\mathbf{y}; \mathbf{C}_k, p_k(z)). \quad (4.11)$$

#### 4.4.3. Estimación de parámetros

Al igual que en el caso del modelo MGSM de señal, en el caso de observación ruidosa también estimamos los parámetros de acuerdo a un criterio de máxima verosimilitud. De acuerdo con la Ecuación 4.11, los parámetros a estimar son  $\mathbf{C}_k$ ,  $p_k(z)$  y  $P(k|\mathbf{y})$ . Es importante destacar que, considerando que  $P(k|\mathbf{y})$  se calcula a través de la regla de Bayes, los parámetros principales a estimar son exactamente los mismos que en el modelo sin ruido, esto es,  $\mathbf{C}_k$ ,  $p_k(z)$  y  $P_k$ . De esta forma, el algoritmo iterativo propuesto anteriormente puede ser utilizado junto con las estimaciones iniciales y las expresiones intermedias, con solo tener en cuenta ligeras diferencias:

- La derivación de máxima verosimilitud de  $p_k(z)$  (Ecuación 4.4) y  $P_k$  (Ecuación 4.3), permanecen inalteradas, solo teniendo en cuenta que las funciones densidad de probabilidad involucradas se refieren a la señal observada  $\mathbf{y}_m$ , no a la original  $\mathbf{x}_m$ .
- Desafortunadamente no hemos encontrado un estimador ML eficiente para estimar  $\mathbf{C}_k$ . En su lugar hemos empleado un estimador consistente que da lugar a la siguiente expresión de actualización:

$$\mathbf{C}_k^{(n)} = \left[ \left( \int_z \mathbf{C}_y(k, z)^{(n)} p_k(z)^{(n)} dz \right) - \mathbf{C}_w \right]_+, \quad (4.12)$$

donde  $[\cdot]_+$  representa la puesta a cero de los autovalores negativos, y

$$\mathbf{C}_y(k, z)^{(n)} = \frac{\sum_{m=1}^M P(k|\mathbf{y}_m)^{(n)} p(z|\mathbf{y}_m, k)^{(n)} \mathbf{y}_m \mathbf{y}_m^T}{\sum_{m=1}^M P(k|\mathbf{y}_m)^{(n)} p(z|\mathbf{y}_m, k)^{(n)}}, \quad (4.13)$$

representa un estimador consistente para  $z\mathbf{C}_k + \mathbf{C}_w$ , es decir,  $\mathbf{C}_y(k, z) \rightarrow z\mathbf{C}_k + \mathbf{C}_w$  cuando  $M \rightarrow \infty$ , hecho que es demostrado en el Apéndice A. De

forma análoga a [10] hemos asumido que  $\int_z z p_k(z) dz = 1$ , para todo  $k$ . De ello se deriva que  $(\int_z (z\mathbf{C}_k + \mathbf{C}_w) p_k(z) dz) - \mathbf{C}_w = \mathbf{C}_k$ , lo que demuestra la consistencia del estimador propuesto, bajo la asunción anterior<sup>6</sup>.

## 4.5. Implementación

Para analizar el rendimiento del modelo MGSM hemos utilizado la pirámide orientada sobre-completa TIHP (ver Sección 2.4). El número de escalas utilizadas es 4 para imágenes de  $256 \times 256$  y 5 para imágenes de  $512 \times 512$ . En cuanto a los parámetros libres del modelo, hemos utilizado un total de  $K = 10$  GSMs por mezcla y, en contraste con los modelos GSM presentados anteriormente, las vecindades GSM se han fijado en  $7 \times 7$  coeficientes ( $N = 49$ ). La eliminación de ruido es realizada de forma independiente en cada subbanda, obteniendo la estimación final a partir de la reconstrucción de las mismas. El algoritmo realiza una *criba* de GSMs (reducción del número de GSMs que forman la mezcla) muy básica: las muestras son clasificadas de acuerdo a  $P(k|\mathbf{y}_m)$ , siendo eliminadas aquellas GSMs que poseen pocas muestras asignadas (menos que el número de coeficientes de la vecindad,  $N$ ), ya que es muy probable que no proporcionen estimaciones precisas. Los problemas numéricos (debidos, por ejemplo, a matrices mal condicionadas) son cuidadosamente tratados. Estos problemas suelen aparecer en las escalas altas de la pirámide, donde el número de muestras es relativamente pequeño, comparado con el número de parámetros a estimar. En estos casos, solo una GSM es utilizada (tratamiento que equivale al modelo GSM original).

## 4.6. Resultados y discusión

En esta Sección ilustramos el comportamiento de MGSM, incluyendo comparaciones con el estado del arte y con dos métodos descritos anteriormente: GSM original [10] (en adelante GSM) y la versión espacialmente variante del mismo [16] (SVGSM). Por claridad se divide la Sección en dos partes, una dedicada a describir las características del modelo y otra a evaluar el rendimiento del mismo cuando es aplicado a la eliminación de ruido.

### 4.6.1. Características del modelo

Como avanzamos al comienzo de este Capítulo, la principal motivación del modelo MGSM era superar algunas de las limitaciones que presentan modelos previos basados en mezclas en escala de Gaussianas. Estas se enumeran a continuación:

---

<sup>6</sup>Nótese que, aunque dichas integrales no cumplieran ser la unidad, el efecto de estos valores sobre la estimación sería meramente un factor de escala en las matrices de covarianza estimadas, lo que no afecta a las propiedades del método.

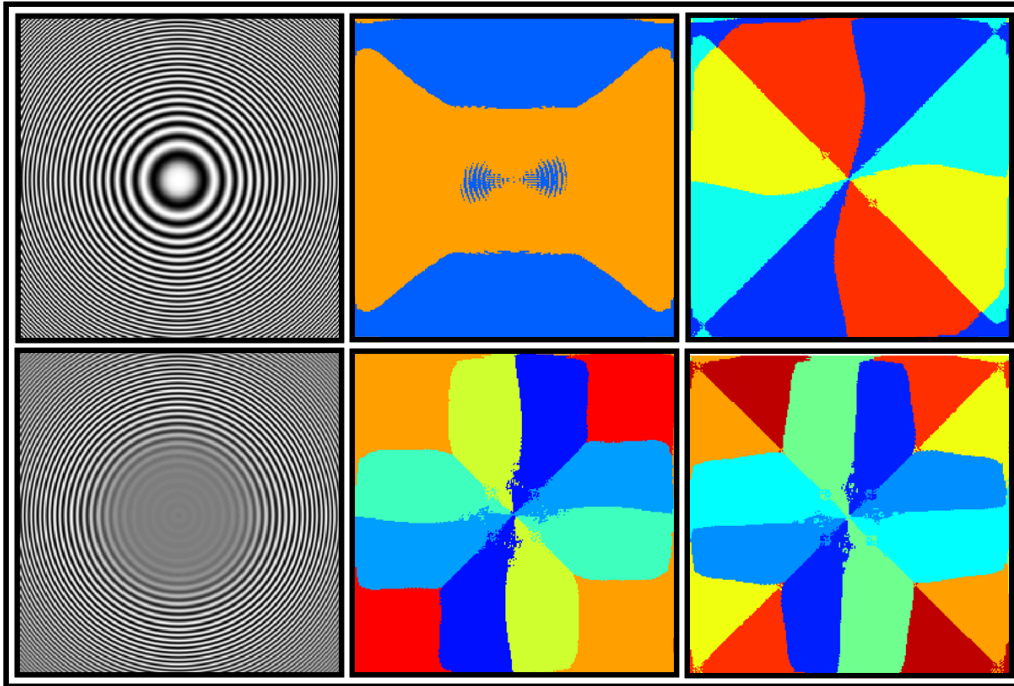
- GSM caracteriza, mediante el uso de una sola covarianza fija global, el comportamiento "promedio" de todas las vecindades de una subbanda.
- SVGSM proporciona una estimación adaptativa local, caracterizando las estructuras de determinadas regiones de la subbanda. Por ello:
  - No caracteriza la variabilidad espacial existente dentro de las regiones locales.
  - No utiliza información fuera de la región seleccionada, desperdiciando en la estimación el uso de muestras que pueden tener la misma estructura.
  - El muestreo de regiones implica un comportamiento variante a traslación, que depende tanto del contenido de la imagen como del tipo de muestreo realizado.

La Figura 4.1 fue utilizada en la Sección 4.3 para describir los problemas asociados a la estimación de covarianzas que acabamos de describir. Las covarianzas enmarcadas con línea discontinua azul son las representantes de cada mezcla cuando aplicamos MGSM con  $K = 4$  (número de GSMs en la mezcla). La zona inferior de la subbanda, enmarcada con cuadro discontinuo azul, representa en cada punto la componente  $k$  de la mezcla que maximiza la probabilidad a posteriori  $P(k|\mathbf{x}_m)$ , obtenida en el ajuste MGSM. Comparando esta región con las seleccionadas con el modelo SVGSM (cuadros verdes y negros punteados), queda claro que el modelo MGSM está caracterizando más de una estructura para una misma región. Más importante, incluso, es el hecho de que el modelo está utilizando todas las muestras con similar estructura presentes en la subbanda, aprovechando así la existencia de redundancia no local para obtener una mejor estimación. Además, al tratarse de un modelo local para todas las vecindades de la subbanda, no se ve afectado por el comportamiento variante a traslación inherente al modelo SVGSM (basado en un muestreo por bloques). Analizando la forma de las covarianzas estimadas usando MGSM, podemos observar en este ejemplo que el nuevo modelo está caracterizando estructuras muy similares a las caracterizadas conjuntamente por GSM y SVGSM (incluso en su versión desplazada), lo que demuestra que, además de superar las limitaciones comentadas, proporciona una mayor capacidad de adaptación.

El número de GSMs que se incluyen en la mezcla es un parámetro importante a determinar. Para analizar el efecto de este parámetro es necesario entender cómo se comporta el método con diferentes componentes GSMs ( $K$ ). En la Figura 4.2 se muestra un "chirp", una imagen con simetría circular ( $\cos(2\pi\omega^2)$ ), junto con una subbanda de alta resolución perteneciente a la pirámide de Simoncelli (*Steerable pyramid* [6]) sin orientaciones<sup>7</sup>. Sobre dicha subbanda

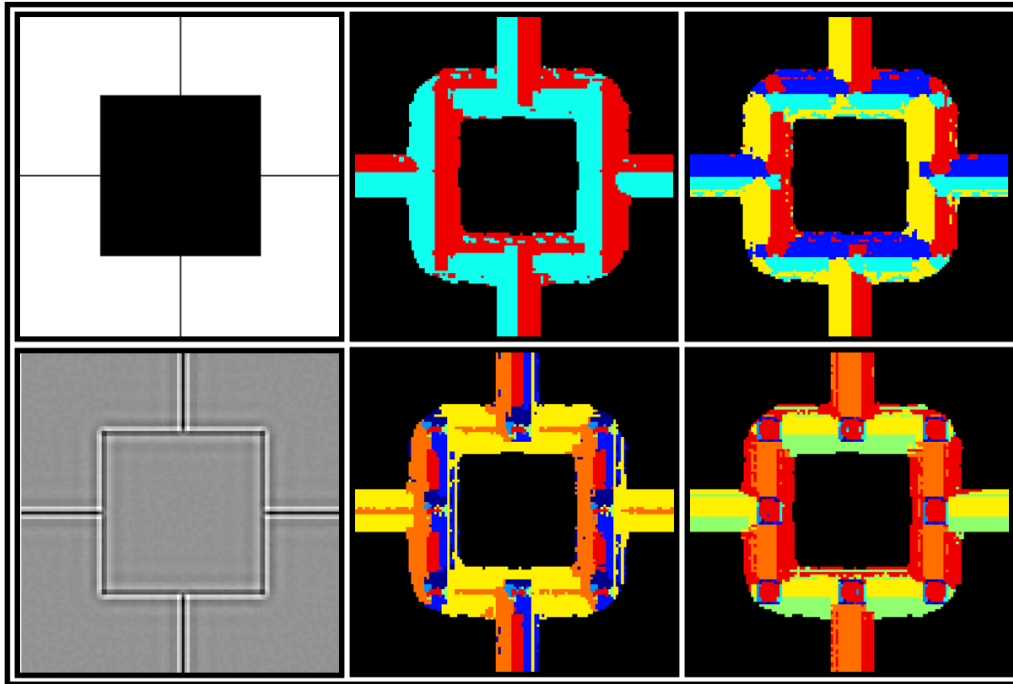
<sup>7</sup>En este caso hemos utilizado esta representación sin orientaciones, en lugar de TIHP, para no tratar de forma preferente ninguna orientación.





**Figura 4.2.** Comportamiento del modelo al aumentar el número de componentes de la mezcla ( $K$ ) usando vecindades  $7 \times 7$ . En la esquina superior izquierda, la imagen original. Debajo, una subbanda de alta resolución de la representación *Steerable pyramid* [6], sin orientaciones. El cuadro  $2 \times 2$  restante, de izquierda a derecha y de arriba a abajo, representan la clasificación de acuerdo a la probabilidad a posteriori  $P(k|\mathbf{x}_m)$  que obtiene el modelo cuando se utilizan  $K = 2, 4, 6$  y  $8$  componentes GSM en el ajuste del modelo.

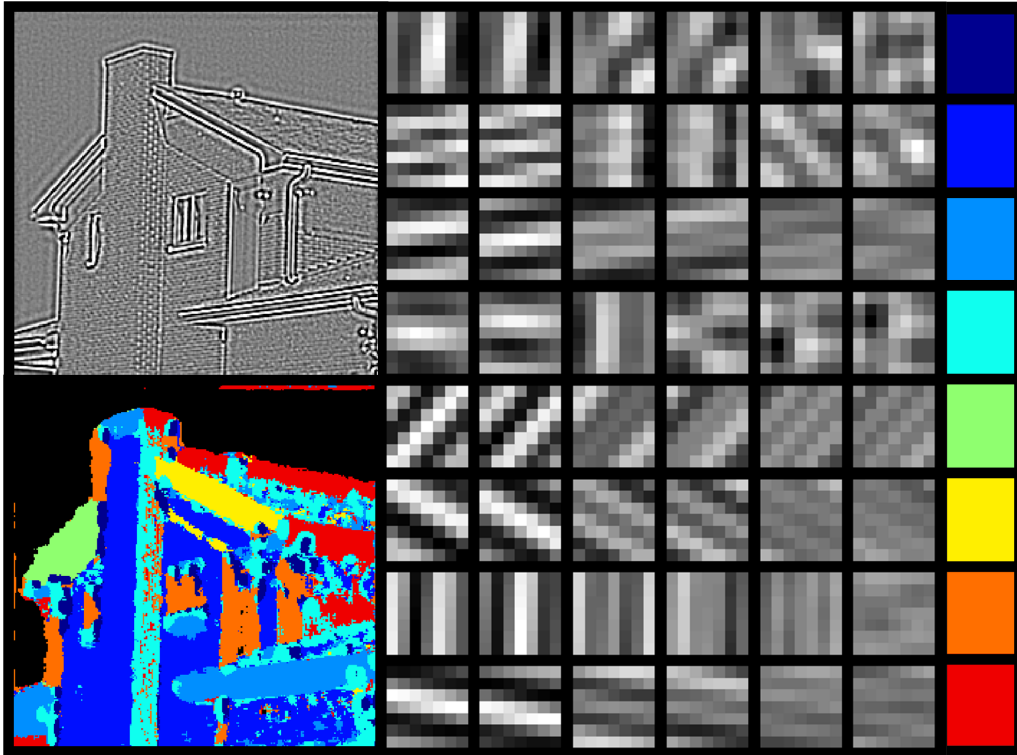
ajustamos nuestro modelo MGSM con  $K = 2, 4, 6, 8$ , mostrando los resultados de clasificar las muestras eligiendo la componente  $k$  que maximiza la probabilidad a posteriori  $P(k|\mathbf{x}_m)$  proporcionada por el ajuste MGSM. Como podemos observar, el modelo discrimina de forma sistemática las orientaciones, ignorando, en este caso, las diferentes frecuencias espaciales disponibles en la subbanda. Esto indica que el número de mezclas a utilizar puede estar en relación con el número de orientaciones dominantes en una imagen. En la Figura 4.3 se muestra una imagen sintética con un limitado número de orientaciones marcadas (2: horizontal y vertical) y que presenta características simples como esquinas, bordes y barras. La imagen utilizada está formada por un cuadrado y dos rectas que lo cruzan. Al igual que en la Figura 4.2, se incluyen una subbanda de alta resolución no orientada y los resultados de clasificar las muestras de acuerdo a  $P(k|\mathbf{x}_m)$ , previo ajuste del modelo MGSM con  $K = 2, 4, 7, 9$ . En este caso, para una mejor visualización de los resultados, las muestras asociadas a regiones de baja energía son descartadas (color negro en la imagen). El análisis de estos resultados indica



**Figura 4.3.** Comportamiento del modelo al aumentar el número de componentes de la mezcla ( $K$ ) usando vecindades  $7 \times 7$ . En la esquina superior izquierda, la imagen original. Debajo, una subbanda de alta resolución de la representación *Steerable pyramid* [6], sin orientaciones. El cuadro  $2 \times 2$  restante, de izquierda a derecha y de arriba a abajo, representan la clasificación de acuerdo a la probabilidad a posteriori  $P(k|\mathbf{x}_m)$  que obtiene el modelo cuando se utilizan  $K = 2, 4, 7$  y  $9$  componentes GSM en el ajuste del modelo.

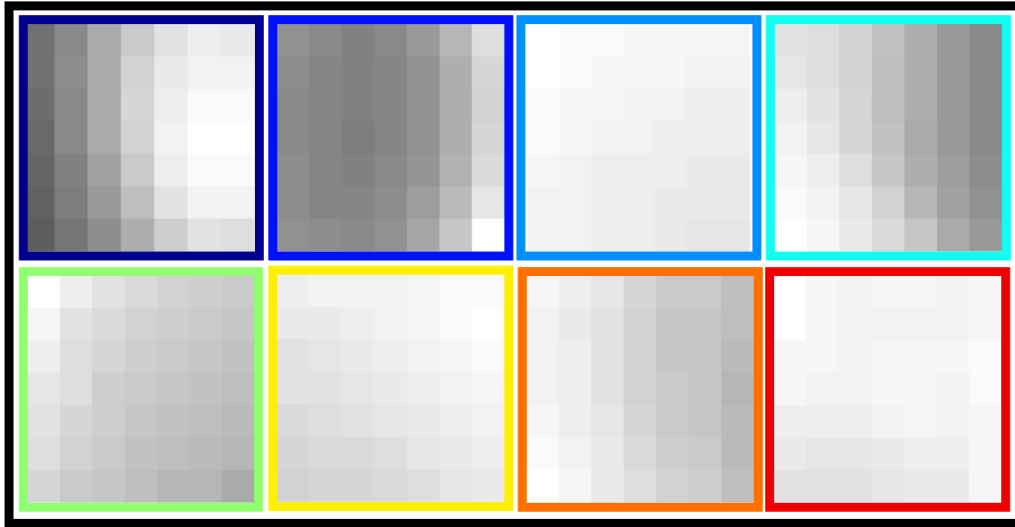
que, de forma sistemática, el modelo no es capaz de distinguir entre bordes y barras (características de fase), sea cual sea el número de componentes. Se observa también que, a medida que se aumenta el número de GSMs, el ajuste comienza a capturar características localizadas, como las esquinas.

Los ejemplos mostrados hasta ahora, aunque arrojan resultados interesantes, no dejan de ser poco realistas, por ser sintéticos. En la Figura 4.4 hemos utilizado la imagen de *House* para examinar el comportamiento del método con una imagen natural. En la misma, se muestra una subbanda de alta resolución así como el resultado de clasificar las muestras tras realizar un ajuste MGSM con  $K = 8$ . Junto a estas imágenes se muestran, por filas, los 6 primeros autovectores (pesados por la raíz cuadrada de sus autovalores asociados) de cada una de las  $K$  covarianzas representantes de la mezcla. El código de colores permite identificar las muestras donde la componente asociada domina sobre las otras. Las covarianzas que caracterizan orientaciones marcadas, corresponden a las filas 3<sup>a</sup> (tercer tono de azul), 5<sup>a</sup> (amarillo), 6<sup>a</sup> (verde), 7<sup>a</sup> (naranja) y 8<sup>a</sup> (rojo). Estas



**Figura 4.4.** Comportamiento del modelo usando la imagen de *House*. En la parte izquierda se muestra una subbanda de alta resolución (*Steerable pyramid*) no orientada. Debajo, se presenta la clasificación que realiza el modelo MGSM cuando se usan  $K = 8$  GSMs en la mezcla. Las vecindades con baja respuesta en energía han sido eliminadas (región en negro) para una mejor visualización. A la derecha de estas imágenes se muestran, por filas, los 6 primeros autovectores de cada componente, escalados por la raíz cuadrada de su autovalor correspondiente.

matrices de covarianza poseen todas una característica común: las dos primeras parejas de autovectores (que capturan prácticamente el 75 % de la energía) están aproximadamente en cuadratura de fase, siendo los componentes de cada pareja versiones desplazadas de una misma característica. La distribución de autovectores en cuadratura proporciona una explicación a la insensibilidad del modelo (basado, a fin de cuentas, en correlación) para discriminar bordes y barras. Este comportamiento, sin embargo, no está tan marcado en el resto de componentes, llamando especialmente la atención la asombrosa adaptación a la textura de la fachada mostrada por la componente asociada a la segunda fila. La Figura 4.5 pretende mostrar la dispersión espacial de energía presente en las vecindades para cada una de las covarianzas de la mezcla. Para ello se muestra la energía



**Figura 4.5.** Dispersión de energía promedio en la vecindad  $7 \times 7$  utilizada en el ejemplo de la Figura 4.4. Se presenta la suma de los autovectores al cuadrado ponderada por sus correspondientes autovalores. El código de colores sigue el mismo criterio que en la citada figura. Los valores oscuros representan baja energía y los claros alta.

promedio en cada posición de la vecindad<sup>8</sup>, resultado de sumar los autovectores al cuadrado ponderados por sus respectivos autovalores. Dado que el conjunto de muestras está formado por todas las posibles vecindades de la subbanda, en dicho conjunto están presentes todas las posibles versiones desplazadas de una misma característica, con lo que es de esperar un comportamiento invariante a traslación en el que no existan posiciones espaciales privilegiadas dentro de las vecindades. Esta hipótesis se cumple de forma razonable para las covarianzas asociadas a la 2<sup>a</sup>, 3<sup>a</sup>, 4<sup>a</sup>, 5<sup>a</sup>, 6<sup>a</sup>, 7<sup>a</sup> y 8<sup>a</sup> fila, para las cuales se muestra una dispersión de energía bastante uniforme. La única que parece favorecer de forma clara algunas posiciones espaciales es la covarianza de la primera fila, que como podemos ver en la imagen, se corresponde con esquinas muy localizadas. Una posible explicación a este comportamiento puede encontrarse en la variabilidad espectral que introduce este tipo de características. Según la posición espacial de la vecindad, cambia la proporción de contenido en una u otra dirección de los dos bordes convergentes, por lo que cambia el contenido espectral promedio en la vecindad, efecto que no es tan marcado en el caso de orientaciones simples.

<sup>8</sup>Un nivel claro representa alta energía, un nivel oscuro, baja.



**Figura 4.6.** Resultados visuales para los algoritmos de eliminación de ruido basados en GSM. A la izquierda la imagen de *House* y a la derecha la imagen de *Barbara*. Para una mejor visualización, se muestran regiones de  $128 \times 128$  píxeles comenzando en el píxel (24, 64) y (288, 64) respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo: Observación con ruido simulado (*House* con  $\sigma_w = 10$ , *Barbara* con  $\sigma_w = 25$ ); Resultado de GSM original [10]; Resultado de SVGSM [16]; Resultado de MGS. Los valores PSNR pueden consultarse en la Tabla 4.1

## 4.6.2. Eliminación de ruido

### 4.6.2.1. Comparación de métodos basados en GSM

En la Tabla 4.1 se muestran los resultados del algoritmo de eliminación de ruido propuesto, en términos de relación señal ruido pico (PSNR) en dB, utilizando  $K = 10$  GSMs en la mezcla. Comparamos el nuevo método (filas etiquetadas con MGS) con el método GSM [10] (filas etiquetadas con GSM) y la versión espacialmente variante que presentamos en la Sección 3.4 (filas etiquetadas con SVGSM). En todos los casos la representación utilizada ha sido TIHP. Como se puede observar, la mejora es significativa y generalizada, para todas las imágenes y todos los niveles de ruido, sin excepción (los mejores resultados se resaltan en negrita). La mejora promedio sobre GSM original es de aproximadamente 0.7 dB y de 0.4 dB sobre SVGSM. A diferencia del método basado en SVGSM, donde el tamaño de las regiones estaba en función de los niveles de ruido, el algoritmo descrito muestra un buen comportamiento para todos los niveles analizados, sin necesidad de variar ningún parámetro. La Figura 4.6 muestra una comparación visual con las imágenes de *House* y *Barbara*<sup>9</sup>

<sup>9</sup>Se muestra ampliada una región de  $128 \times 128$  para favorecer la visibilidad de los artefactos.

$\sigma_w$ / PSNR	Method	Lena	Barbara	Boat	House	Peppers
1 / 48.13	GSM	47.33	46.83	46.87	47.57	46.49
	SVGSM	47.45	46.99	46.91	47.67	46.48
	MGSM	<b>48.12</b>	<b>47.69</b>	<b>47.46</b>	<b>48.37</b>	<b>47.31</b>
5 / 34.15	GSM	38.19	37.19	36.73	38.23	37.57
	SVGSM	38.34	37.84	36.87	38.55	37.72
	MGSM	<b>38.64</b>	<b>38.02</b>	<b>37.30</b>	<b>39.30</b>	<b>37.75</b>
10 / 28.13	GSM	35.23	33.13	33.48	35.32	34.06
	SVGSM	35.38	34.13	33.56	35.56	34.24
	MGSM	<b>35.70</b>	<b>34.45</b>	<b>33.78</b>	<b>36.06</b>	<b>34.30</b>
15 / 24.61	GSM	33.50	30.76	31.63	33.73	32.02
	SVGSM	33.65	31.85	31.69	33.94	32.18
	MGSM	<b>34.03</b>	<b>32.38</b>	<b>31.91</b>	<b>34.30</b>	<b>32.28</b>
20 / 22.11	GSM	32.25	29.08	30.31	32.54	30.57
	SVGSM	32.40	30.19	30.37	32.71	30.67
	MGSM	<b>32.81</b>	<b>30.90</b>	<b>30.61</b>	<b>33.08</b>	<b>30.84</b>
25 / 20.17	GSM	31.26	27.81	29.27	31.59	29.45
	SVGSM	31.40	28.81	29.31	31.71	29.50
	MGSM	<b>31.85</b>	<b>29.80</b>	<b>29.62</b>	<b>32.12</b>	<b>29.74</b>
50 / 14.15	GSM	28.21	24.33	26.15	28.36	26.11
	SVGSM	28.21	25.03	26.15	28.38	26.11
	MGSM	<b>28.81</b>	<b>26.19</b>	<b>26.55</b>	<b>28.89</b>	<b>26.37</b>
75 / 10.63	GSM	26.48	22.96	24.51	26.42	24.07
	SVGSM	26.48	23.36	24.51	26.42	24.07
	MGSM	<b>27.05</b>	<b>24.11</b>	<b>24.94</b>	<b>26.86</b>	<b>24.43</b>
100 / 8.13	GSM	25.34	22.18	23.47	25.15	22.61
	SVGSM	25.34	22.38	23.47	25.15	22.61
	MGSM	<b>25.88</b>	<b>22.79</b>	<b>23.91</b>	<b>25.53</b>	<b>22.99</b>

**Tabla 4.1.** Rendimiento del método de eliminación de ruido en términos de relación señal ruido pico, en dB (PSNR). La primera columna muestra la desviación estándar del nivel de ruido, así como la PSNR asociada. En cada fila se muestran resultados para diferentes métodos BLS basados en GSM, todos ellos para la representación TIHP: GSM [10], espacialmente variante (SVGSM) y el nuevo método presentado (MGSM).

para los tres métodos GSM analizados en este apartado. Las imágenes han sido degradadas con ruido blanco Gaussiano de desviación estándar  $\sigma_w = 10$  y  $\sigma_w = 25$  respectivamente. Podemos observar una mejora progresiva y significativa en ambos ejemplos, destacando cómo los resultados de MGSM muestran una mejor recuperación de la textura (fachada de *House*), proporcionan un aspecto

más natural de los bordes y no están tan afectados por artefactos tipo ondícula (aparición de funciones base aisladas).

#### 4.6.2.2. Comparación con el estado del arte

En la Tabla 4.2 se muestra una comparativa, en términos de relación señal ruido pico (PSNR, en dB), entre los métodos que representan, a nuestro leal conocer, el estado del arte actual y los obtenidos por el algoritmo presentado en este Capítulo. Los métodos seleccionados coinciden con los comparados en el Capítulo anterior: SA-DCT [53] y KSVD [54] basados en representaciones adaptativas, la propuesta de 3DBM [52] que explota la redundancia no local en las imágenes y el modelo reciente basado en campos de GSMs, FoGSM [56]. Como podemos observar, el método 3DBM [52] presenta los mejores resultados de forma generalizada, seguido por FoGSM [56] y nuestra propuesta. Cabe destacar que, a medida que se incrementa el nivel de ruido, la diferencia entre 3DBM [52] y MGSM se ve reducida, llegando prácticamente a desaparecer en los niveles más altos.

En la Figura 4.7 mostramos una comparación visual entre los mismos métodos [53, 58, 54, 56] utilizando de nuevo las imágenes de *Barbara* y *House*. En comparación con SA-DCT [53], nuestro resultado presenta una mejor "recuperación visual" de la información, hecho que se puede apreciar en la textura de la pared en la imagen de *House* y en las barras del mantel de la figura de *Barbara*. También es destacable la menor presencia de artefactos en nuestro resultado, así como la existencia de bordes más definidos (ver por ejemplo la línea del tejado en *House*). En relación al resto de los métodos, nuestros resultados son similares a los presentados en [54] y visualmente competitivos con los que representan el estado del arte [52, 56].

En la Figura 4.8, hemos incluido una comparativa gráfica en términos de incremento de PSNR vs. PSNR de entrada, en dB, tomando como referencia el método presentado (MGSM). Se muestran las curvas de rendimiento de los métodos comparados en esta Sección para las imágenes de *House*, *Barbara*, *Peppers* y *Boats*, que nos permiten visualizar gran parte de los resultados incluidos en las Tablas 4.1 y 4.2. Por último, la Figura 4.9 resume las gráficas de la Figura anterior, presentando el rendimiento promedio<sup>10</sup> para las 4 imágenes utilizadas, aunque esta vez tomando como referencia el método GSM. Como podemos observar, la gráfica confirma lo que hasta ahora hemos discutido: el método MGSM representa una mejora sustancial y generalizada de nuestras propuestas previas basadas en GSM, siendo solo marginalmente inferior al estado del arte actual [52, 56].

<sup>10</sup>Para cada curva, se muestra el incremento PSNR resultante de promediar el error cuadrático medio (MSE) obtenido en cada una de las cuatro imágenes.

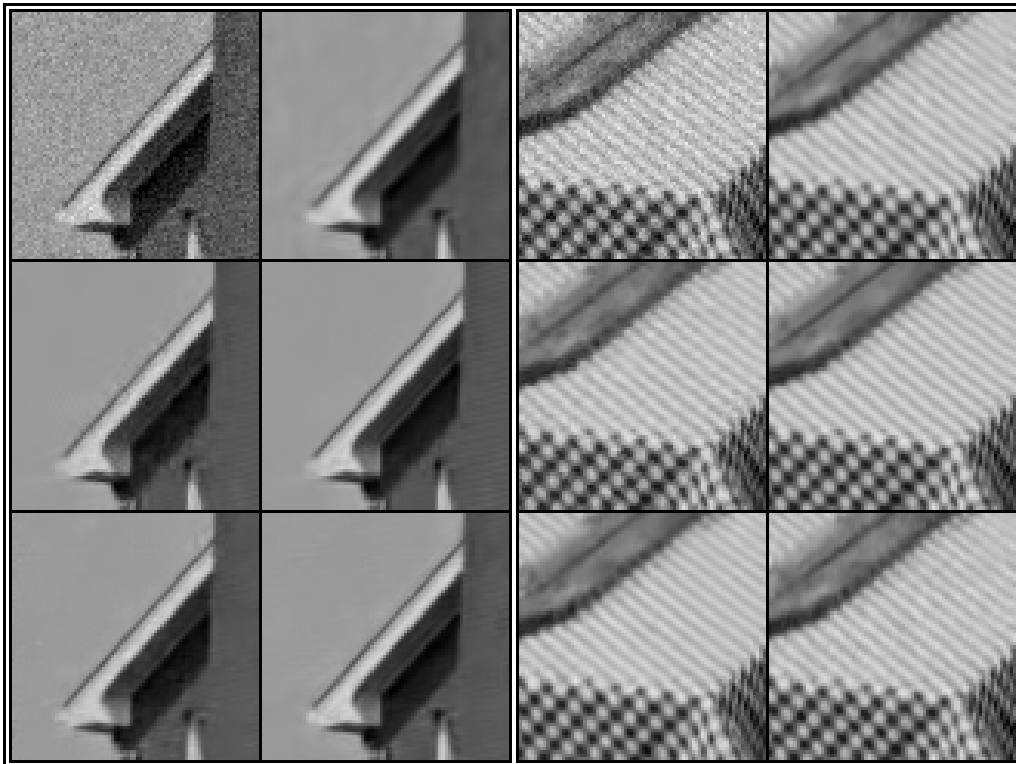
$\sigma_w$ / PSNR	Method	Lena	Barbara	Boat	House	Peppers
1/ 48.13	[52]	48.66	48.70	48.49	<b>49.14</b>	48.60
	[56]	<b>48.93</b>	<b>48.75</b>	<b>48.59</b>	48.97	<b>49.03</b>
	[53]	48.63	48.64	48.45	49.06	48.57
	[54]	48.56	48.67	48.43	<b>49.14</b>	48.46
	<i>MGS</i> M	48.12	47.69	47.46	48.37	47.31
5/ 34.15	[52]	<b>38.72</b>	38.31	37.28	<b>39.83</b>	<b>38.12</b>
	[56]	38.66	<b>38.65</b>	<b>37.39</b>	38.98	37.91
	[53]	38.54	37.47	37.14	39.38	37.99
	[54]	38.60	38.08	37.22	39.37	37.78
	<i>MGS</i> M	38.64	38.02	37.30	39.30	37.75
10/ 28.13	[52]	35.93	34.98	33.92	<b>36.71</b>	<b>34.68</b>
	[56]	<b>35.94</b>	<b>35.01</b>	<b>34.12</b>	35.63	34.38
	[53]	35.58	33.48	33.63	35.98	34.46
	[54]	35.47	34.42	33.64	35.98	34.28
	<i>MGS</i> M	35.70	34.45	33.78	36.06	34.30
15/ 24.61	[52]	34.27	<b>33.11</b>	32.14	<b>34.94</b>	<b>32.70</b>
	[56]	<b>34.28</b>	32.85	<b>32.31</b>	33.89	32.34
	[53]	33.86	31.37	31.79	34.14	32.44
	[54]	33.70	32.37	31.73	34.32	32.22
	<i>MGS</i> M	34.03	32.38	31.91	34.30	32.28
20/ 22.11	[52]	33.05	<b>31.78</b>	30.88	<b>33.77</b>	<b>31.29</b>
	[56]	<b>33.07</b>	31.31	<b>31.02</b>	32.64	30.89
	[53]	32.62	29.99	30.49	32.92	31.04
	[54]	32.38	30.83	30.36	33.20	30.82
	<i>MGS</i> M	32.81	30.90	30.61	33.08	30.84
25/ 20.17	[52]	32.08	<b>30.72</b>	29.91	<b>32.86</b>	<b>30.16</b>
	[56]	<b>32.11</b>	30.10	<b>30.03</b>	31.64	29.78
	[53]	31.66	28.94	29.47	31.92	29.92
	[54]	31.32	29.60	29.28	32.15	29.73
	<i>MGS</i> M	31.85	29.80	29.62	32.12	29.74
50/ 14.15	[52]	28.86	<b>27.17</b>	26.64	<b>29.37</b>	26.41
	[56]	<b>29.12</b>	26.40	<b>27.01</b>	28.51	26.43
	[53]	28.60	25.44	26.26	28.67	<b>26.55</b>
	[54]	27.79	25.47	25.95	27.95	26.13
	<i>MGS</i> M	28.81	26.19	26.55	28.89	26.37
75/ 10.63	[52]	27.02	<b>25.10</b>	24.96	<b>27.20</b>	24.48
	[56]	<b>27.37</b>	24.29	<b>25.33</b>	26.69	24.53
	[53]	26.76	23.51	24.62	26.58	<b>24.57</b>
	[54]	25.80	23.01	23.98	25.22	23.69
	<i>MGS</i> M	27.05	24.11	24.94	26.86	24.43
100/ 8.13	[52]	25.57	<b>23.49</b>	23.74	25.50	22.91
	[56]	<b>26.12</b>	23.01	<b>24.20</b>	25.33	<b>23.17</b>
	[53]	25.50	22.49	23.52	25.07	23.15
	[54]	24.46	21.89	22.81	23.71	21.75
	<i>MGS</i> M	25.88	22.79	23.91	<b>25.53</b>	22.99

**Tabla 4.2.** Rendimiento del método de eliminación de ruido en términos de relación señal ruido pico, en dB (PSNR). La primera columna muestra la desviación estándar del nivel de ruido, así como la PSNR asociada. En cada fila se muestran resultados para diferentes métodos que representan el estado del arte, así como los resultados obtenidos con *MGS*M.

## 4.7. Conclusiones

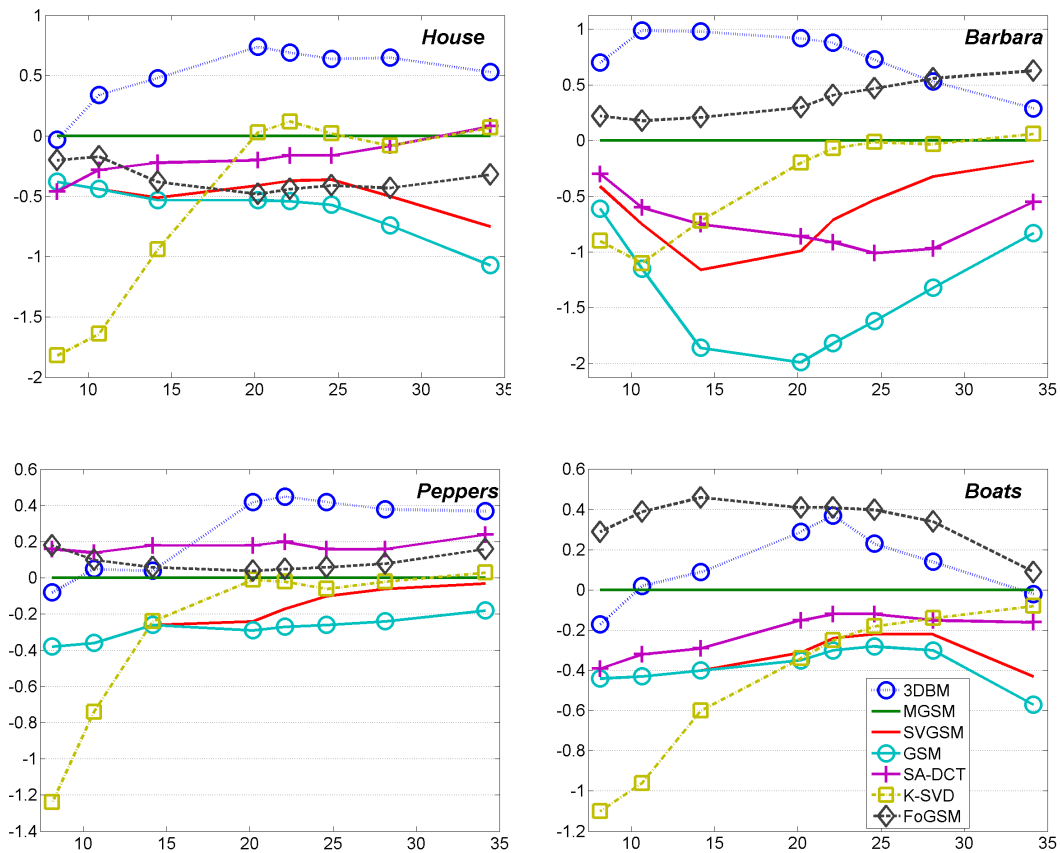
En este Capítulo hemos presentado un modelo de imagen en el dominio ondicular sobre-completo (pirámides orientadas sobre-completas) basado en una





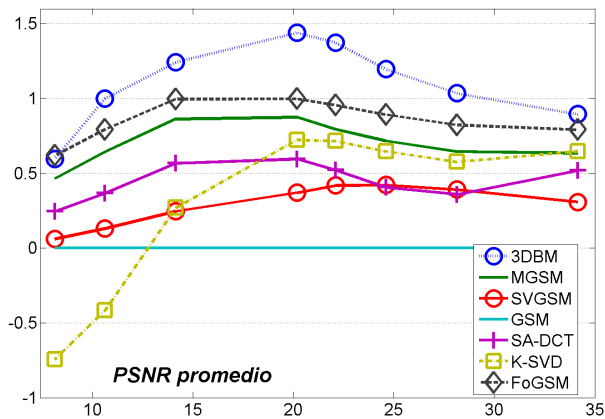
**Figura 4.7.** Comparativa de resultados visuales, sobre las imágenes de *House* y *Barbara*, entre métodos del estado del arte y los resultados obtenidos con MGSM. Para una mejor visualización de los detalles se muestran regiones de  $80 \times 80$  píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo se muestra (*House/Barbara* PSNR en dB): observaciones ruidosas utilizando  $\sigma_w = 10$  (28.13/28.13); resultados de FoGSM [56] (35.63/35.01); resultado de [53] (35.98/33.48); resultados de [52] (36.71/34.98); resultados de [54] (35.94/34.42); nuestros resultados con MGSM (36.06/34.45).

mezcla de mezclas en escala de Gaussianas (MGSM). El modelo es flexible y conceptualmente limpio, proporcionando un único modelo local para todas las vecindades de cada subbanda. Este hecho permite, además de mantener la potencia de los métodos adaptativos a la varianza, superar algunas de las limitaciones de propuestas previas basadas en GSM, proporcionando una mayor capacidad de adaptación y explotando la redundancia no local existente en las subbandas. Otro aspecto importante del modelo es que, a diferencia de GSM y SVGSM y gracias al algoritmo de ajuste iterativo de máxima verosimilitud, no necesita utilizar priors (prior de Jeffrey, por ejemplo), ya que el método presentado proporciona, para cada imagen, distribuciones adaptativas (de forma similar a [59]). El modelo MGSM ha demostrado ser efectivo para describir la estadística



**Figura 4.8.** Incremento PSNR vs. PSNR de entrada, en dB, tomando como referencia el método presentado, MGSM. Se muestran los resultados para las imágenes de *House*, *Barbara*, *Peppers* y *Boats*, todas ellas procesadas usando TIHP. Comparamos con cuatro métodos del estado del arte y nuestros métodos previos basados en GSM: línea punteada con círculos (azul) para 3DBM [52]; línea continua con cruces (lila) para SA-DCT [53]; línea discontinua con cuadrados (amarilla) para KSVD [54]; línea continua con círculos (turquesa) para GSM [10]; línea continua (roja) para SVGSM [16]; línea discontinua con rombos (gris) para FoGSM [56]; línea continua (verde) para el método MGSM.

local de las imágenes en el dominio ondicular, apoyándose en descriptores globales que aprovechan las típicas formas de redundancia existentes en las imágenes naturales. Prueba de ello son los resultados obtenidos por el método de eliminación de ruido MGSM, que además de suponer una mejora significativa y generalizada de sus antecesores (GSM y SVGSM), se comporta de forma competitiva con los que representan el estado del arte actual.



**Figura 4.9.** Comparativa de rendimiento promedio para las imágenes de *House*, *Barbara*, *Peppers* y *Boats* tomando como referencia el método GSM. Las curvas muestran el incremento PSNR vs. PSNR de entrada para cuatro métodos del estado del arte y nuestros métodos previos basados en GSM: línea punteada con círculos (azul) para 3DBM [52]; línea continua con cruces (lila) para SA-DCT [53]; línea discontinua con cuadrados (amarilla) para KSVD [54]; línea continua (turquesa) para GSM [10]; línea continua con círculos (roja) para SVGSM [16]; línea discontinua con rombos (gris) para FoGSM [56]; línea continua (verde) para el método MGSM.

## Capítulo 5

# Deconvolución basada en métodos de eliminación de ruido

### 5.1. Introducción

La restauración de imágenes es un problema clásico que trata de estimar una imagen, dada una versión filtrada y ruidosa de la misma. En el mundo real existen multitud de situaciones que podrían modelarse con un filtrado lineal y adición de ruido Gaussiano, como por ejemplo la adquisición de imágenes con un dispositivo óptico imperfecto o desenfocado y sensor ruidoso. Obtener una estimación en estas condiciones requiere atacar un problema inverso mal condicionado, difícil de resolver incluso si los parámetros de degradación se asumen, como en esta Tesis, conocidos. La estimación Bayesiana, haciendo uso del conocimiento a priori sobre las imágenes naturales, es una herramienta muy potente para la resolución de este problema. Normalmente, los métodos de restauración son globales e iterativos [60, 61, 62]. Algunos de los métodos globales [63, 64, 65, 66, 67] han seguido una estrategia de resolución basada

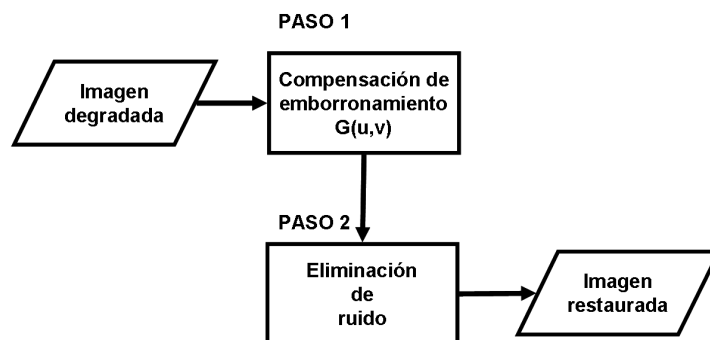


Figura 5.1: Esquema de dos pasos.

en un esquema de dos pasos (ver la Figura. 5.1): (a) compensación global de emborronamiento, y (b) eliminación de ruido. La principal ventaja de este esquema es que el problema de deconvolución se convierte así en un problema de menor complejidad: la eliminación de ruido Gaussiano no blanco<sup>1</sup>. El precio a pagar es la más que probable posibilidad de perder la optimalidad global. Dado el reciente auge de métodos de eliminación de ruido de gran rendimiento (ver Capítulos 3 y 4), este esquema representa una alternativa especialmente atractiva desde el punto de vista práctico. Otra aproximación mucho menos explorada, ha sido atacar el problema de la restauración utilizando modelos locales (ver por ejemplo [68, 19]). La descripción local de la señal simplifica la formulación y la resolución del problema, pero impone serias limitaciones: ¿cómo podemos compensar un emborronamiento causado por un filtro de gran soporte espacial si estamos utilizando una pequeña vecindad para la descripción local de la señal? Como veremos a lo largo de este Capítulo, un esquema multi-resolución reduce el impacto de este problema, pero no lo soluciona completamente. En la Sección 2.6.2 presentamos la versión de BLS-GSM para restauración [19]. A pesar de dar unos buenos resultados mediante una estimación no iterativa, esta aproximación presenta dos serias limitaciones. La primera está relacionada con las restricciones inherentes a los modelos locales y la segunda con la necesidad de tener una estimación precisa de la densidad espectral de potencia de la imagen original.

En este Capítulo proponemos un método de restauración [12, 16] que, basado en una estrategia de dos pasos, hace uso del método SVGSM (descrito en el Capítulo 3) como método de eliminación de ruido local. Esta aproximación preserva las ventajas de usar un modelo local para la eliminación de ruido (simple y no iterativo) a la vez que evita los problemas asociados a la compensación local de convoluciones. Mediante la elección de un adecuado filtro de deconvolución global, el método presenta un compromiso marginalmente óptimo entre la compensación de emborronamiento (global) y la eliminación (local) de ruido. Además, el método de deconvolución mejora, tanto en robustez como en rendimiento, a un método previo basado en BLS-GSM [19].

## 5.2. Motivación

En el Capítulo 2 presentamos el problema de la deconvolución en presencia de ruido, cuya resolución pretende obtener una estimación de la imagen original a partir de una observación filtrada linealmente y con ruido añadido. Es decir, nuestro modelo de observación es:

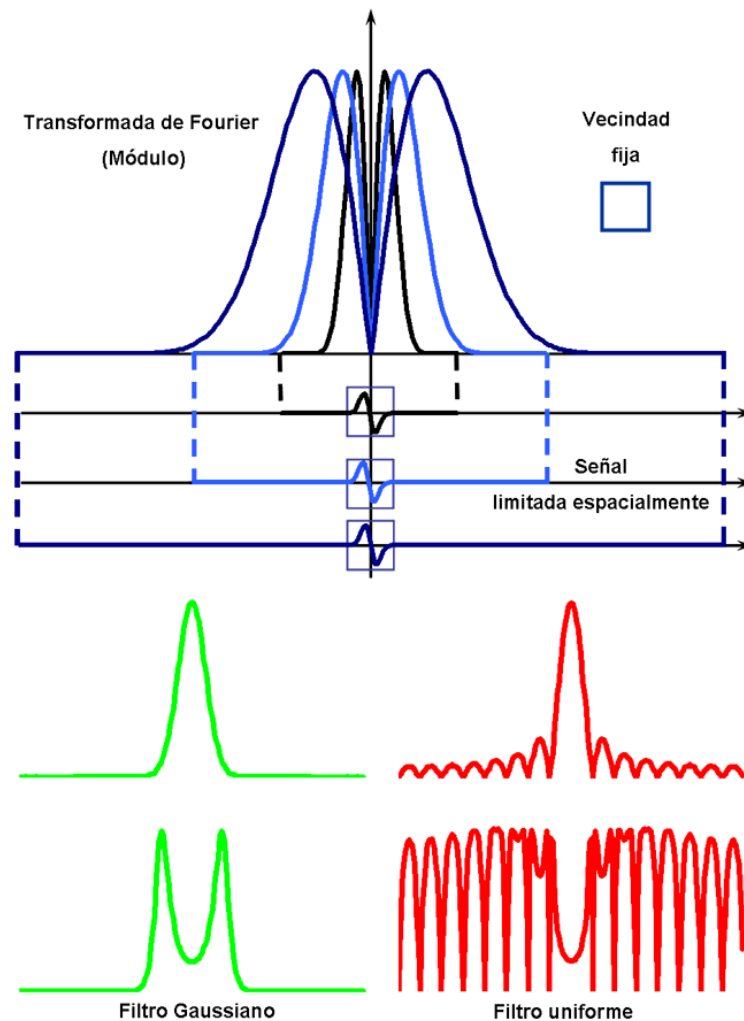
$$y_0(n, m) = h(n, m) * x(n, m) + w_0(n, m), \quad (5.1)$$

---

<sup>1</sup>Nótese que aun modelando el ruido inicialmente como blanco, este queda correlacionado tras la compensación global.

donde  $h$  es una función localizada (asumida conocida) que se convoluciona con la imagen original y  $w_0$  es ruido Gaussiano de densidad espectral de potencia arbitraria, pero conocida. En la Sección 2.6.2 vimos cómo el método clásico basado en GSM podía ser aplicado de forma análoga a la eliminación de ruido y emborronamiento [19]. Según esto, resulta natural pensar en una mejora de este algoritmo en la misma línea de SVGSM, esto es, realizar una deconvolución espacialmente adaptativa. Sin embargo, al reto técnico de implementar un algoritmo robusto, se añade en este caso un problema intrínseco que presenta el modelo GSM para la deconvolución: el trabajar con una representación local de la imagen no permite compensar adecuadamente determinados filtros de convolución. Analicemos el problema detenidamente. Considerando una vecindad pequeña en un modelo local, solo podemos aspirar a compensar filtros que carezcan de oscilaciones alrededor de cero en frecuencia y que varíen suavemente en dicho dominio. La razón es que, bajo esas condiciones, el filtro inverso regularizado asociado será también suave en frecuencia y, por tanto, tendrá un soporte espacial pequeño (representable con la vecindad local). Está claro, en consecuencia, que una vecindad fija limita fuertemente la capacidad para deconvolucionar ciertas degradaciones. Sin embargo, si el modelo local se combina con un esquema piramidal multi-resolución (como es el caso), el tener una vecindad fija en todas las escalas permite representar vecindades comparativamente mayores a medida que subimos en la pirámide. Ilustramos este comportamiento en la Figura 5.2. Arriba mostramos la transformada de Fourier (magnitud) de una señal limitada espacialmente en distintas escalas. El tamaño de la vecindad de coeficientes es el mismo para todas las escalas. A medida que subimos la escala, el soporte frecuencial se hace menor y se obtienen respuestas que varían más rápidamente. De esta forma, gracias a la multi-resolución podemos compensar filtros (sin oscilaciones alrededor de cero, en frecuencia) que sean suaves en las frecuencias altas, menos suaves en las medias y que varíen rápidamente en las bajas. Estas características son compatibles con muchos filtros paso bajo, por ejemplo filtros Laplacianos o Gaussianos (ver la Figura 5.2, abajo a la izquierda). Sin embargo, existen filtros importantes en la práctica, como aquellos que integran de forma uniforme en un soporte espacial dado, cuyo comportamiento en las altas frecuencias está lejos de ser suave, presentando fuertes oscilaciones alrededor de cero. Para compensar estos ceros se necesitan respuestas abruptas en el filtro inverso regularizado (ver la Figura 5.2, abajo a la derecha), lo que implica necesariamente el uso de un soporte espacial grande, dificultando así su compensación con el método local.

Otra limitación importante de [19] es que, para un buen funcionamiento, necesita de una estimación precisa de la covarianza cruzada entre la señal original y la filtrada. Esto implica la necesidad de estimar el espectro de potencia de la imagen original ( $P_x$ ) a partir de la observación degradada, problema este de alta dificultad.



**Figura 5.2.** Ilustración de las restricciones a la forma de los filtros en frecuencia cuando se utiliza una aproximación local (vecindad pequeña de tamaño fijo) junto con un esquema multi-resolución. Se muestra la misma señal limitada espacialmente en tres escalas diferentes. El esquema permite representar y compensar, siempre que no haya cruces por cero en frecuencia, filtros con variaciones lentas en las frecuencias altas y con variaciones rápidas en las frecuencias bajas (ver texto para más detalles). En la parte inferior de la Figura se muestran dos filtros paso bajo típicos (Gaussiano y uniforme, ambos  $1D$ ) sobre sus correspondientes filtros inversos regularizados. El filtro Gaussiano puede ser compensado satisfactoriamente con esta aproximación, mientras el uniforme no.

El método presentado en este Capítulo persigue solventar estos problemas manteniendo la contrastada potencia de los métodos locales. Para ello utilizaremos el esquema de deconvolución en dos pasos (ver 5.1) y el método local de eliminación de ruido presentado en el Capítulo 3.

### 5.3. Deconvolución basada en SVGSM

Como avanzamos en la Sección anterior, la solución que proponemos se basa en un esquema de dos pasos presentado hace más de diez años [63] y que ha sido utilizado por muchos autores en la literatura [64, 65, 66, 67, 69]. Los pasos son: (a) compensación lineal de emborronamiento mediante un filtrado global, y (b) eliminación no lineal de ruido (en nuestro caso, de forma local) a la observación compensada.

Aplicando la transformada de Fourier a la Ecuación 5.1 podemos obtener el modelo de observación en el dominio de la frecuencia:

$$Y_0(u, v) = H(u, v)X(u, v) + W_0(u, v),$$

donde<sup>2</sup>  $H$  es la transformada de Fourier del kernel de convolución. Llamando  $G$  al filtro de compensación global del primer paso (en adelante, prefiltro), lo que se pretende es obtener una expresión del mismo que minimice el error cuadrático medio de nuestra estimación. Para ello el cálculo del filtro óptimo,  $G_{opt}$ , debe acoplar ambos pasos:

$$G_{opt} = \arg \min_G \{\mathbb{E}\{\|X - f(GY_0)\|^2\}\}, \quad (5.2)$$

donde  $f(GY_0)$  es el resultado de aplicar la función de eliminación de ruido del paso dos a la observación compensada por  $G$  en el primer paso. La esperanza dentro de la Ecuación 5.2 puede descomponerse como sigue:

$$\begin{aligned} & \mathbb{E}\{\|X - f(GY_0)\|^2\} \\ &= \mathbb{E}\{\|(X - GHX) + (GHX - f(GY_0))\|^2\} \\ &\simeq \mathbb{E}\{\|(X - GHX)\|^2\} + \mathbb{E}\{\|(GHX - f(GY_0))\|^2\}, \end{aligned}$$

donde la última expresión representa la suma de los errores cuadráticos cometidos en cada paso. La aproximación realizada implica asumir descorrelación de ambos errores, lo que es razonable en condiciones normales<sup>3</sup>. Así pues, para calcular el prefiltro óptimo necesitamos un modelo del error del algoritmo de eliminación de ruido utilizado en el segundo paso. El carácter adaptativo y no lineal del algoritmo que vamos a utilizar (SVGSM), supone un problema para el desarrollo de este modelo, que a nuestro leal conocer, no ha sido resuelto hasta ahora para ningún método de eliminación de ruido de rendimiento y complejidad comparables. En ausencia de una solución óptima global, presentamos aquí un procedimiento que, a pesar de ser solo marginalmente óptimo, proporciona una solución eficiente y robusta. A continuación describimos más en detalle los pasos del algoritmo.

<sup>2</sup>Para aligerar la notación, en lo sucesivo prescindimos de los subíndices  $(u, v)$ .

<sup>3</sup>Hemos comprobado empíricamente esta descorrelación utilizando simulaciones.



**Paso 1:** *Compensación global de emborronamiento*

Este paso proporciona una imagen compensada, resultado de aplicar un filtrado global a la observación. En lugar de estimar un filtro óptimo, reducimos los grados de libertad de  $I_h \times I_v$  (tamaño de la imagen y del filtro) a tan solo uno, que corresponde al parámetro de regularización  $\alpha$ . Como en otros trabajos [64, 65, 69] utilizamos un filtro de Wiener generalizado, que posee la siguiente expresión:

$$G = \frac{H^*}{|H|^2 + \alpha \cdot \left(\frac{P_{w_0}}{P_x}\right)}, \quad (5.3)$$

donde  $P_{w_0}$  y  $P_x$  son la densidad espectral de potencia de ruido y señal respectivamente. Esta inversión regularizada, al igual que el filtro de Wiener, compensa completamente la fase introducida por el kernel de convolución. El equilibrio entre la eliminación de ruido y deconvolución está controlado por el parámetro  $\alpha$ . Valores bajos de  $\alpha$  proporcionan resultados muy ruidosos ( $\alpha = 0$  es el filtro inverso<sup>4</sup>) mientras que valores altos atenúan excesivamente la señal. Cuanto más alto sea el valor de  $\alpha$ , mayor será la distorsión lineal (sesgo en la estimación), pero cuanto más pequeño, más ruidosa será la estimación (más varianza tendrá ésta). Un valor  $\alpha = 1$  proporciona el filtro de Wiener (ver Sección 2.3.1), que, como veremos más adelante, resulta ser un valor demasiado alto cuando usamos imágenes típicas y un algoritmo de eliminación de ruido no lineal.

**Paso 2:** *Eliminación de ruido usando SVGSM*

Tras el primer paso, el modelo de observación posee la siguiente expresión:

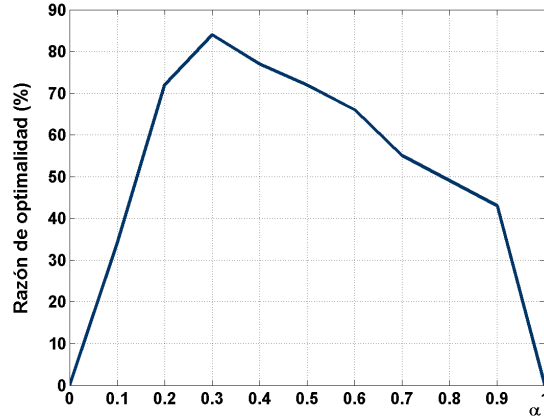
$$Y_1 = GY_0 = GHX + GW_0 = H_r X + W_1.$$

Como el filtro  $G$  ha sido optimizado en el primer paso, carece de sentido corregir la distorsión lineal residual  $H_r = GH$ , con lo que directamente aplicamos el método de eliminación de ruido (descrito en la Sección 3.4) a la observación compensada  $Y_1$ , considerando los estadísticos del ruido prefiltrado,  $\mathbf{w}_1$ .

**5.3.1. Estimación de parámetros del prefiltrado.**

Como dijimos al principio de este Capítulo, los parámetros de la degradación que ha sufrido la imagen (distorsión lineal y ruido) se asumen conocidos. Así pues, de acuerdo con el prefiltrado descrito en la Ecuación 5.3, los parámetros a estimar son la densidad espectral de potencia de la señal ( $P_x$ ) y el factor de regularización ( $\alpha$ ):

<sup>4</sup>Es importante destacar que para el buen comportamiento del filtro  $G$ , es crítico asegurar que  $P_{w_0}$  es estrictamente positiva en todo el plano frecuencial. Esto es especialmente importante cuando  $P_{w_0}$  es estimada a partir de imágenes observadas reales (no simulaciones).



**Figura 5.3.** Razón de optimalidad del parámetro  $\alpha$ . Ver texto para más detalles.

#### *Densidad espectral de potencia de la señal*

Basándonos en los espectros típicos de las imágenes naturales [30, 31], hemos utilizado un modelo extremadamente sencillo para  $P_x$ :

$$\hat{P}_x = k/f^2,$$

donde  $f$  representa frecuencia absoluta y  $k$  es una constante arbitraria. Para forzar al modelo a ser consistente con la varianza original estimada ( $\hat{\sigma}_x^2 = \sigma_{y_0}^2 - \sigma_{w_0}^2$ ), se debe cumplir la siguiente expresión:

$$\hat{\sigma}_x^2 = \int_{f>0} |H(f)|^2 \hat{P}_x f df = \int_{f>0} |H(f)|^2 \frac{k}{f} df,$$

con lo que, despejando  $k$ , obtenemos:

$$k = \hat{\sigma}_x^2 / \int_{f>0} |H(f)|^2 / f df.$$

A pesar de ser un modelo simple, es efectivo y, en contraste con otros métodos (como [19]), el error de estimación asociado a  $P_x$  no produce un impacto crítico en el rendimiento del método.

#### *Parámetro de regularización*

En contraste con otros autores [63, 65] que utilizan un modelo para elegir el parámetro de regularización  $\alpha$ , en esta Tesis hemos propuesto un cálculo empírico del mismo, que asegura un buen rendimiento en las condiciones estudiadas. Para ello definimos los siguientes conjuntos:

- Un conjunto de imágenes: Seleccionamos 5 imágenes test estándar, *House*, *Cameraman*, *Barbara*, *Boat* y *Lena*.

- Un conjunto de filtros: Filtros Gaussianos con desviación estándar de 0.4 a 3.2, en una escala geométrica.
- Un conjunto de niveles de ruido: Utilizaremos ruido blanco Gaussiano con desviación estándar de 1 a 16, en una escala geométrica.

La batería de experimentos está formada por todas las combinaciones de los conjuntos descritos, obteniendo así un total de  $5 \times 4 \times 5 = 100$ . Para cada experimento hemos calculado el valor óptimo de  $\alpha$  utilizando un algoritmo de búsqueda en línea (sección áurea). También hemos calculado un intervalo de tolerancia alrededor del óptimo tal que la disminución en rendimiento no supere los 0.05 dB. Calculando, para cada valor de  $\alpha$ , el número de experimentos que contienen dicho  $\alpha$  en su intervalo de tolerancia, obtenemos la razón de optimalidad. El valor elegido es aquel con la razón de optimalidad mayor, es decir, aquel valor de  $\alpha$  que resulta en una mayor proporción de experimentos con resultados cuasi-óptimos (a 0.05 dB o menos por debajo del resultado que se obtiene con el  $\alpha$  óptimo). En la Figura 5.3 podemos ver una gráfica de la razón de optimalidad obtenida. Como podemos observar hay un pico en  $\alpha = 0.3$ , que proporciona un 84% de experimentos cuasi-óptimos. El entrenamiento ha sido repetido para distorsiones lineales muy diferentes [12] (como el filtro uniforme) obteniendo también un  $\alpha$  cercano a 0.3, aunque con un porcentaje de optimalidad menor (73%). Por último hemos comprobado, comparando estos resultados con los obtenidos al excluir la imagen a procesar en cada caso, que los resultados son poco dependientes de las imágenes elegidas.

## 5.4. Implementación

El primer paso, es decir, el prefiltrado global, ha sido implementado en el dominio de Fourier, utilizando un valor de  $\alpha = 0.3$ . Para la eliminación de ruido (paso 2), hemos utilizado dos representaciones, TIHP y FSP: TIHP (con 3 orientaciones fijas) para imágenes simples y FSP (con 8 orientaciones) para imágenes altamente texturadas. En ambos casos hemos utilizado 4 escalas, independientemente del tamaño de la imagen. El tamaño de vecindad GSM es de  $3 \times 3$ , sin utilizar en este caso coeficiente *padre*. El tamaño de bloque elegido ha sido  $32 \times 32$ .

El tiempo de computación del algoritmo es, a efectos prácticos, el correspondiente al algoritmo usado en el segundo paso (ver Tabla 3.1 en la Sección 3.5.3), al ser el tiempo de computación del modelado de  $P_x$  (cálculo de la constante  $k$ ) y del prefiltrado global (paso 1) comparativamente despreciable.

Filtrado ( $\sigma_b$ )	Niveles de ruido ( $\sigma_w$ )				
	1	2	4	8	16
0.4	2.86	1.86	2.81	5.03	7.63
0.8	6.65	4.84	3.73	3.92	5.85
1.6	3.87	3.18	2.72	3.01	4.73
3.2	3.21	2.85	2.56	2.66	3.87

**Tabla 5.1.** Resultados promedio del conjunto de entrenamiento (ver Sección 5.3.1) en términos de incremento de relación señal ruido (ISNR) en dB.

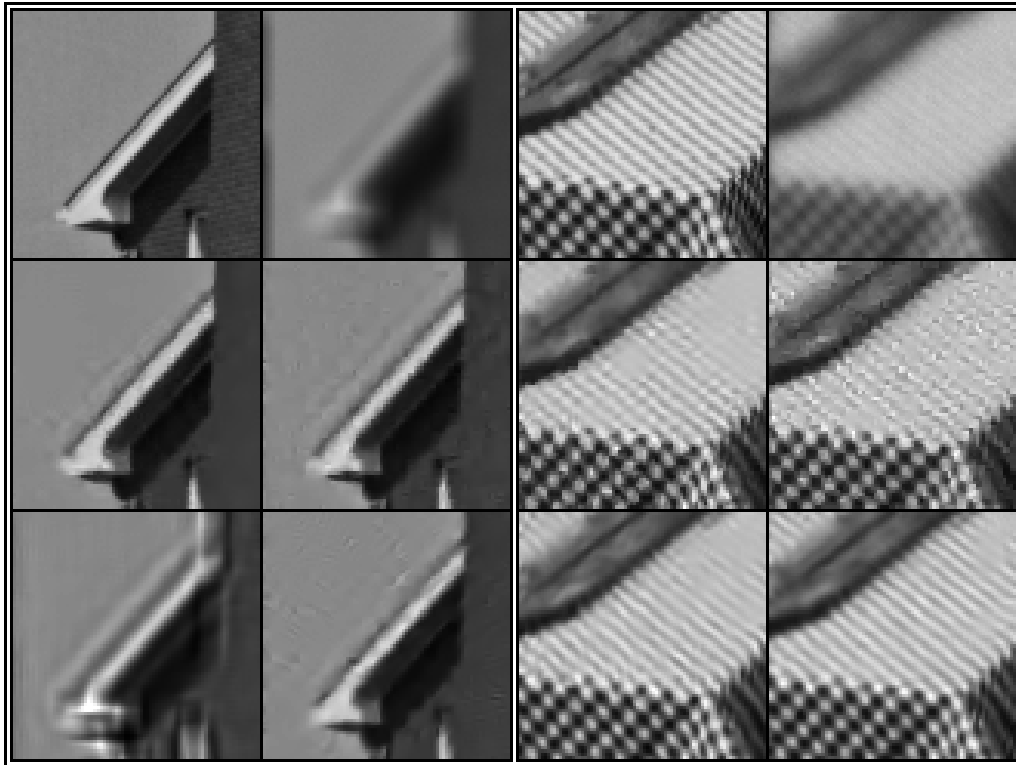
## 5.5. Resultados y discusión

En la Tabla 5.1 mostramos los resultados promediados, en términos de incremento de relación señal ruido (ISNR, del inglés, *Increment of Signal to Noise Ratio*), obtenidos al aplicar nuestro método sobre el conjunto de entrenamiento descrito en la Sección 5.3.1. Como podemos observar, los resultados muestran una mejora significativa respecto a la observación para todo el rango de degradaciones analizado. En la Tabla 5.2 mostramos los resultados para seis tipos de experimentos diferentes usando las imágenes de *House*, *Cameraman* (ambas usando TIHP) y *Barbara* (usando FSP). Se muestra una comparación de nuestros resultados con la versión BLS-GSM para deconvolución [19] y los métodos que, a nuestro leal conocer, representan el estado del arte actual [69, 61]. Las dos primeras columnas corresponden a experimentos replicados de [67]. El filtro usado (PSF1) es  $h_{i,j} = (1 + i^2 + j^2)^{-1}$ , para  $i, j = -7 \dots 7$ . La siguiente columna reproduce los experimentos realizados en [63, 65], en el que se usa un filtro uniforme de soporte  $9 \times 9$  (PSF2). En la cuarta columna, y al igual que en [71], se usa un filtro de soporte  $5 \times 5$  separable (PSF3) con los coeficientes correspondientes al producto externo de  $[1, 4, 6, 4, 1]/16$  consigo mismo, en las direcciones vertical y horizontal. Las últimas dos columnas representan filtros Gaussianos con desviaciones estándar de 1.6 y 0.4 (PSF4 y PSF5 respectivamente). Los niveles de ruido (desviación estándar) se muestran en la segunda fila de la Tabla. Los resultados arrojan que [69] proporciona los mejores resultados en un 44.4% de los experimentos, los métodos [61]+[70] lo son en un 16.7% y nuestra aportación en un 38.9%. Comparando con la versión BLS-GSM [19], el nuevo algoritmo proporciona una mejora muy significativa para todos los experimentos, especialmente en aquellos que usan el filtro uniforme (PSF2), donde el modelo local fracasa, confirmando los problemas para compensar cierto tipo de filtros que analizamos en la Sección 5.2. En una comparación con [69], es importante destacar que, en general, nuestro método proporciona mejores resultados con la imagen de *Barbara*, y peores con *House* y *Cameraman*. Siendo [69] un método basado en un esquema en dos pasos, con un primer paso muy similar al que proponemos, la diferencia de resultados estriba en el método de eliminación de ruido (basado en la transformación SA-DCT). Este método es muy adecuado para preservar bordes nítidos y contornos

Filtro	<i>PSF 1</i>		<i>PSF 2</i>	<i>PSF 3</i>	<i>PSF 4</i>	<i>PSF 5</i>
$\sigma_w \Rightarrow$	$\sqrt{2}$	$\sqrt{8}$	$\sqrt{0.308}$	7	2	8
	<i>HOUSE</i>					
PSNR	25.62	25.47	24.11	28.10	27.83	30.01
[61]	8.47	6.63	<b>10.71</b>	4.22	<b>4.49</b>	4.76
[19]	8.46	6.93	-0.44	4.37	4.34	5.98
[69]	<b>9.05</b>	<b>7.64</b>	<b>10.71</b>	<b>5.10</b>	4.03	<b>6.13</b>
[16]	8.64	7.03	9.04	4.30	4.11	6.02
	<i>CAMERAMAN</i>					
PSNR	22.23	22.16	20.77	24.63	23.36	29.83
[61]	7.46	5.24	8.16	2.84	3.18	3.65
[70]	6.93	4.88	7.59	<b>2.94</b>	-/-	-/-
[62]	7.40	5.15	8.10	2.85	-/-	-/-
[19]	6.84	5.29	-1.61	2.56	2.83	3.81
[69]	<b>8.25</b>	<b>6.34</b>	<b>8.57</b>	2.56	3.05	<b>5.15</b>
[16]	7.45	5.55	7.33	2.73	<b>3.25</b>	4.19
	<i>BARBARA</i>					
PSNR	23.33	23.26	22.49	24.21	23.77	29.77
[61]	3.76	1.99	3.98	0.9	0.92	2.55
[19]	5.70	3.28	-0.27	1.44	0.95	4.91
[69]	5.73	3.01	4.88	1.58	0.91	4.04
[16]	<b>6.85</b>	<b>3.80</b>	<b>5.07</b>	<b>1.94</b>	<b>1.36</b>	<b>5.27</b>

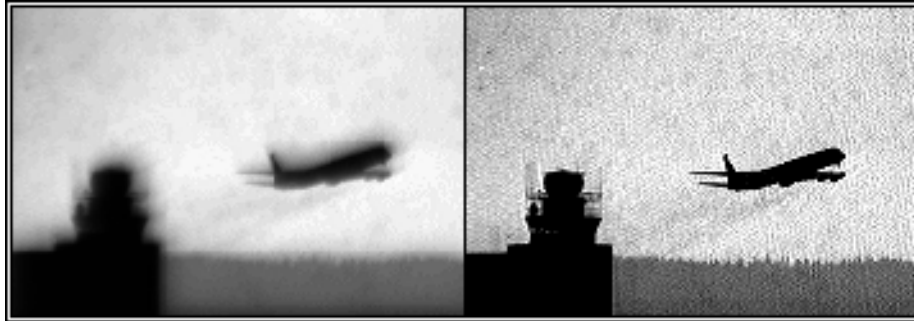
**Tabla 5.2.** Comparación de nuestra aportación [16] con otros métodos de la literatura, en términos de incremento de relación señal ruido (ISNR) en dB. La primera fila muestra el filtro de convolución utilizado en la degradación (ver texto para más detalles) y la segunda muestra las desviaciones estándar del ruido. Los mejores resultados en cada caso se resaltan en negrita.

(muy presentes en *House* y *Cameraman*), mientras que no es tan potente como SVGSM para caracterizar el espectro local. Por ello su rendimiento decrece cuando procesa imágenes muy ricas en texturas (como *Barbara*). En la Figura 5.4 mostramos una comparación visual entre los métodos del estado del arte [61, 69], BLS-GSM [19] (deconvolución) y nuestra aportación. Para la imagen de *Barbara*, la degradación utilizada corresponde al filtro de convolución PSF1 con un nivel de ruido  $\sigma_w = \sqrt{2}$  (primera columna de la Tabla 5.2). En este caso nuestro método proporciona el mejor resultado, tanto visualmente como en términos de PSNR. En particular, hay una gran diferencia en las líneas diagonales del mantel, donde nuestro método permite una mejor recuperación de las mismas. En el caso de *House*, la imagen ha sido degradada usando el filtro de convolución PSF2 con un nivel de ruido  $\sigma_w = \sqrt{0.308}$  (tercera columna de la Tabla 5.2). En esta ocasión destaca el pobre resultado proporcionado por la versión de restauración BLS-GSM (debido a las razones que explicamos en secciones anteriores). Respecto a los métodos del estado del arte [61, 69], nuestro algoritmo en este caso, aun por debajo en términos de error cuadrático, proporciona resultados competitivos.



**Figura 5.4.** Comparación visual de resultados para las imágenes de *House* y *Barbara*. Se muestran regiones de  $80 \times 80$  píxeles comenzando en el píxel (1, 78) y (1, 221) respectivamente (columna, fila). De izquierda a derecha y de arriba a abajo (*House/Barbara* PSNR en dB): original; imagen degradada (24.11/23.33) usando PSF2 y  $\sigma_w = \sqrt{0.308}$  para *House* y PSF1 y  $\sigma_w = \sqrt{2}$  para *Barbara*; resultados de [69] (34.76/29.05); resultados de [61] (34.76/27.09); resultados de GSM estándar [19] usando FSP (23.68/29.02); nuestra aportación [16] usando TIHP con *House* y FSP con *Barbara* (33.11/30.17).

Finalmente, también hemos probado nuestro algoritmo de deconvolución con degradaciones reales [15, 16, 14]. Las observaciones corresponden a capturas de una micro-pantalla LCD en condiciones de laboratorio controladas. Para producir el emborronamiento se utilizó una lente inclinada. El nivel de ruido presente en la observación utilizada corresponde al promediado de 30 capturas. En la Figura 5.5 se muestra la imagen degradada junto con el resultado de nuestro método, tras haber realizado una estimación de los parámetros de degradación. Como podemos comprobar, el algoritmo muestra un buen comportamiento, mejorando sustancialmente la calidad visual de la imagen.



**Figura 5.5.** Resultado del método de deconvolución SVGSM aplicado a una degradación real.

## 5.6. Conclusiones

En este Capítulo hemos presentado un algoritmo de eliminación de emborronamiento en presencia de ruido basado en un esquema en dos pasos: compensación global lineal de emborronamiento y eliminación local no lineal de ruido no blanco (SVGSM). Esta estrategia proporciona flexibilidad para usar cualquier método de eliminación de ruido siempre que pueda tratar con ruido Gaussiano de densidad espectral arbitraria. El método obtenido supera las limitaciones impuestas por aproximaciones locales. Además, y a diferencia de algunos métodos de la literatura, proporciona una solución simple y robusta al problema de modelar y estimar el espectro de potencia de la señal original, sin necesidad de realizar re-estimaciones.

Como hemos mostrado, el acoplamiento empírico entre los dos pasos del método es sub-óptimo. Una solución más elegante hubiera sido obtener un modelo del error del algoritmo SVGSM. Lamentablemente, dicho modelo es difícil de obtener, dado el carácter no lineal del método eliminación de ruido. A pesar de ello, la aproximación empírica para la elección de los parámetros es simple y robusta: hemos comprobado que los mismos parámetros pueden ser utilizados en degradaciones muy diferentes, sin provocar una caída seria del rendimiento. Esta aproximación es también muy flexible, al permitir ajustar los parámetros para cualquier degradación sin más que entrenar con filtros de la forma deseada. Todas estas características junto con un coste computacional moderado (e independiente de la imagen) hacen que este método sea adecuado para restaurar imágenes degradadas en situaciones reales [15].

## Capítulo 6

# Identificación Bayesiana de cámaras a partir de sus imágenes

### 6.1. Introducción

El estudio forense de imágenes es un área que comprende multitud de aplicaciones cuyo objetivo común es extraer información relacionada con la veracidad o procedencia de una imagen, como por ejemplo detectar si la imagen ha sido trucada para modificar la escena que contiene (autenticación de contenido), o manipulada en algún sentido para incluir información secreta (esteganografía), etc... En este Capítulo presentamos una aplicación que pretende, a partir de la imagen observada, determinar el modelo de cámara con la que fue obtenida.

La mayoría de las cámaras digitales incluyen en las cabeceras de los archivos de imagen información sobre el dispositivo, condiciones de captación, fecha, hora, etc... Sin embargo, esta información puede ser fácilmente editada y modificada sin dejar rastro alguno, haciendo que estos datos sean poco fiables. En los últimos años han proliferado numerosos estudios y propuestas sobre marcas de agua digital (*watermarking*, en inglés), que proporcionarían una vía para resolver el problema de la autenticación (ver [72, 73, 74], para una descripción general). La idea principal se basa en la introducción de algún tipo de *firma* en el momento de adquirir la escena. Parece lógico suponer que este tipo de tecnología permitiría también resolver el problema de la identificación de cámaras. Sin embargo, estas técnicas necesitarían que los dispositivos de adquisición estuviesen dotados de una electrónica dedicada, hecho que actualmente no es contemplado por los fabricantes. Para intentar dar solución al problema, algunos autores [75, 76, 77] propusieron métodos de identificación que utilizaban como característica discriminante el rastro dejado por las imperfecciones de la electrónica de adquisición de cada dispositivo particular. Para su aplicación, los métodos requieren un estudio previo minucioso de los dispositivos involucrados,



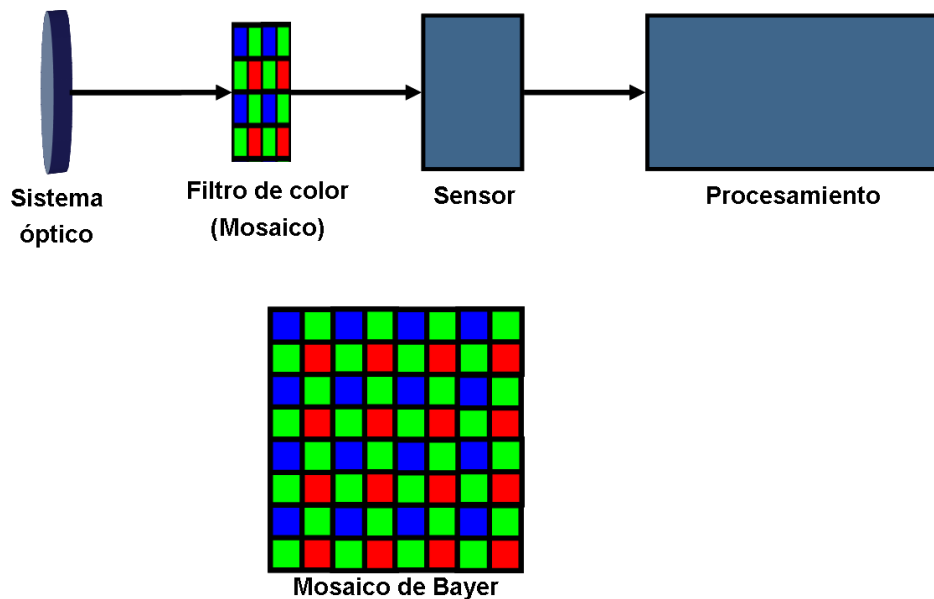
quedando pues descartada su aplicación a otras muchas situaciones prácticas en las que no se dispone de esta información.

En 2004, Kharrazi *et al.* [78] propusieron utilizar un conjunto de medidas de la imagen (media de cada píxel, correlación y energía de pares de color, estadísticos de coeficientes ondícula, etc) como características para identificar el dispositivo utilizado. Los resultados obtenidos fueron muy positivos, aunque su aplicabilidad en condiciones reales es más que dudosa, ya que el clasificador utilizado, basado en máquinas de vectores soporte [79] (SVM, del inglés *Support Vector Machines*), era entrenado con imágenes adquiridas en entornos controlados y que compartían el mismo tipo de contenido. Un año después, Bayram *et al.* [80, 81] inspirados en trabajos de autenticación [82, 83], presentaron un algoritmo que utilizaba como propiedad discriminante las trazas que dejaban los algoritmos de interpolación de las cámaras. Al igual que el trabajo anterior, el clasificador se basaba en máquinas de vectores soporte (SVM) y era entrenado con imágenes tomadas en entornos controlados. Ese mismo año, de forma paralela e independiente, Portilla *et al.* [20, 21] registraron una patente donde presentaban, entre otras cosas, un método para obtener estas trazas, reivindicando la potencial utilidad de las mismas para identificar dispositivos de captación.

En este Capítulo presentamos un algoritmo de identificación basado también en las trazas de los algoritmos de interpolación. A diferencia de otros trabajos, las características no se extraen de forma independiente en cada canal de color [83], sino que se extraen de forma óptima teniendo en cuenta todos los canales, siguiendo el procedimiento propuesto por Portilla *et al.* [20, 21]. Nuestra solución se basa en la construcción de un clasificador Bayesiano, presentando un modelo estadístico de cada una de las cámaras implicadas en la identificación. Además, el entrenamiento de los clasificadores se lleva a cabo con imágenes adquiridas en entornos reales, sin restringir ninguno de los parámetros de captación, por lo que los resultados obtenidos pueden considerarse mucho más realistas que los presentados en los trabajos previos mencionados.

## 6.2. Formación de imágenes en cámaras digitales

Los procesos involucrados en la formación de una imagen digital pueden variar enormemente dependiendo del fabricante, incluso podrían existir diferencias entre distintos modelos de una misma marca. Sin embargo, podemos establecer un esquema simplificado que se ajusta a la inmensa mayoría de las cámaras digitales de consumo. En la parte de arriba de la Figura 6.1 podemos ver los procesos que permiten formar la imagen digital. La luz se introduce dentro de la cámara a través del sistema óptico y es dirigida hacia un sensor compuesto por una matriz de fotodetectores. Cada uno de ellos se corresponde con un píxel de la imagen. El sistema más común (y barato) consiste en utilizar una matriz de fotodetectores monocromáticos (sensor) sobre el cual se coloca un mosaico de filtros de color



**Figura 6.1.** Arriba: Esquema de la formación de la imagen. Abajo: Mosaico de Bayer.

(CFA, del inglés *Color Filter Array*). De esta forma, en cada píxel, se capta un solo componente de color de los tres necesarios (RGB del inglés *Red-Green-Blue*), de acuerdo a la estructura establecida por el mosaico CFA. Uno de los patrones más utilizados es el mosaico de Bayer [84]. En la parte de abajo de la Figura 6.1 se muestra una matriz  $8 \times 8$  formada por repeticiones de un patrón Bayer  $2 \times 2$ . Con este sistema, la imagen adquirida (ya en formato electrónico) está incompleta, por lo que se le aplica una interpolación para recuperar los colores perdidos, proceso conocido como "de-mosaico". La imagen completa es finalmente sometida, entre otros procesos, a correcciones de color y compresión antes de su almacenamiento final en la memoria del dispositivo. En particular, el proceso de de-mosaico se realiza, típicamente, mediante una combinación (lineal o no) de los píxeles vecinos utilizando una matriz de pesos (filtro). En general, los fabricantes de cámaras aplican sus propios algoritmos de de-mosaico, utilizando filtros de diferentes formas y/o tamaños y con funciones de interpolación muy variadas.

El hecho de que muchos de los fabricantes compartan los procesos descritos implica que las imágenes tomadas con la mayoría de las cámaras posean características similares. Sin embargo, como hemos avanzado, el proceso de de-mosaico va fuertemente ligado al fabricante de cada dispositivo y por ello es susceptible de ser utilizado como característica diferenciadora entre imágenes de distintas cámaras. En la siguiente Sección describimos los procedimientos presentados por Portilla *et al.* [20, 21] para extraer esta información a partir de una imagen.

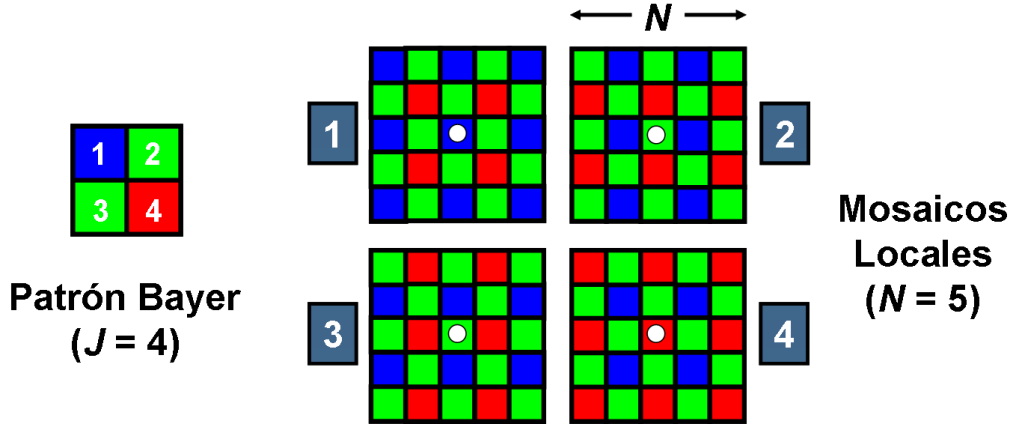


Figura 6.2. Mosaicos locales para un patrón de Bayer clásico. Figura tomada de [21]

### 6.3. Extracción de características

Las imágenes tomadas por cámaras digitales provistas de un mosaico de color (CFA) pueden modelarse como:

$$\mathbf{y} = \mathbf{S}\tilde{\mathbf{x}},$$

donde  $\mathbf{S}$  representa el muestreo de color llevado a cabo por el filtro CFA. Siendo  $P$  el número de píxeles de la imagen,  $\tilde{\mathbf{x}}$  corresponde a la imagen de color completa que contiene un total de  $3P$  elementos, a diferencia de la imagen mosaico,  $\mathbf{y}$ , que posee tan solo  $P$  elementos. Considerando, como aproximación, que el proceso de de-mosaico es lineal, la imagen interpolada observada se obtiene como sigue:

$$\tilde{\mathbf{y}} \approx \mathbf{D}\mathbf{y}. \quad (6.1)$$

La información que utilizaríamos para identificar las cámaras está contenida en la matriz  $\mathbf{D}$ , de dimensión  $3P \times P$ . Asumiendo conocidas la estructura y fase del mosaico, podríamos utilizar la pseudoinversa de  $\mathbf{y}$  ( $\mathbf{y}^\# = (\mathbf{y}\mathbf{y}^T)^{-1}\mathbf{y}$ ) para obtener la información deseada. Sin embargo, aun en el hipotético caso de poder calcular la inversa involucrada (de tamaño  $P \times P$ ), el sistema planteado posee muchas más incógnitas ( $3P^2$ ) que ecuaciones ( $3P$ ), proporcionando soluciones inestables. En lugar de este enfoque, Portilla *et al.* [20, 21] propusieron una solución local, forzando a los estimadores lineales a tener un soporte espacial limitado que solo considera a un conjunto de vecinos alrededor del coeficiente de referencia. Si bien esta ventana local proporciona una solución más abordable, el hecho de limitar el soporte de estimación provoca que aparezcan diferentes posibilidades de estructura de mosaico local, dependiendo de la localización en el sensor. El mosaico está hecho de la repetición de un patrón básico formado por  $J$  elementos,

de los  $N_c$  colores distintos del sensor ( $J \geq N_c$ ). De acuerdo a esto, existen  $J$  posibilidades de mosaico local en forma de ventana cuadrada de tamaño impar alrededor de cada píxel. En la Figura 6.2 se ilustra este hecho para el caso típico de un mosaico tipo Bayer en el cual  $N_c = 3$  (3 colores: rojo, verde y azul) y  $J = 4$ . Existen por tanto 4 posibilidades de mosaico local, resultado de colocar como referencia cada uno de los  $J$  componentes del patrón básico.

Para tratar el problema de forma local se hace necesaria una reordenación de la información. Asumiendo conocida la fase del patrón básico de mosaico, se divide la imagen en  $J$  conjuntos, de acuerdo al tipo de mosaico al que pertenece cada píxel. El número de píxeles de cada conjunto es  $L_j$ . Sea  $\mathbf{y}_i^j$  el vector de las muestras teóricamente<sup>1</sup> desconocidas de un píxel  $(i, j)$ , donde  $j$  indica un conjunto de los  $J$  posibles, e  $i$  indica el número de píxel de los  $L_j$  posibles pertenecientes al conjunto de mosaico  $j$ . Construimos las matrices  $\mathbf{Y}_j = \{\mathbf{y}_1^j, \mathbf{y}_2^j, \dots, \mathbf{y}_{L_j}^j\}$ . Cada una de estas matrices tiene dimensión  $(N_c - 1) \times L_j$ . En este punto, hacemos explícita la restricción de estimación local forzando a que la predicción sea realizada utilizando tan solo  $N \times N$  muestras alrededor de cada píxel. Así, definimos el vector  $\tilde{\mathbf{y}}_i^j$  como aquel que contiene todas las muestras de color teóricamente<sup>2</sup> observadas en una ventana  $N \times N$  alrededor del píxel  $(i, j)$  ( $N^2$  elementos). Se construye una matriz con todas las vecindades, siguiendo el orden anterior:  $\tilde{\mathbf{Y}}_j = \{\tilde{\mathbf{y}}_1^j, \tilde{\mathbf{y}}_2^j, \dots, \tilde{\mathbf{y}}_{L_j}^j\}$ . De forma análoga al planteamiento global de la Ecuación 6.1, podemos predecir las muestras de color desconocidas, para cada tipo de mosaico  $j$ , como combinación lineal de los vecinos observados:

$$\mathbf{Y}_j = \mathbf{D}_j \tilde{\mathbf{Y}}_j.$$

Al igual que anteriormente, la resolución de este sistema implica una pseudoinversa, pero en este caso de dimensión mucho menor que antes ( $N^2 \times N^2$ ). De esta forma, los  $J$  filtros interpoladores se obtendrían como:

$$\mathbf{D}_j = \mathbf{Y}_j (\tilde{\mathbf{Y}}_j)^\# = \mathbf{Y}_j (\tilde{\mathbf{Y}}_j \tilde{\mathbf{Y}}_j^T)^{-1} (\tilde{\mathbf{Y}}_j). \quad (6.2)$$

El procedimiento descrito asume la fase del mosaico conocida, sin embargo, en general, no se conocerá ni la fase ni la estructura del mosaico (aunque esta última será típicamente Bayer). Para solventar este problema se utiliza el método de estimación de fase, descrito también en [20]. El método consiste en calcular los  $J$  interpoladores para las  $J$  fases posibles y para las  $T$  estructuras de mosaico consideradas y elegir aquellos que proporcione menor error cuadrático medio entre la observación y el resultado de interpolar linealmente un mosaico de la observación con la estructura y la fase de mosaico seleccionadas.

<sup>1</sup>Nótese que se parte de una imagen observada que posee todos los componentes de color. Las muestras "desconocidas" serán resultado de aplicar una máscara de acuerdo a la fase del mosaico.

<sup>2</sup>De forma análoga a lo anterior, las muestras "observadas" se corresponden con los componentes de color del mosaico local asociado.

Los vectores de características,  $\mathbf{x}$ , que utilizaremos para identificar las cámaras son extraídos de los  $\mathbf{D}_j$  filtros. Se dispone de  $J(Nc - 1)$  filtros de  $N^2$  coeficientes, uno para cada tipo de mosaico  $j$  y para cada color desconocido  $(Nc - 1)$ . Gracias a la simetría que presentan los filtros podemos reducir el tamaño de los mismos a un total de  $((N + 1)/2)^2$  coeficientes, siendo la dimensión final de las características de  $(Nc - 1)J((N + 1)/2)^2$ .

## 6.4. Clasificador Bayesiano para la identificación de cámaras

Retomando el ejemplo que mostramos en el Capítulo 2, en esta Sección describiremos en detalle la construcción de un clasificador Bayesiano para la identificación de cámaras, usando como característica diferenciadora los filtros de interpolación de de-mosaico extraídos en la Sección anterior.

A partir de un conjunto de  $K$  cámaras,  $\{c_1, c_2, \dots, c_K\}$ , y de una imagen observada, se pretende determinar la cámara que tomó la citada imagen. Para ello, como vimos anteriormente, nos basaremos en el paradigma Bayesiano:

$$P(c_k|\mathbf{x}) = \frac{p(\mathbf{x}|c_k)P(c_k)}{p(\mathbf{x})}, k = 1..K, \quad (6.3)$$

donde  $\mathbf{x}$  representa un vector de características obtenido de la imagen observada, en este caso los coeficientes de los filtros de interpolación. A lo largo de la Sección describiremos dos modelos de cámaras para caracterizar la verosimilitud de las características ( $p(\mathbf{x}|c_k)$ ) y discutiremos sobre la utilización de dos posibles reglas de decisión.

### 6.4.1. Modelo de cámaras

De acuerdo a la regla de Bayes de la Ecuación 6.3, para construir nuestro clasificador necesitaremos una medida que nos indique cómo de probable es cada cámara (prior,  $P(c_k)$ ) y una función densidad de probabilidad que nos dé información sobre la verosimilitud que proporciona un determinado vector de características,  $\mathbf{x}$ , para una determinada cámara  $c_k$  (verosimilitud,  $p(\mathbf{x}|c_k)$ ). En este trabajo hemos considerado que todas las cámaras son igualmente probables (prior uniforme). Por otro lado, hemos modelado la verosimilitud siguiendo dos modelos diferentes, uno basado en una distribución Gaussiana y otro, un poco más elaborado, basado en una mezcla de Gaussianas:

**Modelo Gaussiano** Este modelo coincide con el ejemplo que mostramos en la Sección 2.3.2, en el cual las  $K$  funciones densidad de probabilidad son modeladas como distribuciones Gaussianas, quedando perfectamente

definidas por su media y covarianza:

$$p(\mathbf{x}|c_k) = \frac{\exp(-(\mathbf{x} - \boldsymbol{\mu}_k)^T \mathbf{C}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)/2)}{((2\pi)^N |\mathbf{C}_k|)^{1/2}}.$$

**Modelo de mezclas de Gaussianas** A diferencia del modelo anterior, el modelo de mezclas de Gaussianas asume que cada distribución,  $p(\mathbf{x}|c_k)$ , es una suma ponderada de  $G_k$  Gaussianas, que, en general, están gobernadas por medias y covarianzas diferentes. El modelo de cada densidad posee la siguiente expresión:

$$p(\mathbf{x}|c_k) = \sum_{i=1}^{G_k} \alpha_{i,k} g_{i,k}(\mathbf{x} : \boldsymbol{\mu}_{i,k}, \mathbf{C}_{i,k}).$$

Como podemos observar, este modelo requiere para cada densidad  $c_k$ ,  $G_k$  veces más parámetros que el modelo anterior, más los  $G_k$  pesos que mezclan las distintas Gaussianas.

#### *Estimación de parámetros*

La estimación de los parámetros se corresponde con el entrenamiento del clasificador. Para ambos modelos y para cada cámara  $c_k$ , se cuenta con un conjunto de  $M_k$  muestras que se asumen independientes e idénticamente distribuidas (i.i.d.). Teniendo esto en cuenta:

$$p(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{M_k} | c_k) = \prod_{m=1}^{M_k} p(\mathbf{x}_m | c_k). \quad (6.4)$$

De esta forma, los parámetros de ambos modelos se estimarán siguiendo un criterio de máxima verosimilitud, es decir, obtendremos los parámetros que maximicen la probabilidad descrita en la Ecuación 6.4.

Para el caso de modelo Gaussiano, los parámetros a estimar son la media y covarianza de una distribución Gaussiana, cuyos estimadores de máxima verosimilitud son bien conocidos [25]:

$$\hat{\boldsymbol{\mu}}_k = \frac{\sum_{m=1}^{M_k} \mathbf{x}_m}{M_k}; \quad \hat{\mathbf{C}}_k = \frac{\sum_{m=1}^{M_k} (\mathbf{x}_m - \boldsymbol{\mu}_k)(\mathbf{x}_m - \boldsymbol{\mu}_k)^T}{M_k}.$$

La estimación de parámetros para el modelo de mezclas de Gaussianas es un poco más elaborada, aunque igualmente bien conocida [85]. Asumiendo conocido el número de Gaussianas que forman cada mezcla,  $G_k$ , necesitamos estimar un total de  $G_k$  medias, covarianzas y pesos. Siguiendo el mismo criterio que en el caso Gaussiano (ML), podemos encontrar un algoritmo iterativo (que puede formularse en términos de *Expectation-Maximization*) que converge a una solución

de máxima verosimilitud. El algoritmo, partiendo de un estado inicial, actualiza iterativamente los estimadores de acuerdo a las siguientes expresiones:

$$\alpha_{i,k}^{n+1} = \frac{1}{M_k} \sum_{m=1}^{M_k} \frac{g_{i,k}(\mathbf{x}_m) \alpha_{i,k}^n}{\sum_{j=1}^{G_k} g_{j,k}(\mathbf{x}_m) \alpha_{j,k}^n};$$

$$\boldsymbol{\mu}_{i,k}^{n+1} = \frac{\sum_{m=1}^{M_k} w_{i,k}(m) \mathbf{x}_m}{\sum_{m=1}^{M_k} w_{i,k}(m)};$$

$$\mathbf{C}_{i,k}^{n+1} = \frac{\sum_{m=1}^{M_k} w_{i,k}(m) (\mathbf{x}_m - \boldsymbol{\mu}_{i,k})(\mathbf{x}_m - \boldsymbol{\mu}_{i,k})^T}{\sum_{m=1}^{M_k} w_{i,k}(m)},$$

teniendo  $w_{i,k}(m)$  la siguiente expresión:

$$w_{i,k}(m) = \frac{g_{i,k}(\mathbf{x}_m) \alpha_{i,k}}{\sum_{j=1}^{G_k} g_{j,k}(\mathbf{x}_m) \alpha_{j,k}}.$$

#### 6.4.2. Regla de decisión

La finalidad última de nuestro clasificador Bayesiano es determinar la cámara con la que fue tomada una determinada imagen cuyo vector de características es  $\mathbf{x}$ . Según esto, la cámara elegida será aquella que, de acuerdo a la regla de Bayes (ver Ecuación 6.3), proporcione mayor valor de la probabilidad a posteriori,  $P(c_k|\mathbf{x})$ . Como dijimos en anteriores secciones, esta regla de decisión minimiza la probabilidad del error. No obstante, por cuestiones computacionales, podemos dividir la imagen observada en  $M$  regiones de un determinado tamaño ( $B \times B$ , por ejemplo) y considerar las mismas como distintas adquisiciones de una misma cámara, que serán clasificadas de forma independiente. En este particular, se hace necesaria una forma de combinar las decisiones de cada una de las regiones. En este trabajo estudiaremos dos maneras diferentes:

**Ad-hoc** Este criterio simplemente realiza un conteo de los resultados parciales (cámaras más probables a posteriori en cada región), seleccionando la cámara  $c_k$  con mayor número de votos.

**Basada en modelo** En esta ocasión, se considera que las distintas regiones son muestras independientes, con lo que la cámara  $c_k$  seleccionada será aquella que proporcione un mayor valor de probabilidad a posteriori, calculada en los mismos términos que planteaba la Ecuación 6.4.

## 6.5. Implementación

Para el desarrollo del clasificador Bayesiano hemos asumido una estructura de patrón básico inicial tipo Bayer [84] (ver Figura 6.2), lo que implica que  $N_c = 3$  (colores: rojo, verde y azul) y  $J = 4$  (mosaicos locales). La vecindad de estimación local utilizada es una región cuadrada de  $5 \times 5$  ( $N = 5$ ). Con estos parámetros, y tras la reducción por simetría, el vector de características queda con una dimensión de  $72 \times 1$ . La fase del mosaico es estimada de acuerdo al método descrito en la Sección 6.3 y detallado en [20].

Respecto al algoritmo iterativo para estimar los parámetros de la mezcla, se ha realizado una inicialización ad-hoc, tomando como covarianzas iniciales la covarianza global, y como medias iniciales  $G_k$  vectores de la muestra seleccionados aleatoriamente. La distribución de los pesos de la mezcla,  $\alpha_{i,k}$ , se supone inicialmente uniforme. Debido a la gran cantidad de parámetros a estimar en proporción con el número de muestras, en ocasiones es necesario regularizar las matrices de covarianzas que estén mal condicionadas. Para ello hemos añadido una constante a la diagonal de las mismas. La constante elegida ha sido el 1% de la varianza global.

## 6.6. Resultados y discusión

### 6.6.1. Fuente de los datos

Para analizar el rendimiento de los clasificadores se han seleccionado 4 cámaras comerciales de características similares:

- Nikon D70, con una resolución de  $3008 \times 2000$  ( $6 \cdot 10^6$  píxeles efectivos).
- Canon EOS 300D, con una resolución de  $3072 \times 2048$  ( $6.3 \cdot 10^6$  píxeles efectivos).
- Olympus E300, con una resolución de  $3264 \times 2448$  ( $8 \cdot 10^6$  píxeles efectivos).
- Konica Minolta A2, con una resolución de  $3264 \times 2448$  ( $8 \cdot 10^6$  píxeles efectivos).

Según fuentes contrastadas [86], todas poseen un filtro de color (CFA) basado en un mosaico de Bayer, aunque de fase desconocida. No obstante, si la estructura del mosaico fuese también desconocida, podríamos aplicar el método descrito en la Sección 6.3, no solo para identificar la fase, sino también para identificar la estructura del mosaico. No se conocen detalles sobre el algoritmo de interpolación utilizado por las cámaras seleccionadas. Se obtienen de [87] 10 imágenes de cada cámara, que toman escenas de muy variada índole. Las imágenes son divididas en 77 regiones de tamaño  $256 \times 256$ , obteniendo un total de 770 muestras por



cámara. Estas muestras son utilizadas para identificar la fase del mosaico de cada una de las cámaras, siguiendo el procedimiento descrito en la Sección 6.3. Una vez identificada la fase, se extraen las características (ver Sección 6.3) que serán finalmente utilizadas para el entrenamiento y prueba de los clasificadores.

### 6.6.2. Entrenamiento de los clasificadores

Tanto para el modelo de cámaras Gaussiano, como para el de mezcla de Gaussianas, hemos realizado el entrenamiento utilizando el 20 %, 30 % y 40 % de las muestras, en sucesivos experimentos. Para ello hemos tomado aleatoriamente 2, 3 y 4 imágenes completas por cámara, respectivamente. Es importante destacar que de esta forma se evita la *contaminación* de los datos que provocaría el utilizar regiones en lugar de imágenes completas, las cuales distribuirían contenidos similares en los conjuntos de entrenamiento y de prueba, destruyendo la independencia de ambos conjuntos. Hemos comprobado empíricamente que una elección de las muestras totalmente aleatoria (sin restringir la selección al uso de imágenes completas) proporciona un incremento espectacular de los resultados, que, al igual que ocurre en otros trabajos [78, 80, 81], dejan así de ser representativos de una situación realista. Cabe destacar también que las características utilizadas pueden considerarse como separables, ya que hemos obtenido prácticamente un 100 % de aciertos cuando se prueba el clasificador con el mismo conjunto utilizado en el entrenamiento.

En este trabajo mostramos un clasificador que utiliza todas las cámaras ( $K = 4$ ) así como clasificadores para todas las posibles parejas de cámaras ( $K = 2$ ). El entrenamiento del clasificador consiste en modelar las  $K$  densidades  $p(\mathbf{x}|c_k)$  de cada cámara. En los casos de clasificadores con modelo de cámara Gaussiano, el entrenamiento se realiza simplemente estimando las medias y covarianzas de cada clase de acuerdo a las ecuaciones mostradas en la Sección 6.4.1. En los casos en los que se utiliza un modelo de cámara de mezcla, el entrenamiento es algo más elaborado. En primer lugar tenemos que determinar el número máximo de componentes por mezcla, que debe representar un compromiso razonable entre el número de parámetros a estimar y el tamaño de la muestra. En este caso, y por simplicidad, hemos seleccionado un mismo valor ( $G_{max} = 5$ ) para todas las mezclas ( $K$ ). De esta forma, para un clasificador de  $K$  cámaras tendremos  $G_{max}^K$  posibles configuraciones. El siguiente paso consiste en seleccionar una de las configuraciones posibles. Para ello hemos utilizado el algoritmo de validación cruzada de orden  $p$ , cuyas entradas son: un clasificador y un conjunto de muestras. Las muestras se dividen en  $p$  partes disjuntas, de las cuales  $p - 1$  se destinan a entrenamiento y 1 a prueba. La salida del algoritmo es el número de aciertos obtenidos en las  $p$  posibles particiones. En este caso, hemos considerado un valor de  $p = 5$  como un buen compromiso entre los tamaños de la partición y la muestra. Cada una de las  $G_{max}^K$  posibles configuraciones del clasificador es evaluada usando

validación cruzada un total de  $R = 20$  veces, para compensar la componente aleatoria del algoritmo de estimación de la mezcla (tipo *Expectation-Maximization*, con punto de partida aleatorio). La configuración seleccionada es aquella que proporciona un mayor número de aciertos. Finalmente, solo queda estimar los parámetros de las  $K$  mezclas (de acuerdo a la configuración seleccionada) utilizando el algoritmo iterativo presentado en la Sección 6.4.1. Es importante destacar que el proceso de entrenamiento descrito, aunque computacionalmente costoso (del orden de horas), solo tiene que realizarse una vez. Este alto coste en el entrenamiento contrasta con la rapidez de la prueba, que requiere menos de un segundo para clasificar 2000 muestras.

### 6.6.3. Prueba de los clasificadores

En la Tabla 6.1 se muestran los resultados para los 7 clasificadores entrenados, en términos de porcentaje de aciertos, cuando se introducen como entrada las muestras descritas en la Sección 6.6.2. La primera columna muestra los resultados para el modelo de cámara Gaussiano mientras que la segunda corresponde a los obtenidos con la mezcla de Gaussianas. En este último caso se muestra el porcentaje de acierto medio, así como la desviación típica del resultado, para un total de 100 repeticiones, teniendo así en cuenta la componente aleatoria del método de estimación de los parámetros de la mezcla. Los resultados se encuentran divididos en tres bloques, dependiendo del tamaño del conjunto de entrenamiento. Como podemos observar, el rendimiento del clasificador más sencillo (Gaussiano) es prácticamente el mismo que el obtenido por la mezcla de Gaussianas. La explicación de este comportamiento podría estar en la enorme cantidad de parámetros a estimar y lo reducido de las muestras. Sin embargo, este comportamiento se mantiene para todos los tamaños de conjunto de entrenamiento. En la Figura 6.3 se muestra, para una mejor visualización de los resultados, un diagrama de barras con los datos del bloque central de la Tabla 6.1. Como podemos observar, el comportamiento de ambos modelos es similar y parece indicar que la versión más elaborada no compensa, en términos de rendimiento, la sofisticación añadida.

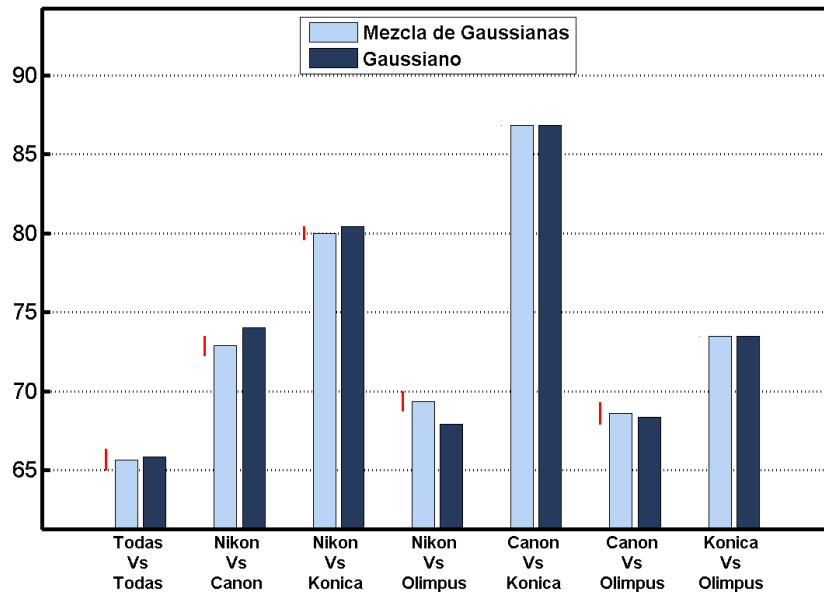
Como avanzábamos en la Sección 6.4.2, la finalidad última del clasificador es determinar la cámara con la que fue tomada una imagen. Dado que hemos dividido las imágenes en 77 regiones, es necesario combinar las respuestas. En la Figura 6.4 podemos ver un diagrama de barras que muestra los resultados obtenidos con las dos estrategias de decisión descritas con anterioridad: votación vs. modelo de probabilidad a posteriori. En la parte de arriba se muestran los resultados del modelo Gaussiano mientras que en la de abajo se muestran los asociados al modelo de mezcla. En ambos casos se ha seleccionado el 30% de las muestras como conjunto de entrenamiento. Según los resultados obtenidos, el esquema de voto proporciona un mayor rendimiento. Una posible explicación es

Modelo	<i>Gaussiano</i>	<i>Mezcla de Gaussianas</i>	
	<i>Entrenamiento: 20% de las muestras</i>		
<i>Comparativa</i>	Aciertos	Aciertos (promedio)	$\sigma_p$
Nikon Vs Canon	67.94 %	95.56 %	1.23 %
Nikon Vs Konica	76.70 %	75.83 %	1.33 %
Nikon Vs Olympus	69.97 %	69.30 %	1.07 %
Canon Vs Konica	85.06 %	84.14 %	0.98 %
Canon Vs Olympus	67.94 %	67.08 %	2.51 %
Konica Vs Olympus	76.54 %	73.13 %	3.66 %
Todas las cámaras	62.78 %	63.28 %	0.96 %
	<i>Entrenamiento: 30% de las muestras</i>		
<i>Comparativa</i>	Aciertos	Aciertos (promedio)	$\sigma_p$
Nikon Vs Canon	74.03 %	72.87 %	1.2 %
Nikon Vs Konica	80.43 %	80.01 %	0.84 %
Nikon Vs Olympus	67.90 %	69.35 %	1.22 %
Canon Vs Konica	86.83 %	86.83 %	0 %
Canon Vs Olympus	68.37 %	68.58 %	1.38 %
Konica Vs Olympus	73.47 %	73.47 %	0 %
Todas las cámaras	65.82 %	65.65 %	1.34 %
	<i>Entrenamiento: 40% de las muestras</i>		
<i>Comparativa</i>	Aciertos	Aciertos (promedio)	$\sigma_p$
Nikon Vs Canon	70.56 %	68.70 %	2.09 %
Nikon Vs Konica	80.95 %	80.34 %	1.58 %
Nikon Vs Olympus	79.55 %	78.19 %	2.55 %
Canon Vs Konica	86.26 %	85.81 %	0.72 %
Canon Vs Olympus	84.74 %	84.04 %	1.49 %
Konica Vs Olympus	90.26 %	89.42 %	0.93 %
Todas las cámaras	73.05 %	73.30 %	1.69 %

**Tabla 6.1.** Porcentaje de aciertos para distintos tamaños del conjunto de entrenamiento. La primera columna muestra los resultados de modelar las clases como Gaussianas. La segunda columna muestra, para un modelo de mezcla de Gaussianas, el porcentaje de acierto promedio junto con la desviación típica para un total de 100 repeticiones.

la presencia de regiones en la imagen cuyas características proporcionan valores atípicos, que a su vez provocan un valor de probabilidad asociado muy bajo. Estos valores castigan duramente al modelo de producto, mientras que no afectan gravemente al esquema de voto.

Respecto al comportamiento de los distintos modelos, en esta ocasión el modelo de mezcla supera sistemáticamente al modelo más sencillo cuando se utiliza el esquema de votación. Por contra, cuando se utiliza el modelo de probabilidad a posteriori, el comportamiento es similar.

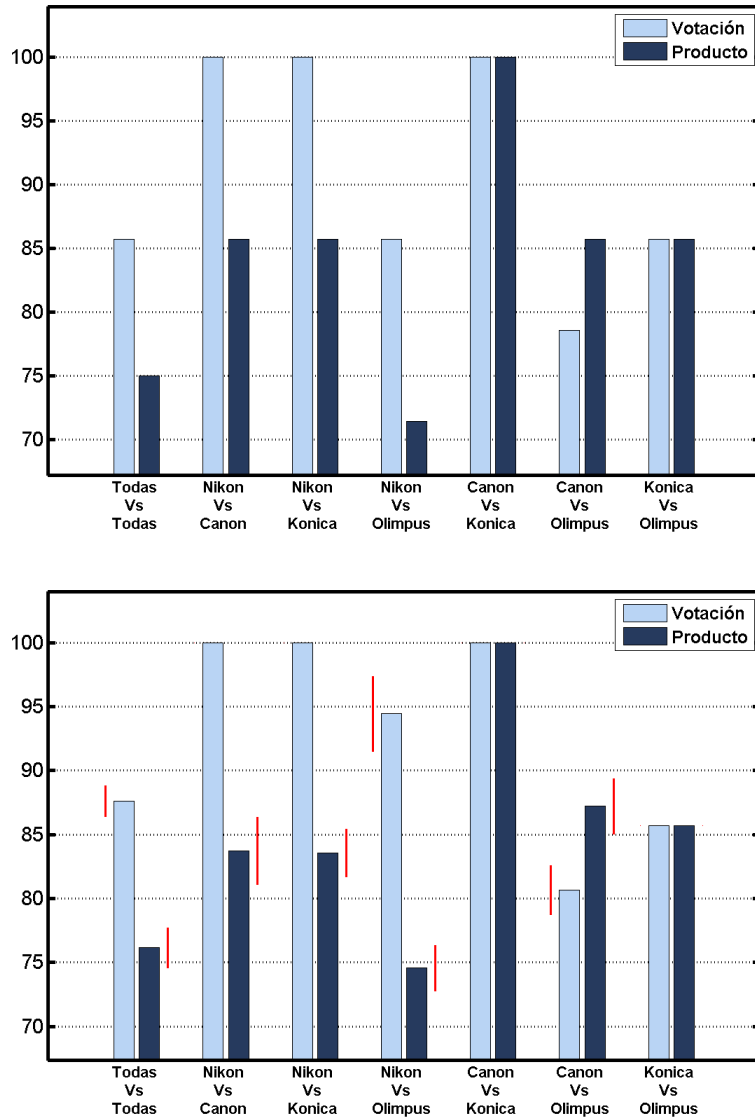


**Figura 6.3.** Comparativa de porcentaje de aciertos para un modelo Gaussiano y otro de mezcla de Gaussianas (incluyendo desviación media).

## 6.7. Conclusiones

En este Capítulo ha quedado demostrado que las trazas que dejan los algoritmos de interpolación en las imágenes permiten identificar, de forma razonablemente precisa, las cámaras con las que fueron adquiridas, mediante la utilización de clasificadores Bayesianos. Los filtros de interpolación utilizados como vectores de características dependen fuertemente del contenido de la escena, hecho que hemos comprobado empíricamente. En este trabajo hemos propuesto y comparado dos modelos, uno que caracteriza los descriptores de cada cámara con distribuciones Gaussianas y otro que utiliza mezclas de Gaussianas, concluyendo que la complejidad de este último no compensa en términos de rendimiento. Respecto a la regla de decisión ha quedado constatado que, en ausencia de procesos de detección de valores atípicos, el esquema ad-hoc (voto) proporciona mayor robustez y eficacia.

En definitiva, y a pesar de utilizar imágenes de muy diversa índole, el clasificador Bayesiano presentado proporciona resultados muy prometedores. No obstante, pensamos que los resultados pueden mejorar mediante el desarrollo de un procedimiento de detección de valores atípicos, y/o a través de un estudio más detallado de las características que permita encontrar una transformación que las haga menos dependientes del contenido de la escena.



**Figura 6.4.** Comparativa para reglas de decisión por voto y modelo producto. Arriba para un modelo de cámara Gaussiano, abajo para un modelo de mezcla de Gaussianas. En este último caso las barras rojas indican la desviación típica respecto de la media.

## Capítulo 7

# Conclusiones y trabajo futuro

### 7.1. Conclusiones

Uno de los objetivos principales de esta Tesis era desarrollar nuevos modelos estadísticos de imágenes que mejoraran al que hemos considerado como nuestro punto de partida: el modelo local basado en mezclas en escala de Gaussianas (GSM, del inglés *Gaussian scale mixtures*) descrito en [10]. En este trabajo hemos hablado de las limitaciones inherentes al modelo cuando intentamos representar características espectrales locales utilizando una sola matriz de covarianza por subbanda. Para superar estas limitaciones, hemos presentado dos nuevos modelos de imagen que, aunque están basados en GSM, siguen aproximaciones muy diferentes.

El primer modelo presentado, basado en la estimación de estadísticos locales, caracteriza a las subbandas de la imagen utilizando GSMs espacialmente variantes (SVGSM [11, 16]). Esta simple extensión proporciona una mejora significativa en términos de adaptación a las características locales, que será más o menos acentuada en función de la variabilidad espacial en covarianza existente en las subbandas. El método de eliminación de ruido presentado (basado en SVGSM), que hereda todas las propiedades deseables de su predecesor, proporciona una mejora significativa y generalizada, tanto en términos de error cuadrático como en aspecto visual, compensando claramente el moderado incremento de coste computacional. El método ha sido analizado en dos representaciones diferentes, con características complementarias: *Full Steerable Pyramid* (FSP) [10] y *Translation Invariant Haar Pyramid* (TIHP) [16]. De dicho análisis podemos extraer que la idoneidad de una representación frente a otra está en función del contenido de las imágenes en cuestión, concluyendo que FSP debería ser utilizada preferentemente para imágenes altamente texturadas, mientras que TIHP es más adecuada para imágenes con menos textura, especialmente si tienen orientaciones verticales y horizontales dominantes. Sin embargo, desde el punto de vista del coste computacional, TIHP representa una mejor opción, ya que, al poseer un factor

de redundancia menor, proporciona menos subbandas a procesar. Finalmente, al estar el rendimiento del método relacionado con la variabilidad espacial de la covarianza, hemos comprobado que la mejora relativa de SVGSM frente a GSM es mayor para la representación TIHP, ya que, en contraste con FSP, proporciona subbandas con mayor variabilidad espectral local.

Las limitaciones asociadas al modelo GSM local original se hacen más evidentes cuando se utiliza en aplicaciones más complejas, como la eliminación de emborronamiento en presencia de ruido aditivo Gaussiano [19]. En este trabajo, hemos analizado las serias limitaciones que plantean los modelos locales al tratar de compensar ciertos tipos de convolución, incluso cuando estos modelos son utilizados conjuntamente con esquemas multi-resolución. La solución que hemos presentado [12, 16, 14] se basa en un esquema en dos pasos: compensación lineal global de emborronamiento y eliminación local no lineal de ruido (basada en SVGSM). Este esquema permite superar algunas de las limitaciones inherentes a los modelos locales, mientras mantiene sus ventajas. Pese a que el acoplamiento de ambos pasos se realiza de forma empírica, el procedimiento propuesto es flexible y simple, mostrándose, a diferencia de otros métodos del estado del arte, como una solución robusta frente a errores en la estimación de la densidad espectral de potencia y frente a diferentes tipos de emborronamiento. Todos estos factores, junto con el moderado coste computacional (no dependiente del contenido de la imagen), hacen a este método muy útil para restaurar imágenes afectadas por fuentes de degradación real [15, 14].

A pesar de sus buenas propiedades, SVGSM no explota la redundancia no local presente en las imágenes naturales, a la vez que impone, en la práctica, severas limitaciones motivadas por el esquema de muestreo utilizado (basado en bloques). Estas limitaciones han servido de motivación para el desarrollo del otro modelo de imagen presentado en esta Tesis [18]. Como avanzamos, el enfoque de este modelo es muy diferente al propuesto anteriormente, presentando una solución que mantiene la aproximación local, pero utilizando una caracterización estadística global a nivel de subbanda. El modelo, basado en una mezcla de mezclas en escala de Gaussianas (MGSM, del inglés *Mixture of Gaussian Scale Mixtures*), es capaz de representar diferentes características espectrales locales dentro de una misma subbanda. Este hecho permite, además de aumentar el potencial de adaptación, obtener estimaciones más precisas gracias al aprovechamiento de la redundancia no local existente entre regiones no adyacentes de la subbanda. La solución presentada es conceptualmente limpia y el algoritmo de estimación de parámetros, basado en criterios de máxima verosimilitud, se plantea de forma iterativa, muy similar a los (bien conocidos) algoritmos EM. Esta forma de estimar los parámetros representa un cambio importante con respecto a los modelos GSM original y SVGSM: MGSM no necesita imponer priors sobre los parámetros de la mezcla, al estimarse de forma adaptativa para cada subbanda (siguiendo criterios de máxima verosimilitud). El modelo MGSM ha demostrado ser efectivo para

describir la estadística de las imágenes en el dominio transformado. Esto lo hace especialmente útil para la eliminación de ruido en imágenes, obteniendo una mejora significativa y generalizada sobre los modelos GSM y SVGSM. Creemos que el modelo MGSM representa una aportación potencialmente significativa al estado del arte, siendo la versión aquí presentada un primer paso hacia un modelo flexible y elegante que proporcione un nuevo compromiso, más favorable, entre capacidad de adaptación y sesgo en la estimación.

El otro objetivo planteado en esta Tesis consistía en el desarrollo de un procedimiento para identificar el modelo de cámara con el que fue adquirida una determinada imagen. Para ello hemos utilizado un clasificador Bayesiano, caracterizando las cámaras con dos sencillos modelos estadísticos: una Gaussiana y una mezcla de Gaussianas, ambos modelando la distribución de un vector de características propuesto con anterioridad [20, 21]. Las características utilizadas se basan en las trazas de los algoritmos de interpolación en los mosaicos CFA, que son, en general, diferentes para cada modelo de cámara. Nuestra propuesta, en contraste con otros métodos del estado del arte, hace especial hincapié en utilizar una metodología rigurosa que no restringe en absoluto los conjuntos de entrenamiento y prueba. Los razonables resultados obtenidos, incluso en condiciones no restringidas, nos hace pensar que el método es potencialmente aplicable en situaciones reales.

En resumen, este trabajo es una muestra más de la flexibilidad y potencia que ofrecen el paradigma Bayesiano y las mezclas de Gaussianas (continuas y discretas) como herramientas para desarrollar aplicaciones y modelos en el campo del procesamiento digital de imágenes. En particular, hemos experimentado ampliamente que la combinación de ambas proporciona excelentes resultados tanto en tareas de estimación como de clasificación.

## 7.2. Trabajo futuro

Creemos que la más inmediata fuente de futuras investigaciones puede venir de un desarrollo más amplio del nuevo modelo MGSM. A pesar del buen comportamiento mostrado, el modelo se encuentra en etapas preliminares y aún es necesario un estudio profundo y riguroso de sus propiedades. La principal limitación del modelo actual es el alto coste computacional que demanda su aplicación (típicamente alrededor de un orden de magnitud por encima de nuestra referencia basada en GSM local). Otro de los factores principales a mejorar está relacionado con el número de componentes de la mezcla utilizada en cada subbanda (número de GSMs). El ajuste adaptativo de este parámetro, que lógicamente dependerá del contenido de la subbanda, puede proporcionar una mejora tanto del rendimiento, como de la eficiencia computacional. Lo mismo puede decirse respecto a la elección, para cada GSM, de los componentes de la mezcla en escala. Otro punto importante viene dado por la forma de



considerar las muestras dentro de una subbanda. Actualmente, utilizamos todas las posibles vecindades (solapadas) dentro de las subbandas. Nos gustaría explorar dos alternativas diferentes. La primera de ellas es incluir un elaborado proceso de criba para reducir significativamente el solapamiento entre vecindades. Esto permitiría, no solo una reducción considerable del coste computacional y un fuerte aumento de la verosimilitud media (y, por tanto, de la capacidad descriptiva del modelo), sino la posibilidad de obtener un modelo sensible a la fase local. La principal desventaja sería la significativa pérdida de redundancia, un factor fundamental que afecta al sesgo de las estimaciones. Por contra, la segunda alternativa consiste en estudiar las posibles coincidencias de estructuras que se obtendrían al realizar operaciones sencillas (rotaciones y/o escalados) sobre las vecindades. Esta opción aumentaría sensiblemente la redundancia al precio de incrementar significativamente el coste computacional. Finalmente nos gustaría explorar otros criterios de estimación alternativos a la máxima verosimilitud, como por ejemplo, "Score Matching" [22, 23].

Respecto a las aplicaciones, pretendemos estudiar nuevos métodos de procesamiento de imágenes para el modelo MGSM, como por ejemplo la eliminación de emborronamiento en presencia de ruido o el análisis de texturas. Respecto a la aplicación forense, nos gustaría aumentar la robustez, eficiencia y tasa de aciertos del algoritmo de identificación de cámaras, mediante: (a) el desacoplamiento de la dependencia detectada entre el contenido de la imagen y el vector de características, (b) la reducción de la dimensión de los descriptores, y (c) utilizando clasificadores más elaborados, así como posibles combinaciones entre ellos.

# Apéndice



# Apéndice A

## Estimación de parámetros MGSM

### A.1. Modelo de señal: estimación de parámetros

En esta Sección describimos en detalle el cálculo de los parámetros del modelo de señal MGSM siguiendo un criterio de máxima verosimilitud. Asumiendo la existencia de un conjunto independiente de observaciones  $\{\mathbf{x}_m, m = 1 \dots M\}$  correspondientes a las vecindades observadas en una subbanda, la expresión a maximizar (verosimilitud) corresponde a:

$$L = \log \left[ \prod_{m=1}^M p(\mathbf{x}_m) \right] = \sum_{m=1}^M \log p(\mathbf{x}_m) = \sum_{m=1}^M \log \left[ \sum_{k=1}^K P_k \int_z p(\mathbf{x}_m|k, z) p_k(z) dz \right],$$

donde se ha aplicado la expresión de la función densidad de probabilidad de MGSM de la Ecuación 4.1. Las siguientes Secciones muestran por separado los detalles de cálculo de cada uno de los parámetros del modelo MGSM.

#### A.1.1. Probabilidad $P_k$ de cada componente de la mezcla:

La probabilidad  $P_k$  se puede expresar como:

$$P_k = \int_{\mathbf{x}} P(k|\mathbf{x}) p(\mathbf{x}) d\mathbf{x} = \mathbb{E}_{\mathbf{x}} \{P(k|\mathbf{x})\} \simeq \langle P(k|\mathbf{x}_m) \rangle_m,$$

donde  $\langle \cdot \rangle_j$  representa la media muestral en el índice  $j$ . La media de los posteriores asociados a cada muestra proporciona el estimador de máxima verosimilitud de la función de probabilidad involucrada [25]. Aplicando la regla de Bayes a cada posterior obtenemos una expresión de actualización recursiva (Ecuación 4.3):

$$P_k^{(n+1)} = P_k^{(n)} \frac{1}{M} \sum_{m=1}^M \frac{p(\mathbf{x}_m|k)^{(n)}}{\sum_{j=1}^K p(\mathbf{x}_m|j)^{(n)} P_j^{(n)}}, \quad (\text{A.1})$$

que es también el resultado del método *Expectation-Maximization* (EM) [88] clásico para el cálculo de la probabilidad de cada componente en una mezcla discreta. La expresión de verosimilitud condicionada a un componente de la mezcla se obtiene integrando su dependencia con el factor de escala:

$$p(\mathbf{x}|k)^{(n)} = \int_z p(\mathbf{x}|k, z)^{(n)} p_k(z)^{(n)} dz.$$

### A.1.2. Densidades $p_k(z)$ de los factores de escala:

Cada función densidad de probabilidad del factor de escala del componente  $k$  de la mezcla puede ser expresada como  $p_k(z) = p(z|k) = \int_{\mathbf{x}} p(z|\mathbf{x}, k) p(\mathbf{x}|k) d\mathbf{x}$ . Aplicando la regla de Bayes a  $p(\mathbf{x}|k)$ , obtenemos:

$$p_k(z) = \frac{1}{P_k} \int_{\mathbf{x}} p(z|\mathbf{x}, k) P(k|\mathbf{x}) p(\mathbf{x}) d\mathbf{x},$$

que tras descomponer los posteriores (regla de Bayes) y utilizar los estimadores muestrales de máxima verosimilitud (promedios muestrales) resulta finalmente (Ecuación 4.4):

$$p_k(z)^{(n+1)} = \frac{p_k(z)^{(n)}}{MP_k^{(n)}} \sum_{m=1}^M \frac{p(\mathbf{x}_m|k, z)^{(n)} P(k|\mathbf{x}_m)^{(n)}}{p(\mathbf{x}_m|k)^{(n)}}. \quad (\text{A.2})$$

### A.1.3. Matrices de covarianza $\mathbf{C}_k$ :

Para obtener las estimaciones de máxima verosimilitud a partir de  $L = \sum_{m=1}^M \log \left[ \sum_{k=1}^K P_k \int_z p(\mathbf{x}_m|k, z) p_k(z) dz \right]$ , es necesario derivar primero la siguiente verosimilitud condicional:

$$\begin{aligned} \frac{\partial p(\mathbf{x}_m|k, z)}{\partial \mathbf{C}_k} &= \frac{\partial}{\partial \mathbf{C}_k} \left[ \frac{\exp(-\mathbf{x}_m^T (z\mathbf{C}_k)^{-1} \mathbf{x}_m / 2) \equiv a}{(2\pi)^{N/2} |z\mathbf{C}_k|^{1/2} \equiv b} \right] \\ &= \frac{ab(z\mathbf{C}_k)^{-1} [\mathbf{x}_m \mathbf{x}_m^T \mathbf{C}_k^{-1} - I]}{2b^2} = \\ &= \frac{1}{2} p(\mathbf{x}_m|k, z) (z\mathbf{C}_k)^{-1} [\mathbf{x}_m \mathbf{x}_m^T \mathbf{C}_k^{-1} - I], \end{aligned}$$

donde se ha aplicado que:

$$\begin{aligned} \partial a / \partial \mathbf{C}_k &= \frac{a}{2} (z\mathbf{C}_k)^{-1} \mathbf{x}_m \mathbf{x}_m^T \mathbf{C}_k^{-1} \\ \partial b / \partial \mathbf{C}_k &= \frac{b}{2} (z\mathbf{C}_k)^{-1}. \end{aligned}$$

Sustituyendo y eliminando constantes<sup>1</sup>, la derivada con respecto a  $\mathbf{C}_k$  queda como:

$$\begin{aligned}
 \frac{\partial L}{\partial \mathbf{C}_k} &= \sum_{m=1}^M \frac{P_k}{p(\mathbf{x}_m)} \int_z p_k(z) \frac{\partial p(\mathbf{x}_m|k, z)}{\partial \mathbf{C}_k} dz \\
 &= \sum_{m=1}^M \frac{P_k}{p(\mathbf{x}_m)} \int_z p_k(z) p(\mathbf{x}_m|k, z) (z\mathbf{C}_k)^{-1} [\mathbf{x}_m \mathbf{x}_m^T \mathbf{C}_k^{-1} - I] dz \\
 &= \int_z \sum_{m=1}^M \frac{p(z|\mathbf{x}_m, k) p(\mathbf{x}_m|k) P_k}{p(\mathbf{x}_m)} (z\mathbf{C}_k)^{-1} [\mathbf{x}_m \mathbf{x}_m^T \mathbf{C}_k^{-1} - I] dz \\
 &= \int_z \sum_{m=1}^M p(z|\mathbf{x}_m, k) P(k|\mathbf{x}_m) (z\mathbf{C}_k)^{-1} [\mathbf{x}_m \mathbf{x}_m^T \mathbf{C}_k^{-1} - I] dz = 0.
 \end{aligned}$$

Descomponiendo la última ecuación en dos términos y pre/post multiplicando por  $\mathbf{C}_k$  obtenemos:

$$\int_z \sum_{m=1}^M p(z|\mathbf{x}_m, k) P(k|\mathbf{x}_m) \mathbf{x}_m \mathbf{x}_m^T z^{-1} dz = \int_z \sum_{m=1}^M p(z|\mathbf{x}_m, k) P(k|\mathbf{x}_m) z^{-1} \mathbf{C}_k dz,$$

que tras una reordenación de términos nos proporciona la expresión de actualización deseada (Ecuación 4.5):

$$\begin{aligned}
 \mathbf{C}_k^{(n+1)} &= \\
 &= \frac{\int_z \sum_{m=1}^M p(z|\mathbf{x}_m, k)^{(n)} P(k|\mathbf{x}_m)^{(n)} \mathbf{x}_m \mathbf{x}_m^T z^{-1} dz}{\int_z \sum_{m=1}^M p(z|\mathbf{x}_m, k)^{(n)} P(k|\mathbf{x}_m)^{(n)} z^{-1} dz}, \forall k.
 \end{aligned}$$

## A.2. Modelo de observación: demostración de estimación consistente

En esta Sección presentamos una demostración formal de la consistencia del estimador de la covarianza de la observación dados  $k$  y  $z$ ,  $\mathbf{C}_y(k, z)$ , mostrado en la Ecuación 4.13 de la Sección 4.4.3. Nuestro objetivo es demostrar que:

$$\lim_{M \rightarrow \infty} \mathbf{C}_y^{(M)}(k, z) = z\mathbf{C}_k + \mathbf{C}_w.$$

---

<sup>1</sup>Irrelevantes a efectos de maximización.

Para ello desarrollamos:

$$\begin{aligned}
z\mathbf{C}_k + \mathbf{C}_w = \mathbb{E}_{\mathbf{y}}\{\mathbf{y}\mathbf{y}^T|k, z\} &= \int_{\mathbf{y}} p(\mathbf{y}|k, z)\mathbf{y}\mathbf{y}^T d\mathbf{y} = \int_{\mathbf{y}} \frac{p(k, z|\mathbf{y})p(\mathbf{y})}{p(k, z)}\mathbf{y}\mathbf{y}^T d\mathbf{y} \\
&= \frac{\int_{\mathbf{y}} p(k, z|\mathbf{y})p(\mathbf{y})\mathbf{y}\mathbf{y}^T d\mathbf{y}}{\mathbb{E}_{\mathbf{y}}\{p(k, z|\mathbf{y})\}} \\
&= \frac{\int_{\mathbf{y}} p(k, z|\mathbf{y})p(\mathbf{y})\mathbf{y}\mathbf{y}^T d\mathbf{y}}{\int_{\mathbf{y}} p(k, z|\mathbf{y})p(\mathbf{y}) d\mathbf{y}} \\
&= \frac{\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M p(k, z|\mathbf{y}_m)\mathbf{y}_m\mathbf{y}_m^T}{\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M p(k, z|\mathbf{y}_m)} \\
&= \frac{\lim_{M \rightarrow \infty} \sum_{m=1}^M P(k|\mathbf{y}_m)p(z|\mathbf{y}_m, k)\mathbf{y}_m\mathbf{y}_m^T}{\lim_{M \rightarrow \infty} \sum_{m=1}^M P(k|\mathbf{y}_m)p(z|\mathbf{y}_m, k)} \\
&= \lim_{M \rightarrow \infty} \mathbf{C}_{\mathbf{y}}^{(M)}(k, z),
\end{aligned}$$

como queríamos demostrar. En la anterior derivación hemos aplicado que, en el límite, la media y el promedio coinciden, con lo que:

$$\begin{aligned}
\int_{\mathbf{y}} p(k, z|\mathbf{y})p(\mathbf{y})\mathbf{y}\mathbf{y}^T d\mathbf{y} &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M p(k, z|\mathbf{y}_m)\mathbf{y}_m\mathbf{y}_m^T \\
\int_{\mathbf{y}} p(k, z|\mathbf{y})p(\mathbf{y}) d\mathbf{y} &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M p(k, z|\mathbf{y}_m).
\end{aligned}$$

## Apéndice B

# Listado de publicaciones

A continuación se muestra una lista de trabajos que dan sustento a esta Tesis doctoral:

- J. A. Guerrero-Colón, M. A. T. Figueiredo and J. Portilla, "Realistic Bayesian camera identification based on demosaicing traces", IEEE Int'l Conf. on Image Proc., 2008, Submitted.
- J. A. Guerrero-Colón, E. P. Simoncelli and J. Portilla, "Image denoising using mixtures of Gaussian scale mixtures", IEEE Int'l Conf. on Image Proc., 2008, Submitted.
- J. A. Guerrero-Colón, L. Mancera, J. Portilla: "Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids". IEEE Trans. Image Proc., vol. 17, no. 1, pp. 27-41, January 2008.
- C. Dorronsoro, J. A. Guerrero-Colón, M. C. de la Fuente, J. M. Infante, J. Portilla, "Low-cost wavefront coding using coma and a denoising-based deconvolution" Comunicación Oral en SPIE Europe Symposium Optics/Photonics in Security & Defence. Electro-Optical and Infrared Systems: Technology and Applications. (September 2007).
- J. Portilla, J. A. Guerrero-Colón, "Image restoration using adaptive Gaussian scale mixtures in overcomplete pyramids". Invited talk in Wavelets XII, in SPIE's International Symposium on Optical Science and Technology. SPIE's 52th Annual Meeting Proc. of the SPIE (August 2007).
- Patente P200602116: "Procedimiento de restauración de imágenes afectadas por imperfecciones, dispositivo para llevarlo a cabo y sus aplicaciones", Javier Portilla (45%), J. A. Guerrero-Colón (45%) y C. Dorronsoro (10%). Entidades: Universidad de Granada y CSIC
- J. A. Guerrero-Colón and J. Portilla, "Deblurring-by-Denoising using spatially adaptive Gaussian scale mixtures in overcomplete pyramids", IEEE



Int'l Conf. on Image Proc., Atlanta, vol. I, pp. 625-628, Atlanta (GA USA), October 2006.

- J. A. Guerrero-Colón and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented Pyramids", IEEE Int'l Conf. on Image Proc., vol. I, pp. 105-108, Genoa (Italy) September 2005.

# Bibliografía

- [1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison-Wesley, second edition, 2002.
- [2] H. B. Barlow, “Possible principles underlying the transformation of sensory messages,” in *Sensory Communication*, W. A. Rosenblith, Ed., pp. 217–234. MIT Press, Cambridge, MA, 1961.
- [3] D. L. Ruderman and W. Bialek, “Statistics of natural images: Scaling in the woods,” *Phys. Rev. Letters*, vol. 73, no. 6, pp. 814–817, 1994.
- [4] D. J. Field, “Relations between the statistics of natural images and the response properties of cortical cells,” *J. Opt. Soc. Am. A*, vol. 4, no. 12, pp. 2379–2394, 1987.
- [5] A. B. Watson, “Efficiency of a model human image code,” *J. Opt. Soc. Am. A*, vol. 12, pp. 2401–2417, 1987.
- [6] E. P. Simoncelli and W. T. Freeman, “The steerable pyramid: A flexible architecture for multi-scale derivative computation,” in *IEEE Int’l Conf. on Image Proc.* Washington, DC (USA), October 1995, vol. III, pp. 444–447, IEEE Computer Society.
- [7] D. S. Taubman and M. W. Marcellin, *JPEG 2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, Norwell, MA, USA, 2001.
- [8] M. Raphan and E. P. Simoncelli, “Optimal denoising in redundant bases,” in *IEEE Int’l Conf. on Image Proc.* San Antonio, TX (USA), 2007, pp. III: 113–116, IEEE Computer Society.
- [9] M. J. Wainwright and E. P. Simoncelli, “Scale mixtures of Gaussians and the statistics of natural images,” in *Adv. Neural Information Processing Systems (NIPS)*, S. A. Solla, T. K. Leen, and K.-R. Müller, Eds., Cambridge, MA, May 2000, vol. 12, pp. 855–861, MIT Press.

- 
- [10] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, “Image denoising using scale mixtures of Gaussians in the wavelet domain,” *IEEE Trans. Image Proc.*, vol. 12, pp. 1338–1351, November 2003.
- [11] J. A. Guerrero-Colón and J. Portilla, “Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids,” in *IEEE Int’l Conf. on Image Proc.* Genoa, Italy, September 2005, vol. I, pp. 105–108, IEEE Computer Society.
- [12] J. A. Guerrero-Colón and J. Portilla, “Deblurring-by-denoising using spatially adaptive Gaussian scale mixtures in overcomplete pyramids,” in *IEEE Int’l Conf. on Image Proc.* Atlanta, GA (USA), October 2006, pp. 625–628, IEEE Computer Society.
- [13] J. Portilla and J. A. Guerrero-Colón, “Image restoration using adaptive Gaussian scale mixtures in overcomplete pyramids,” in *SPIE International Symposium on Optical Science and Technology*. San Diego, CA (USA), August 2007.
- [14] C. Dorronsoro, J. A. Guerrero-Colón, M. C. de la Fuente, J. M. Infante, and J. Portilla, “Low-cost wavefront coding using coma and a denoising-based deconvolution,” in *SPIE Europe Symposium Optics/Photonics in Security & Defence*. Florence, Italy, September 2007.
- [15] J. Portilla, J. A. Guerrero-Colón, and C. Dorronsoro, “Procedimiento de restauración de imágenes afectadas por imperfecciones, dispositivo para llevarlo a cabo y sus aplicaciones,” 2006, Patent, P2006021116.
- [16] J. A. Guerrero-Colón, L. Mancera, and J. Portilla, “Image restoration using space-variant Gaussian scale mixture in overcomplete pyramids,” *IEEE Trans. Image Proc.*, vol. 17, no. 1, pp. 27 – 41, January 2008.
- [17] J. A. Guerrero-Colón, M. A. T. Figueiredo, and J. Portilla, “Realistic Bayesian camera identification based on demosaicing traces,” in *IEEE Int’l Conf. on Image Proc.* San Diego, CA (USA), 2008, IEEE Computer Society, Submitted.
- [18] J. A. Guerrero-Colón, E. P. Simoncelli, and J. Portilla, “Image denoising using mixtures of Gaussian scale mixtures,” in *IEEE Int’l Conf. on Image Proc.* San Diego, CA (USA), 2008, IEEE Computer Society, Submitted.
- [19] J. Portilla and E. P. Simoncelli, “Image restoration using Gaussian scale mixtures in the wavelet domain,” in *IEEE Int’l Conf. on Image Proc.* Barcelona, Spain, September 2003, vol. 2, pp. 965–968, IEEE Computer Society.

- [20] J. Portilla, C. Dorronsoro, and D. Otaduy, “Método de predicción lineal óptima para la reconstrucción de la imagen en cámaras digitales con sensor de mosaico,” 2005, Patent, P200502100.
- [21] J. Portilla, D. Otaduy, and C. Dorronsoro, “Low-complexity linear demosaicing using joint spatial-chromatic image statistics,” in *IEEE Int’l Conf. on Image Proc.* Genoa, Italy.
- [22] A. Hyvärinen, “Estimation of non-normalized statistical models by score matching,” *Journal of Machine Learning Research*, vol. 6, pp. 695–709, 2005.
- [23] A. Hyvärinen, “Some extensions of score matching,” *Computational Statistics & Data Analysis*, vol. 51, no. 5, pp. 2499–2512, 2007.
- [24] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 2002.
- [25] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley-Interscience Publication, 2000.
- [26] S. Mallat, *A wavelet tour of signal processing*, Academic Press, second edition, 2001.
- [27] A. Haar, “Zur Theorie der orthogonalen Funktionensysteme,” *Annals of Mathematics*, vol. 69, pp. 331–371, 1910.
- [28] E. J. Candès and D. L. Donoho, *Curvelets: A Surprisingly Effective Nonadaptive Representation of Objects with Edges*, Vanderbilt University Press, Nashville, TN, 1999.
- [29] N. Kingsbury, “Image processing with complex wavelets,” September 1999, Phil. Trans. Royal Society London A.
- [30] E. R. Kretzmer, “The statistics of television signals,” *Bell System Tech. J.*, vol. 31, pp. 751–763, 1952.
- [31] N. G. Deriugin, “The power spectrum and the correlation function of the television signal,” *Telecommunications*, vol. 1, no. 7, pp. 1–12, 1956.
- [32] S. Mallat, “A theory for multiresolution signal decomposition: The wavelet representation,” *IEEE Pat. Anal. Mach. Intell.*, vol. 11, pp. 674–693, July 1989.
- [33] E. P. Simoncelli, “Statistical models for images: Compression, restoration and synthesis,” in *Proc. 31st Asilomar Conf on Signals, Systems and Computers*, Pacific Grove, CA, November 1997, pp. 673–678, IEEE Computer Society.

- [34] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Proc.*, vol. 41, no. 12, pp. 3445–3462, December 1993.
- [35] R. W. Buccigrossi, *Compression and Segmentation of Images Using an Inter-Subband Wavelet Probability Model*, Ph.D. thesis, University of Pennsylvania, Department of Computer and Information Science, Philadelphia PA, June 1999.
- [36] J. Portilla and E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *International Journal of Computer Vision*, vol. 40, no. 1, pp. 49–70, 2000.
- [37] D. Andrews and C. Mallows, "Scale mixtures of normal distributions," *J. Royal Stat. Soc.*, vol. 36, pp. 99–102, 1974.
- [38] J. S. Lee, "Digital image enhancement and noise filtering by use of local statistics," *IEEE Pat. Anal. Mach. Intell.*, vol. PAMI-2, pp. 165–168, March 1980.
- [39] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.
- [40] S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," in *IEEE Int'l Conf. on Image Proc.* Chicago, IL (USA), October 1998, vol. 1, pp. 535–539, IEEE Computer Society.
- [41] A. Pižurica, W. Philips, I. Lemahieu, and M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Proc.*, vol. 11, no. 5, pp. 545–557, May 2002.
- [42] L. Şendur and I. W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Trans. Signal Proc.*, vol. 50, no. 11, pp. 2744–2756, November 2002.
- [43] M. K. Mihçak, I. Kozintsev, K. Ramchandran, and P. Moulin, "Low-complexity image denoising based on statistical modeling of wavelet coefficients," *IEEE Trans. Signal Proc.*, vol. 6, no. 12, pp. 300–303, December 1999.
- [44] X. Li and M. T. Orchard, "Spatially adaptive image denoising under overcomplete expansion," in *IEEE Int'l Conf. on Image Proc.* Vancouver, Canada, September 2000, vol. 3, pp. 300–303, IEEE Computer Society.
- [45] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Adaptive Wiener denoising using a Gaussian scale mixture model in the wavelet

- domain,” in *IEEE Int’l Conf. on Image Proc.* Thessaloniki, Greece, October 2001, pp. 37–40, IEEE Computer Society.
- [46] S. Mallat, G. Papanicolaou, and Z. Zhang, “Adaptive covariance estimation of locally stationary processes,” *Ann. Stat.*, vol. 26, no. 1, pp. 1–47, 1998.
- [47] D. K. Hammond and E. P. Simoncelli, “Image denoising with an orientation-adaptive Gaussian scale mixture model,” in *IEEE Int’l Conf. on Image Proc.*, Atlanta, GA (USA), October 2006, pp. 1433–1436, IEEE Computer Society.
- [48] A. Buades, B. Coll, and J. M. Morel, “A non-local algorithm for image denoising,” in *IEEE Conf. on Comp. Vision and Pattern Rec.*, 2005, vol. 2, pp. 60–65.
- [49] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” 1998, pp. 839–846.
- [50] C. Kervrann and J. Boulanger, “Optimal spatial adaptation for patch-based image denoising,” *IEEE Trans. Image Proc.*, vol. 15, no. 10, pp. 2866–2878, October 2006.
- [51] K. Hirakawa and T. W. Parks, “Image denoising using total least squares,” *IEEE Trans. Image Proc.*, vol. 15, no. 9, pp. 2730–2742, September 2006.
- [52] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3-D transform-domain collaborative filtering,” *IEEE Trans. Image Proc.*, vol. 16, no. 8, pp. 2080–2095, August 2007.
- [53] A. Foi, V. Katkovnik, and K. Egiazarian, “Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images,” *IEEE Trans. Image Proc.*, vol. 16, no. 5, pp. 1395–1411, May 2007.
- [54] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Trans. Image Proc.*, vol. 15, no. 12, pp. 3736–3745, December 2006.
- [55] J. L. Starck, D. L. Donoho, and E. J. Candès, “Very high quality image restoration,” in *Proc. SPIE conf. Signal and Image Processing*, San Diego, August 2001, vol. 4478, pp. 9–19, A. Laine, M. Unser and A. Aldroubi, Eds.
- [56] S. Lyu and E. P. Simoncelli, “Statistical modeling of images with fields of Gaussian scale mixtures,” in *Adv. Neural Information Processing Systems (NIPS)*, May 2007, vol. 19.
- [57] E. Vansteenkiste, D. Van der Weken, W. Philips, and E. E. Kerre, “Perceived image quality measurement of state-of-the-art noise reduction schemes,” in *Advanced Concepts for Intelligent Vision Systems*, 2006, pp. 114–126.

- [58] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising with block-matching and 3D filtering,” in *Proc. SPIE Electronic Imaging: Algorithms and Systems V*, San Jose, CA, USA, January 2006, vol. 6064A-30.
- [59] J. Portilla, “Full blind denoising through noise covariance estimation using Gaussian scale mixtures in the wavelet domain,” in *IEEE Int’l Conf. on Image Proc.*, IEEE, Ed. Singapore, October 2004, pp. 1217–1220, IEEE Computer Society.
- [60] R. Molina, J. Mateos, A. K. Katsaggelos, and M. Vega, “Bayesian multichannel image restoration using compound Gauss-Markov random fields,” *IEEE Trans. Image Proc.*, vol. 12, no. 12, pp. 1642–1654, December 2003.
- [61] M. A. T. Figueiredo and R. Nowak, “A bound optimization approach to wavelet-based image deconvolution,” in *IEEE Int’l Conf. on Image Proc.* Genoa, Italy, vol. 2, pp. 782–785.
- [62] J. Bioucas-Dias, “Bayesian wavelet-based image deconvolution: a GEM algorithm exploiting a class of heavy-tailed priors,” *IEEE Trans. Image Proc.*, vol. 4, pp. 937–951, April 2006.
- [63] M. R. Banham and A. K. Katsaggelos, “Spatially adaptive wavelet-based multiscale image restoration,” *IEEE Trans. Image Proc.*, vol. 5, pp. 619–634, April 1996.
- [64] R. Neelamani, H. Choi, and R. G. Baraniuk, “Wavelet-domain regularized deconvolution for ill-conditioned systems,” in *IEEE Int’l Conf. on Image Proc.* Kobe, Japan, October 1999, vol. 1, pp. 204–208, IEEE Computer Society.
- [65] R. Neelamani, H. Choi, and R. G. Baraniuk, “ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems,” *IEEE Trans. Signal Proc.*, vol. 52, no. 2, pp. 418–433, February 2004.
- [66] J. Kalifa and S. Mallat, “Mini-max restoration and deconvolution,” in *Bayesian inference in wavelet based methods*. Springer, 1999.
- [67] A. Jalobeanu, N. Kingsbury, and J. Zerubia, “Image deconvolution using hidden Markov tree modeling of complex wavelet packets,” in *IEEE Int’l Conf. on Image Proc.* Thessaloniki, Greece, 2001, vol. 1, pp. 201–204, IEEE Computer Society.
- [68] A. W. Stevenson T. E. Gureyev, Y. I. Nesterets and S. W. Wilkins, “A method for local deconvolution,” *Applied Optics*, vol. 42, no. 32, pp. 6488–6494, November 2003.

- [69] A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-adaptive DCT for denoising and image reconstruction," in *SPIE Electronic Imaging 2006, Image Processing: Algorithms and Systems V*, 2006, pp. 6064A–18.
- [70] M. A. T. Figueiredo and R. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Trans. Image Proc.*, vol. 12, no. 8, pp. 906–916, August 2003.
- [71] J. Liu and P. Moulin, "Complexity-regularized image restoration," in *IEEE Int'l Conf. on Image Proc.* Chicago, IL (USA), 1998, vol. 1, pp. 555–559, IEEE Computer Society.
- [72] S. Katzenbeisser and F. Petitcolas, *Information techniques for Steganography and digital watermarking*, Artech House, 2000.
- [73] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking*, Morgan Kaufmann, 2002.
- [74] "Special issue on data hiding," vol. 41, 2003.
- [75] K. Kurosawa, K. Kuroki, and N. Saitoh, "CCD fingerprint method-identification of a video camera from videotaped images," in *IEEE Int'l Conf. on Image Proc.* Kobe, Japan.
- [76] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh, "Methods for identification of images acquired with digital cameras," in *Proc. SPIE, Enabling Technologies for Law Enforcement and Security*, February 2001, vol. 4232, pp. 505–512.
- [77] J. Lukas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Trans. Info. Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.
- [78] M. Kharrazi, H. T. Sencar, and N. D. Memon, "Blind source camera identification," in *IEEE Int'l Conf. on Image Proc.* Singapore.
- [79] C. Chang and C. Lin, *LIBSVM: a library for support vector machines*, 2001, Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [80] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, "Source camera identification based on CFA interpolation," in *IEEE Int'l Conf. on Image Proc.* Genoa, Italy, September 2005, vol. 3, pp. 69–72, IEEE Computer Society.
- [81] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, "Improvements on source camera-model identification based on CFA interpolation," in *International Conference on Digital Forensics*, February 2006.



- 
- [82] A. C. Popescu and H. Farid, “Exposing digital forgeries by detecting traces of re-sampling,” *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 758–767, 2005.
- [83] A. C. Popescu and H. Farid, “Exposing digital forgeries in color filter array interpolated images,” *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3948–3959, 2005.
- [84] B. E. Bayer, “Color imaging array,” 1976, US Patent, 3971065.
- [85] G. McLachlan and D. Peel, *Finite Mixture Models*, Wiley, 2000.
- [86] DPreview, “Web page,” <http://www.dpreview.com>.
- [87] A. Sierra, “Web page,” <http://www.albertsierra.fotki.com>.
- [88] T. Moon, “The expectation maximization algorithm,” *IEEE Signal Processing Magazine*, November 1996.

# Índice alfabético

- Análisis
  - multi-escala, 18
  - multi-resolución, 20
- autocorrelación, 11
- Bayes Least Squares, 13, 29
- Bayesiano, 12
- BLS, 13, 34
- BLS-GSM, 29
- campos aleatorios, 11
- CFA, 85
- compensación de emborronamiento,
  - 4
- CWT, 19
- deblurring, 4
- deconvolución, 4
- deconvolution, 4
- denoising, 4
- DFT, 15
- eliminación de ruido, 4
- EM, 35, 54
- Expectation-Maximization, 89
- FSP, 24
- Full Steerable Pyramid, 24
- Generalized EM, 54
- GSM, 6, 11, 28
- Haar, 20
- ISNR, 79
- MAP, 13
- MGSM, 49
- ML, 16
- mother wavelet, 19
- MSE, 6, 42
- OAGSM, 34
- ondícula, 19
- PSD, 12
- PSF, 15
- PSNR, 42
- RGB, 85
- SA-DCT, 35, 79
- STFT, 18
- Support Vector Machines, 84
- SVGSM, 37
- SVM, 84
- teorema de Nyquist, 21
- TIHP, 24
- TLS, 35
- Toeplitz, 12
- Total Least Squares, 35
- Transformada
  - ondicular, 19
  - de Fourier, 17
  - de Fourier enventanada, 17
- transformada discreta de Fourier, 15
- Translation Invariant Haar Pyramid,
  - 24
- watermarking, 83
- wavelet, 19
- Wiener, 14