



Descripción de texturas.
Aplicaciones a su comprensión y clasificación.

MEMORIA PARA OPTAR AL GRADO DE DOCTOR PRESENTADA EN
EL DEPARTAMENT D'ESTADÍSTICA I INVESTIGACIÓ OPERATIVA
FACULTAT DE MATEMÀTIQUES
UNIVERSITAT DE VALÈNCIA

Irene Epifanio López

Dirección

Guillermo Ayala Gallego y Jesús Malo López

ABRIL 2002

GUILLERMO AYALA GALLEGO, profesor del Departament d'Estadística i Investigació Operativa de la Universitat de València y

JESÚS MALO LÓPEZ, profesor del Departament d'Òptica de la Universitat de València

CERTIFICAN:

Que la presente memoria, *Descripción de texturas. Aplicaciones a su compresión y clasificación*, ha sido realizada bajo su dirección en el Departament d'Estadística i Investigació Operativa de la Universitat de València por Irene Epifanio López, y constituye su tesis para optar al grado de Doctor por la Universitat de València.

Y para que así conste presentan la referida memoria, firmando el presente certificado en Valencia, a 29 de abril de 2002.

GUILLERMO AYALA GALLEGO

JESÚS MALO LÓPEZ

A la meua benvolguda família

Agradecimientos

En estas líneas quiero agradecer a diversas personas el apoyo personal y/o profesional que desinteresadamente me han prestado, a lo largo del intenso desarrollo de este trabajo.

Primeramente, a mi familia, que han estado ahí (y allá) en todo momento. MIL GRACIAS.

A aquellos con los que compartí los primeros momentos (y muchos más), amigos y compañeros del concurrido laboratorio 2 del departament d'Informàtica de la Universitat de València, Ricardo Ferrís y Elena Díaz, y especialmente a Esther de Ves y Juan Gutiérrez porque son realmente muy buenos amigos.

Als amics que he trobat a Castelló; a aquells que a més de companys del departament de Matemàtiques de la Universitat Jaume I, també et donen la seua amistat i et presten la seua ajuda en qualsevol moment, com Toia Ibáñez.

A Victoria Zapater, por su apreciable colaboración en el capítulo 3.

A Juan Domingo, porque es una EXCELENTE persona. Muchísimas GRACIAS por todo (y bien sabemos que es mucho) lo que has hecho por mí desde el principio, y sobre todo en los últimos tiempos, actuando de 'médico particular' de mi enfermizo ordenador.

A Amelia Simó, perquè és una EXCEL·LENT amiga i companya. Moltes GRÀCIES pels teus savis consells i perquè SEMPRE estàs disposada a escoltar i ajudar.

A mis dos directores. A Jesús Malo, por la MUCHAS cosas que me has enseñado, por las innumerables conversaciones *bañadas de cafés y zumos de naranja*. A Guillermo Ayala, por TANTÍSIMAS cosas ... Porque fue por ti que estoy en este *mundillo* y porque en TODO momento, a lo largo de estos tres años, SIEMPRE me has ayudado ENORMEMENTE en todos los aspectos (que han sido numerosos y variados). INFINITAS GRACIAS, Guillermo, he tenido la INMENSA fortuna de contar con un EXCELENTE profesional y SOBRE TODO con el MEJOR amigo que se pueda tener.

Por último, me gustaría dar las gracias a todas aquellas personas que ponen su trabajo a disposición de los demás (aquellas que desarrollan el R, ofrecen sus librerías, etc.), por su loable labor.

Bien sé que unas simples gracias es poco, porque es de corazón que os estoy muy

II

agradecida. En fin, todo llegará a su debido tiempo...

Índice general

1. Introducción	1
1.1. Una visión general	1
1.2. Análisis de texturas	2
1.2.1. Métodos estadísticos	3
1.2.2. Métodos geométricos	4
1.2.3. Métodos basados en modelos	4
1.2.4. Métodos de procesado de la señal	7
1.3. Organización de la memoria	8
2. Descripción de texturas mediante conjuntos aleatorios	11
2.1. Análisis de texturas bajo la perspectiva de los conjuntos aleatorios	12
2.1.1. Texturas binarias	14
2.1.2. Texturas a niveles de gris	16
2.1.3. Matrices de co-ocurrencia	19
2.2. Métricas usadas para la clasificación de texturas	20
2.3. Experimentos	21
2.3.1. Primer experimento: texturas binarias	22
2.3.2. Segundo experimento: texturas a niveles de gris y método de fragmentación de la intensidad	23
2.3.3. Tercer experimento: texturas a niveles de gris y transformaciones TH	24
2.3.4. Cuarto experimento: ruido	24
2.3.5. Quinto experimento: clasificación de Bayes	25
2.3.6. Sexto experimento: comparación con otras características de texturas	26

3. Una aproximación posibilística para la clasificación de texturas	29
3.1. Metodología	30
3.1.1. Primera aproximación	30
3.1.2. Segunda aproximación	33
3.1.3. Problema de clasificación	35
3.2. Experimentos y evaluación del procedimiento posibilístico	36
3.2.1. Hardening	38
3.2.2. Etiquetas probabilísticas versus posibilísticas : ejemplos sintéticos	41
3.2.3. Uso del método de aleatorización en la comparación de características de texturas	43
4. Transformación lineal para la decorrelación estadística y perceptual en codificación de imágenes	45
4.1. Expresiones matriciales (estadísticas y perceptuales) y su interpretación	46
4.1.1. Matriz de covarianzas	46
4.1.2. Matriz métrica perceptual	46
4.1.3. Interpretación geométrica de estas matrices	47
4.1.4. Cambios matriciales bajo aplicaciones lineales	47
4.2. Modelo visual y matriz métrica perceptual	49
4.3. Diagonalización simultánea de la métrica perceptual W_a y la matriz de covarianzas Σ_a	53
4.4. Resultados de compresión	54
4.4.1. Diseño del cuantizador	54
4.4.2. Detalles de implementación	56
4.4.3. Imágenes decodificadas	57
5. Transformación no lineal para la decorrelación estadística y perceptual	63
5.1. Relaciones estadísticas y perceptuales entre coeficientes	63
5.2. Modelo no lineal y su efecto decorrelador	65
5.3. Método diferencial para la inversión del modelo de normalización no lineal	67
5.4. Comparación entre distintos dominios de representación	72
5.5. Imágenes decodificadas	73

6. Conclusiones y trabajo futuro

77

A. Texturas usadas en los capítulos 2 y 3

81

Capítulo 1

Introducción

1.1. Una visión general

La dificultad de encontrar una base (o un conjunto de características) adecuada sobre la que representar un determinado problema y así facilitar su resolución, es ubicuo, lo encontramos en muchos campos de la ciencia. Muchos problemas que se plantean en el área del análisis de imágenes, conllevan la extracción de ciertas características de la imagen, de manera que una representación apropiada de las imágenes (dependiendo del objetivo del problema) juega un papel primordial. Es evidente que problemas distintos necesitan abordarse de maneras diferentes, según el objetivo del problema, éste podría exigir diferentes representaciones. Así, en el caso de un problema de extracción de características [1], se requieren tratamientos distintos de acuerdo a cual sea el propósito final: por ejemplo, la representación de la señal o la clasificación. En el primer caso, seleccionaríamos las características que mejor resumieran la información, mientras que en el segundo, buscaríamos aquellas características que mejor discriminaran entre las clases. Siguiendo con esta distinción, fijémonos en el gráfico 1.1. En dicho gráfico, se muestra la distribución (bivariante) de dos grupos, A y B (por ejemplo, podría tratarse de la distribución de peso y altura para cada sexo). A la hora de representar los vectores de esta distribución (bien para cada grupo o su mixtura), el eje principal ϕ_1 con un mayor valor propio (primera componente principal) sería una mejor elección que ϕ_2 . Esto es debido a que ϕ_1 produciría un menor error cuadrático medio de la representación que la selección ϕ_2 . En cambio, si nuestro objetivo fuera la discriminación entre ambas clases, es evidente que la mejor elección sería ϕ_2 , ya que conserva mejor la información sobre clasificación. Fijémonos que si proyectásemos sobre ϕ_1 ambas distribuciones, éstas aparecerían fuertemente solapadas. Si por contra las proyectásemos sobre ϕ_2 , las densidades marginales estarían bien separadas, sin apenas solapamiento.

Es indudable que establecer un buen modelo para representar las imágenes no es sencillo (aunque los sistemas visuales biológicos resuelven bien el problema), debido a la infinidad de posibles imágenes con las que podemos encontrarnos en nuestras vidas. Para

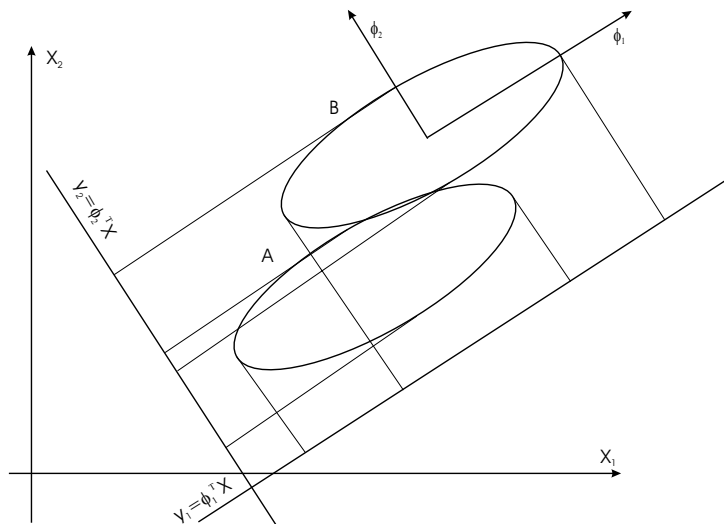


Figura 1.1: Ejemplo sobre la diferencia entre representación de la señal y clasificación

poder trabajar con esta clase de imágenes, conocidas como imágenes naturales [2], suelen asumirse ciertas hipótesis de trabajo [3]: suposición de Markov (localidad), en el sentido de que el valor en un píxel (picture element, elemento de imagen) sólo depende de su vecindad local y suposición de estacionariedad, por la que el valor de un píxel no dependería de su localización absoluta, sino de su posición espacial relativa en la imagen.

Por supuesto, la clase de imágenes que comúnmente se llaman texturas, parecen ser más consistentes con esos modelos locales y estacionarios. A pesar de ser un concepto ampliamente utilizado y que intuitivamente parece bastante claro, no existe una definición precisa sobre qué es una textura. Son 'fáciles' de reconocer, pero 'difíciles' de definir. Prueba de ello es el gran número de definiciones existentes (en [4] existe un extenso catálogo), así como el amplio abanico de métodos que abordan el problema del análisis de texturas. Si tratamos de dar una definición de textura, podríamos decir que se trata de *algo formado por elementos mutuamente relacionados* [5], con lo cual estaríamos considerando un conjunto de píxeles (primitiva de la textura) y también dependería en gran medida de la escala (número de elementos considerados) [6]. En [7] podemos encontrar una excelente explicación acerca de lo que se considera *textura*. Como ejemplo, en el apéndice se muestran las texturas que se analizarán en este trabajo.

1.2. Análisis de texturas

Tradicionalmente, existen dos aproximaciones principales para la descripción de texturas: la estadística y la sintáctica [6], aunque por supuesto, esta distinción no deja de

ser algo artificial. En lo que respecta a la primera aproximación, la textura se describe mediante un vector de propiedades que representa la textura como un punto en un espacio de características multidimensional [5]. En cuanto a la segunda aproximación, se trata de determinar las primitivas que constituyen la textura y sus reglas de colocación. Por otra parte, también existen métodos basados en la detección de *textones* [8, 9, 10].

El problema del análisis de texturas, ha sido ampliamente estudiado y como consecuencia son numerosas y diversas las aportaciones realizadas en este campo. A continuación, vamos a revisar algunas (por supuesto, muchas se quedarán en el tintero, pues no se pretende realizar una enumeración exhaustiva) de las características más usadas en el análisis de texturas, y aunque existen relaciones entre ellas, trataremos de clasificar los métodos en cuatro grandes grupos [5, 11]: métodos estadísticos, geométricos, basados en modelos y de procesamiento de la señal. En concreto, prestaremos mayor atención a aquellas características que después emplearemos en diversos experimentos.

1.2.1. Métodos estadísticos

Estos métodos tienen en cuenta la distribución espacial de los niveles de gris y entre los más conocidos cabe citar: las matrices de co-ocurrencia y la función de autocorrelación. Denotemos mediante $\{f(x, y) : 0 \leq x \leq N - 1, 0 \leq y \leq M - 1\}$, una imagen $N \times M$ con G niveles de gris.

1. Matrices de co-ocurrencia

Una matriz de co-ocurrencia P_d para el vector de desplazamiento d se define como sigue. El elemento (i, j) de P_d es el número de ocurrencias del par de niveles de gris i y j que están separados por el vector d , $P_d(i, j) = \text{Card}\{(s, r) : r - s = d, f(s) = i, f(r) = j\}$. Así definida, esta matriz no sería simétrica, pero bastaría con considerar $P = P_d + P_{-d}$. Para diversas matrices, se calculan distintos descriptores, algunos de los más usuales (y que posteriormente se utilizarán en diversas comparativas) son:

a) Energía:

$$\sum_i \sum_j P_d^2(i, j). \quad (1.1)$$

b) Entropía:

$$-\sum_i \sum_j P_d(i, j) \log_2 P_d(i, j). \quad (1.2)$$

c) Momento de diferencia inversa:

$$\sum_{i,j} \frac{P_d(i, j)}{1 + (i - j)^2}. \quad (1.3)$$

d) Inercia:

$$\sum_{i,j} (i - j)^2 P_d(i, j). \quad (1.4)$$

e) Correlación:

$$\sum_i \sum_j \frac{(i - \mu_1)(j - \mu_2)P_d(i, j)}{\sigma_1 \sigma_2}. \quad (1.5)$$

siendo, μ_1 , μ_2 , σ_1 , σ_2 ,

$$\begin{aligned} \mu_1 &= \sum_i i \sum_j P_d(i, j) \\ \mu_2 &= \sum_j j \sum_i P_d(i, j) \\ \sigma_1 &= \sum_i (i - \mu_1)^2 \sum_j P_d(i, j) \\ \sigma_2 &= \sum_j (j - \mu_2)^2 \sum_i P_d(i, j) \end{aligned}$$

2. Función de autocorrelación

La función de autocorrelación de una imagen puede usarse para evaluar tanto la regularidad como la finura/tosquedad de la textura. La función de autocorrelación en (x, y) es:

$$\rho(x, y) = \frac{NM \sum_{l=0}^{N-x-1} \sum_{k=0}^{M-y-1} f(l, k) f(l+x, k+y)}{(N-x)(M-y) \sum_{l=0}^{N-1} \sum_{k=0}^{M-1} f^2(l, k)} \quad (1.6)$$

La función de autocorrelación, podría determinarse también en el dominio frecuencial, a través del espectro de la transformada de Fourier.

1.2.2. Métodos geométricos

Los métodos contemplados en este apartado consideran la textura compuesta de 'elementos' o primitivas. Tras identificar estos elementos, la textura puede analizarse bajo dos perspectivas: computar propiedades estadísticas de los elementos extraídos o bien extraer la regla de colocación que describa la textura, esta última aproximación puede conllevar métodos geométricos o sintácticos. Dentro de la primera aproximación, podría mencionarse [12] donde se propone la extracción de *tokens* usando las propiedades de la partición de Voronoi de una imagen. En cuanto a la segunda aproximación, en [5] o [11] por ejemplo, se repasan algunos métodos estructurales. No nos extenderemos más en esta clase de métodos, puesto que no los utilizaremos en este trabajo.

1.2.3. Métodos basados en modelos

Estos métodos se basan en la construcción de un modelo, cuyos parámetros estimados describirían las cualidades de la textura. En esta sección consideraremos los campos aleatorios de Markov así como los fractales.

1. Campos aleatorios de Markov

Los métodos que utilizan los campos aleatorios de Markov gaussianos (Gaussian Markov Random Fields) caracterizan la relación estadística entre un píxel y sus vecinos [13]. El modelo estocástico resultante consta de un número de parámetros igual al tamaño de la máscara de la vecindad. Los parámetros se pueden estimar mediante un algoritmo de mínimos cuadrados sobre cada posición de la máscara en la imagen. Una máscara que es bastante habitual es una máscara simétrica, como la que se muestra en la figura siguiente (es la que hemos empleado en los experimentos). Veamos más detalladamente en que consistiría este método. Sea $f(s)$ [$s = (x, y)$]

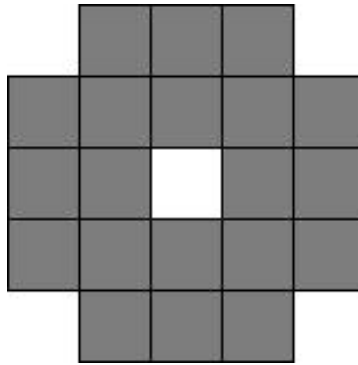


Figura 1.2: Máscara simétrica que define la vecindad

una imagen con media cero y supongamos que los valores de los píxeles siguen una distribución normal. Sea $\Phi = \{(a, b) \dots\}$ el conjunto que define la vecindad, donde cada elemento indicaría la posición relativa respecto del píxel actual, es decir, $(0, 1)$ indicaría el píxel $f(x, y + 1)$ de la imagen. Supongamos que los píxeles de la imagen se encuentran relacionados de la siguiente manera

$$f(s) = \sum_{r=(x,y) \in \Phi} \phi_r f(s+r) + e(s), \quad (1.7)$$

donde $e(s)$ sigue una distribución normal. Si usamos una máscara simétrica y permitimos invarianza por rotación 180° , el número de parámetros se reduce a la mitad, ya que $f(s+r)$ y $f(s-r)$ compartirían parámetro:

$$f(s) = \sum_{r=(x,y) \in \Phi_{sim}} \phi_r (f(s+r) + f(s-r)) + e(s). \quad (1.8)$$

Los parámetros del modelo se pueden obtener resolviendo para ϕ en las ecuaciones 1.7 o 1.8, por medio de un método de mínimos cuadrados como el detallado en [13].

2. Fractales

El análisis de texturas basándose en fractales fue introducido por Pentland [14],

donde se mostró la correlación existente entre la dimensión fractal de una textura y su 'tosquedad' (coarseness). La descripción fractal de texturas se basa en gran medida, en la determinación de la dimensión fractal. La propiedad de auto-similaridad implica que la dimensión fractal de una imagen es independiente de la escala. Un conjunto acotado A es auto-similar, si es la unión de N_r copias no solapadas de un conjunto similar a A , pero escalado por un factor r . A partir de esta definición, la dimensión fractal viene dada por

$$D = \frac{\log(N_r)}{\log(r^{-1})}. \quad (1.9)$$

La dimensión fractal la podemos aproximar si estimamos N_r para varios valores de r y entonces determinamos la pendiente del ajuste lineal por mínimos cuadrados de $\log(N_r)$ versus $\log(r^{-1})$. Para ello, podemos emplear el método *differential box-counting* explicado en [15]. Consideremos una imagen $M \times M$, como una superficie en \mathbb{R}^3 , (x, y) representaría la posición del píxel y z la intensidad. Particionemos el espacio (x, y) en una retícula de tamaño $s \times s$ píxeles. Una estimación de la escala relativa es $r = s/M$. Si denotamos por k y l los valores mínimo y máximo respectivamente, para la imagen $s \times s$ en la posición del retículo (p, q) , podemos definir $n_r(p, q) = l - k + 1$ y N_r lo estimaremos mediante $N_r = \sum_{p,q} n_r(p, q)$. Este procedimiento se repite para varios valores de r (s), representaríamos $\log(N_r)$ frente a $\log(r^{-1})$ y estimaríamos la dimensión fractal como la pendiente de la recta que ajusta estos puntos.

En los experimentos posteriores, se ha extraído la dimensión fractal tanto para la imagen original ($f_1(x, y)$) como para dos transformaciones de la misma ($f_2(x, y)$, $f_3(x, y)$), como en [15]. Veamos como se definen estas dos imágenes.

$$f_2(x, y) = \begin{cases} f_1(x, y) - L_1 & \text{si } f_1(x, y) > L_1 \\ 0 & \text{en otro caso} \end{cases} \quad (1.10)$$

donde $L_1 = g_{min} + \frac{g_{med}}{2}$, con g_{min} y g_{med} el valor mínimo y medio de $f_1(x, y)$.

$$f_3(x, y) = \begin{cases} 255 - L_2 & \text{si } f_1(x, y) > (255 - L_2) \\ f_1(x, y) & \text{en otro caso} \end{cases} \quad (1.11)$$

donde $L_2 = g_{max} - \frac{g_{med}}{2}$, con g_{max} y g_{med} el valor máximo y medio de $f_1(x, y)$.

Además, se ha extraído una cuarta característica: la dimensión multifractal. Ésta se define como

$$D(2) = \lim_{r \rightarrow 0} \frac{\log \sum_{i,j} n_r'^2}{\log r}, \quad (1.12)$$

siendo $n_r' = \frac{n_r}{N_r}$. Para estimar $D(2)$, usamos distintos valores de r y la regresión lineal de $\log \sum_{i,j} n_r'^2$ versus $\log r$.

Todas las características se han normalizado para que estén comprendidas entre 0 y 1.

1.2.4. Métodos de procesado de la señal

Las técnicas anteriores operan sobre la señal definida en el dominio espacial. Sin embargo, puede resultar más conveniente representar la imagen en otro dominio, de forma que en este nuevo dominio se facilite la extracción de determinadas características. Muchas de las técnicas de este apartado consisten en calcular ciertas características de las imágenes tras haber sido filtradas, además algunas se apoyan en el hecho de que en una primera etapa, el cerebro humano lleva a cabo un análisis frecuencial de la imagen. Dividiremos esta sección en tres apartados.

1. Filtros en el dominio espacial

Dentro de esta categoría podemos incluir desde métodos que miden la densidad de bordes a través de la magnitud de las respuestas de máscaras de Robert y laplaciana [16, 17], hasta métodos basados en momentos espaciales [17] y otros obtenidos mediante filtros espaciales y operadores no lineales [18].

2. Filtros en el dominio de Fourier

Siguiendo los resultados psicovisuales, se han desarrollado sistemas de análisis de texturas de filtrado en el dominio de Fourier para obtener distintas características. Cada filtro es selectivo tanto en frecuencia como en orientación [19].

3. Filtros Gabor y ondículas

La transformada de Fourier proporciona un análisis de la frecuencia global. Hay veces que es más apropiado que el análisis esté localizado en el dominio espacial. Una forma clásica de conseguirlo, es a través de la transformada de Fourier ventana. Cuando la función ventana es una gaussiana, la transformación es una función de Gabor [20]. Una función de Gabor bidimensional [21] definida en el dominio espacial ($\mathbf{x} = (x, y)$), viene dada por el producto de dos funciones:

- Una función armónica bidimensional cuya oscilación se extiende perpendicularmente al vector $\mathbf{f}_0 = (f_{0x}, f_{0y})$ con frecuencia $|\mathbf{f}_0|$, con origen en un determinado $\mathbf{x}_0 = (x_0, y_0)$ y fase ϕ .
- Una gaussiana de base elíptica centrada en el punto \mathbf{x}_0 y cuyo eje mayor forma un ángulo θ con el eje X .

$$G(\mathbf{x}, \mathbf{x}_0, \mathbf{f}_0, \sigma_x, \sigma_y, \theta, \phi) = e^{-\pi\left(\frac{x'^2}{\sigma_x^2} + \frac{y'^2}{\sigma_y^2}\right)} e^{i(2\pi\mathbf{f}_0(\mathbf{x}-\mathbf{x}_0)+\phi)}, \quad (1.13)$$

donde $\mathbf{x}' = (x', y')$ resulta de una traslación y un giro de ejes, es decir,

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos(-\theta) & \text{sen}(-\theta) \\ -\text{sen}(-\theta) & \cos(-\theta) \end{pmatrix} \cdot \begin{pmatrix} x - x_0 \\ y - y_0 \end{pmatrix}. \quad (1.14)$$

En particular, introduciendo la metodología que seguiremos en los experimentos, podemos construir diversos filtros de Gabor variando la longitud de onda (λ), la

orientación (θ), la fase (ϕ) y las desviaciones típicas de la gaussiana (σ), usando:

$$G(x, y | \lambda, \theta, \phi, x_0, y_0) = \exp\left(\frac{(x - x_0)^2 + (y - y_0)^2}{-2\sigma^2}\right) \operatorname{sen}\left(\frac{2\pi}{\lambda}(x \cos \theta - y \operatorname{sen} \theta) + \phi\right)$$

La propuesta de usar filtros de Gabor en el análisis de texturas fue realizada por Turner [22] y Clark y Bovik [23]. Además de los filtros de Gabor, las ondículas (*wavelets*) [24] también se han empleado con éxito en el análisis de texturas. Algunos de los numerosos trabajos que utilizan las transformadas de Gabor o las ondículas son [25, 26, 27, 28].

Las texturas han encontrado multitud de aplicaciones en el área del proceso de imagen. Los métodos de análisis de texturas podemos usarlos en problemas de clasificación, segmentación, síntesis (muy popular en gráficos por computador), compresión, análisis de forma (para extraer información de la superficie), recuperación de imágenes en una base de datos, etc. [7]. Las posibilidades de aplicación son enormes, desde aplicaciones industriales hasta en el campo médico. Existen aplicaciones en inspección de la calidad en industrias textiles y automovilísticas ([29, 30, 31, 32, 33, 34, 35]), multitud de aplicaciones médicas ([36, 37, 38, 39, 40, 41, 42, 43]), en proceso de documentos ([44, 45, 46, 47, 48, 25]), recuperación de imágenes de una base de imágenes ([49]) y por supuesto en teledetección ([50, 51, 52, 53, 54, 55]), entre otras.

1.3. Organización de la memoria

En este trabajo, nos hemos centrado en el problema de la clasificación de texturas y en el de la representación eficiente de imágenes naturales (constituidas por múltiples texturas), para su compresión en el contexto de *transform coding*. Ambos problemas comprenden dos etapas similares. Una primera etapa de extracción de características: por un lado, descriptores de texturas y por otro lado, coeficientes obtenidos tras la transformación de la imagen a un dominio adecuado. Una segunda etapa de determinación (asignación) de pertenencias: por una parte, proceso de clasificación y por otra, proceso de cuantizado que conlleva la asignación de un cierto representante según la zona en que se encuentre nuestra entrada. A continuación, se expone la estructura de esta memoria.

En los capítulos 2 y 3 trataremos el problema de la clasificación de texturas. El problema de clasificación de texturas consiste en decidir a qué clase pertenece una textura dada. Para llevarlo a cabo, se hace necesario conocer a priori las clases que pueden reconocerse, es decir, hemos de definir las categorías. Esto suele hacerse, dando un conjunto de prototipos que definen cada clase. Una vez conocemos las categorías, se extraen una serie de características que se utilizan en la fase de clasificación. El capítulo 2 lo podríamos incluir en la fase primera del problema, la de la extracción de características. Bajo la perspectiva de los conjuntos aleatorios, propondremos unos determinados descriptores. El capítulo 3, en cambio, lo podríamos incluir en la etapa segunda, la del proceso de clasificación. En este

capítulo, propondremos unas etiquetas posibilísticas que analizaremos en el problema de la clasificación de texturas. Si c denota el número de clases que tenemos, una etiqueta posibilística [56] sería un vector en \mathbb{R}^c , cuyas componentes estarían en el intervalo $[0, 1]$, pero sin la restricción propia de las etiquetas probabilísticas de que la suma de las componentes sea 1. Así, la componente i de la etiqueta posibilística nos daría el grado de pertenencia (o tipicalidad) de la textura a clasificar en la clase i . Si en lugar de una clasificación difusa, se requirieran unas etiquetas binarias (*crisp*) (una de las componentes vale uno y el resto son nulas), podríamos obtenerlas mediante un proceso de *hardening* (por ejemplo asignando a la clase dominante).

En los capítulos 4 y 5, abordaremos el problema de encontrar una representación adecuada de una imagen para la compresión de imágenes estáticas, en el contexto de *transform coding* [57]. Este esquema de compresión, podría resumirse básicamente en los siguientes pasos: la imagen se segmenta en pequeños bloques que sufren una determinada transformación, obteniéndose los coeficientes transformados; estos coeficientes se cuantizan individualmente (con un número de bits acorde con su significado) y por último, se codifican. Si únicamente se contemplara la función de densidad de la señal (factor estadístico), el problema de encontrar una base apropiada, que representara óptimamente las características de la imagen, consistiría en conseguir factorizar dicha función de densidad. Por supuesto, si nos enfrentásemos a un proceso gaussiano, el problema vendría resuelto por la clásica transformación Karhunen-Loève (KLT) o análisis de componentes principales (PCA) [2, 58], que se reduce al problema de conocer los valores propios de la matriz de covarianzas (momento de segundo orden). Recientemente, se ha propuesto otro análisis que contempla momentos de orden superior, el llamado análisis de componentes independientes (ICA) [2, 59, 60]. En muchos casos prácticos, en especial aquellos que tratan con imágenes naturales, se utilizan transformaciones fijas como la transformada discreta de coseno (DCT) o las ondículas (wavelets) por su gran similitud con las soluciones PCA o ICA, debido a las características de las imágenes naturales [61, 59, 60, 62, 63].

Sin embargo, existen problemas en los que considerar únicamente el factor estadístico no es suficiente. Este es el caso de problemas que involucran a las imágenes naturales percibidas por los humanos, como son las aplicaciones multimedia. Aquí, las dimensiones del espacio no significan lo mismo, el espacio es 'altamente' no euclídeo debido a las propiedades particulares de la percepción humana [64, 65, 66, 67]: no todas las escalas, texturas o componentes de color tienen la misma relevancia en el sistema visual humano (HVS).

Puesto que en *transform coding*, la imagen ha de representarse en un espacio donde sus componentes puedan cuantizarse escalarmente [68], el objetivo de la transformación tiene que ser la eliminación de las relaciones entre los coeficientes de la imagen, para que de esta manera se pueda llevar a cabo una cuantización escalar en toda regla. Así pues, la representación adecuada sería aquella que eliminara los dos tipos de redundancias presentes en las imágenes [69]: la estadística y la psicovisual. La redundancia estadística procede de la relación existente entre los píxeles, con lo cual no serían estadísticamente independientes (la intensidad de un píxel podría *adivinarsse* a partir de las intensidades de sus vecinos).

La redundancia psicovisual se basa en las características (fenómenos de enmascaramiento [69]) de la percepción visual humana. En el sistema visual humano, la información visual no se percibe de igual manera, sino que alguna información podría ser más importante que otra. En este sentido, alguna información es perceptualmente redundante.

La aproximación estándar al *transform coding* únicamente tiene en cuenta cuando busca la representación adecuada, el factor estadístico (a través de la matriz de covarianzas) [68]. Los factores psicovisuales se introducen empíricamente *después* de la selección de la representación a través de un simple pesado escalar en el dominio transformado [70, 57, 69].

En este trabajo tendremos en cuenta tanto los factores estadísticos como los perceptuales en la búsqueda de la representación. Para ello, se definirá una métrica perceptual que describirá la no uniformidad del espacio imagen desde el punto de vista perceptual [66, 71].

En el capítulo 4, se propondrá una transformación lineal, que contemplará ambos aspectos, los estadísticos y perceptuales por medio de la matriz de covarianzas y la matriz métrica perceptual. En este capítulo, asumiremos una única matriz métrica perceptual para todo el dominio. En base a los elipsoides que definen ambas matrices, especificaremos una transformación lineal que diagonalizará simultáneamente ambas matrices, eliminando de esta manera las relaciones (estadísticas y perceptuales) de segundo orden entre los coeficientes.

En el capítulo 5, en cambio, sí que trabajaremos con matrices métricas perceptuales dependientes de la imagen de entrada. Sugeriremos el uso de una transformación no lineal, inspirada en un modelo de percepción, con la que conseguiremos un dominio 'perceptualmente' euclídeo y en el que además las relaciones estadísticas entre los coeficientes serán reducidas. Por otro lado, se propondrá también un método de integración numérica de ecuaciones diferenciales para su inversión.

Para finalizar, en el capítulo 6 se presentarán las conclusiones de este trabajo y se aportarán algunas ideas sobre las que se podría trabajar en el futuro.

Capítulo 2

Descripción de texturas mediante conjuntos aleatorios

¹ En este capítulo se propondrá la representación de texturas considerándolas como realizaciones de conjuntos aleatorios y se verá su aplicación a la clasificación de las mismas. En concreto, consideraremos que una textura binaria es la realización de un conjunto aleatorio, con lo cual diversos descriptores propios de estos modelos (funciones \mathcal{K} y distribuciones de contacto esféricas y lineales) nos permitirán describir la textura y clasificarla. Por otro lado, si en lugar de contar con una textura binaria, disponemos de una textura a niveles de gris que hemos de clasificar, esta textura la podremos reducir a un conjunto de imágenes binarias. Cada una de estas imágenes (que en su conjunto resumen la textura original), corresponderían con aquellos píxeles que verificaran cierta propiedad local. De esta forma, la textura original se reduciría a un conjunto aleatorio multivariante, cuyas componentes (cada conjunto aleatorio) serían cada una de las imágenes binarias (indicadoras del cumplimiento de una determinada propiedad). Análogamente al caso univariante, podrían emplearse descriptores propios de los conjuntos aleatorios multivariantes para representar las texturas y proceder a su clasificación. En nuestro caso, se han usado las funciones \mathcal{K} marginales y cruzadas, así como las distribuciones de contacto esféricas y lineales, tanto marginales como cruzadas. A grandes rasgos, el esquema del capítulo será el siguiente. En primer lugar, se recordará la definición de conjunto aleatorio, así como algunos de sus descriptores que se emplearán en la descripción de las texturas para su posterior clasificación y se establecerá la conexión existente con las conocidas matrices de co-ocurrencia. En segundo lugar, se expondrán los procedimientos de clasificación que se han utilizado y por último se presentarán los resultados obtenidos con diversos experimentos.

¹El presente capítulo constituye básicamente el artículo '*A Random Set View of Texture Classification*', aceptado para su publicación en IEEE Transactions on Image Processing [72].

2.1. Análisis de texturas bajo la perspectiva de los conjuntos aleatorios

Recordemos en primer lugar la definición de conjunto aleatorio cerrado en el espacio euclídeo \mathbb{R}^2 . Supongamos que \mathcal{F} es la familia de todos los subconjuntos cerrados de \mathbb{R}^2 . Denotemos por \mathcal{A} la σ -álgebra generada por los conjuntos $\{F \in \mathcal{F} : F \cap K \neq \emptyset\}$ donde K es cualquier subconjunto compacto de \mathbb{R}^2 .

Definición 1 *Un conjunto aleatorio cerrado X es una aplicación medible que toma valores en $(\mathcal{F}, \mathcal{A})$. Por tanto, X es una aplicación $(\mathcal{A}, \mathcal{F})$ -medible de un espacio de probabilidad $(\Omega, \mathcal{A}, \mathbf{P})$ en $(\mathcal{F}, \mathcal{A})$. De esta manera, si P denota la distribución generada sobre $(\mathcal{F}, \mathcal{A})$, entonces la terna $(\mathcal{F}, \mathcal{A}, P)$ es un conjunto aleatorio cerrado.*

A partir de ahora los conjuntos aleatorios cerrados los llamaremos simplemente conjuntos aleatorios. Si Φ es un conjunto aleatorio, es decir, un elemento aleatorio del anterior espacio de probabilidad, entonces $\{\Phi \cap K \neq \emptyset\}$ es un suceso aleatorio, con K siendo un conjunto compacto arbitrario de \mathbb{R}^2 .

Un conjunto aleatorio multivariante se define como $\Phi = (\mathcal{F} \otimes \dots \otimes \mathcal{F}, \mathcal{A} \otimes \dots \otimes \mathcal{A}, P)$ donde $(\mathcal{F} \otimes \dots \otimes \mathcal{F}, \mathcal{A} \otimes \dots \otimes \mathcal{A})$ denota el producto de los espacios $(\mathcal{F}, \mathcal{A})$ por sí mismos p veces y P es una medida de probabilidad en dicho espacio producto. Así pues, $\Phi = (\Phi_1, \dots, \Phi_p)$ donde Φ_1, \dots, Φ_p son conjuntos aleatorios univariantes, las componentes de Φ . De nuevo, a partir de la definición de conjunto aleatorio multivariante, los conjuntos básicos $\{\Phi_1 \cap K_1\} \cap \dots \cap \{\Phi_p \cap K_p\}$ son sucesos aleatorios, con lo cual tiene sentido plantearse la probabilidad de los mismos.

Un estudio mucho más detallado puede encontrarse en [73], [74] y [75]. Por otro lado, en [76, 77, 78] podemos encontrar otras aplicaciones de conjuntos aleatorios multivariantes.

Vista ya la definición de conjunto aleatorio, presentemos el marco de análisis de texturas utilizándolos como modelos. Antes de proseguir, notemos que la teoría presentada se refiere a un espacio euclídeo, sin embargo después trabajaremos con imágenes digitales, es decir, en un espacio discreto. Ahora no contemplaremos el proceso de discretización, pero a la hora de proponer los estimadores de los descriptores ya se trabajará con píxeles. Empecemos con las texturas binarias.

La figura 2.1(a) muestra una textura binaria obtenida por la umbralización de la textura original a niveles de gris que aparece en la figura 2.1(b). Una textura binaria puede identificarse con el conjunto de píxeles de valor 1. A partir de la imagen anterior, podemos considerar una textura binaria como la realización de un conjunto aleatorio.

Para texturas a niveles de gris, podrían considerarse dos alternativas. La primera sería identificar la imagen $(\{f(x) : x \in \mathbb{R}^2\})$ con su subgrafo $(X(f) = \{(x, t) : 0 \leq t \leq f(x)\})$ ([79, 80]) y estudiar la imagen como un conjunto aleatorio en \mathbb{R}^3 . La segunda posibilidad, que es la que se ha estudiado, consiste en convertir la textura a niveles de gris en una serie de imágenes binarias, donde cada una de las cuales se corresponderá con una propiedad

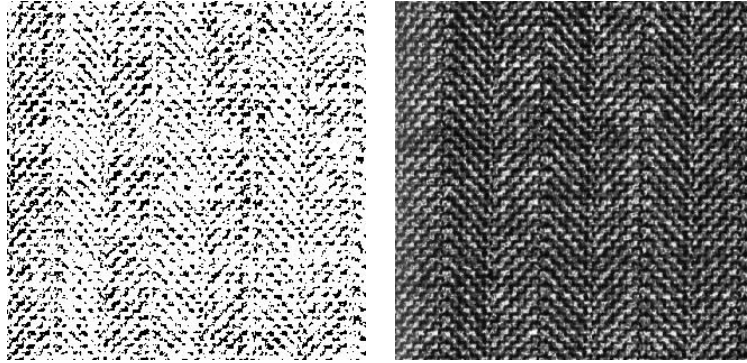


Figura 2.1: Textura D17 de la base de texturas de Brodatz (b) y su versión umbralizada (a)

local o tipo de estructura diferente. Cada una de estas imágenes binarias puede verse como una realización de un conjunto aleatorio y, por tanto, si las consideramos conjuntamente serían la realización de un conjunto aleatorio multivariante. En consecuencia, la textura original se reduce a un conjunto aleatorio multivariante.

Notemos que no se asumirá ningún modelo paramétrico en el análisis que presentamos. Algunos modelos paramétricos de conjuntos aleatorios son los modelos booleanos y los modelos germen-grano ([81, 74]). En [82], [83] y [84] se usó un modelo booleano unidimensional para analizar texturas. Como ya se ha dicho, en este trabajo se adoptará una aproximación no paramétrica. En aplicaciones concretas donde cada textura pueda ser modelada mediante un modelo paramétrico, la clasificación final se podría mejorar al emplear dicho modelo paramétrico. Sin embargo, como no podemos suponer que las texturas que consideraremos sigan un determinado modelo, se ha optado por la aproximación no paramétrica.

Cada textura se describirá mediante los descriptores comunes de estos modelos estocásticos: fracciones de volumen, funciones \mathcal{K} marginales y cruzadas y distribuciones de contacto esféricas y lineales. Estas características son las que se usarán en el proceso de clasificación. La figura 2.2 muestra el esquema que se seguirá para la clasificación de texturas.



Figura 2.2: Metodología

Veamos con detalle cada uno de los descriptores empleados: sus definiciones formales y sus estimadores. Para ello, trataremos por separado el caso de las texturas binarias

y a niveles de gris. En ambos casos, asumiremos que estamos tratando con conjuntos aleatorios estacionarios, es decir, con distribución invariante por traslación. Esta suposición tiene sentido puesto que bajo el concepto de textura subyace la idea de estacionaridad (al menos en las texturas homogéneas), o sea, si se observaran distintas subimágenes de una textura, éstas se parecerían aunque no serían iguales: serían distintas realizaciones de un mismo modelo estocástico.

2.1.1. Texturas binarias

Los descriptores que revisaremos brevemente son: la fracción de volumen, la función de covarianza, la \mathcal{K} -función y las distribuciones de contacto lineales y esféricas.

La fracción de volumen se define como

$$p = P(0 \in \Phi), \quad (2.1)$$

y se puede interpretar como el área media cubierta por unidad de área.

La función de covarianza se define como

$$C(h) = P(0, h \in \Phi) \text{ con } h \in \mathbb{R}^2, \quad (2.2)$$

es decir, sería la probabilidad de que un punto y su trasladado por h pertenezcan al conjunto.

La \mathcal{K} -función de Ripley es

$$\mathcal{K}(t) = \frac{1}{p^2} \int_{B(0,t)} C(h) dh, \quad (2.3)$$

donde $B(0, t)$ denota la bola centrada en el origen de radio t . $p\mathcal{K}(t)$ se puede interpretar como el área media cubierta por el conjunto Φ en una distancia t desde un punto de Φ arbitrariamente elegido.

Sea D la distancia (aleatoria) desde un punto arbitrario al conjunto Φ . Podemos definir la distribución de contacto esférica, si condicionamos la elección del punto a que pertenezca al complementario de Φ , Φ^c . Así tendremos,

$$H^{(s)}(t) = P(D \leq t \mid 0 \in \Phi^c). \quad (2.4)$$

Análogamente, la distribución de contacto lineal de ángulo α , $H^{(l,\alpha)}$, es la función de distribución acumulada de la variable aleatoria, $D^{(\alpha)}$, que se define como la distancia desde un punto arbitrario de Φ^c al punto más cercano de Φ en la línea que pasa por el punto y con orientación α . Obviamente, se obtienen distintas funciones para distintas orientaciones α .

Las características que se emplearán en la clasificación de texturas son los estimadores de las cantidades anteriores que a continuación se presentan.

Denotemos mediante W el retículo de píxeles y por ϕ aquellos píxeles que valen uno. Nótese que $W \cap \phi = \phi$. La fracción de volumen se estimará con

$$\hat{p} = \frac{\nu(\phi)}{\nu(W)}, \quad (2.5)$$

donde ν es el área entendida como el número de píxeles, o sea, $\nu(W)$ es el número total de píxeles. El estimador de la función de covarianza en el punto h es

$$\hat{C}(h) = \frac{\nu(\{x \in \phi : x + h \in \phi\})}{\nu(\{x \in W : x + h \in W\})}. \quad (2.6)$$

Recordemos que si X y T son dos conjuntos, se define la erosión de X mediante T ([85]) como

$$\epsilon_T(X) = \{x \in X : T + x \subset X\}, \quad (2.7)$$

donde $T + x$ es el trasladado de T por x ($T + x = \{t + x : t \in T\}$). Señalemos que

$$\hat{C}(h) = \frac{\nu(\epsilon_{\{0,h\}}(\phi))}{\nu(\epsilon_{\{0,h\}}(W))}. \quad (2.8)$$

\mathcal{K} se ha estimado integrando numéricamente \hat{C} , teniendo en cuenta que

$$\mathcal{K}(t) = \frac{1}{p^2} \int_{-t}^t \int_{-\sqrt{t^2-h_1^2}}^{\sqrt{t^2-h_1^2}} C(h_1, h_2) dh_2 dh_1. \quad (2.9)$$

La función

$$G(h_1) = \int_{-\sqrt{t^2-h_1^2}}^{\sqrt{t^2-h_1^2}} C(h_1, h_2) dh_2, \quad (2.10)$$

se ha estimado con la regla extendida de Simpson (ecuación 4.1.13, página 134 de [86]). Sea \hat{G} la correspondiente estimación. Finalmente, la integral más externa (con respecto a h_1) se ha estimado también aplicando la regla extendida de Simpson. En las páginas 162-163 de [86] pueden encontrarse los detalles precisos.

Para la distribución de contacto esférica, se ha empleado el siguiente estimador

$$\hat{H}^{(s)}(t) = 1 - \frac{\nu(\{x \in \phi^c : B(0, t) + x \subset \phi^c\}) \nu(W)}{\nu(\{x \in W : B(0, t) + x \subset W\}) \nu(\phi^c)} = 1 - \frac{\nu(\epsilon_{B(0,t)}(\phi^c)) \nu(W)}{\nu(\epsilon_{B(0,t)}(W)) \nu(\phi^c)}, \quad (2.11)$$

donde ϕ^c denota el complementario de ϕ con respecto a W . Similarmente, la distribución de contacto lineal con orientación α se ha estimado con

$$\hat{H}^{(l,\alpha)}(t) = 1 - \frac{\nu(\{x \in \phi^c : S(t, \alpha) + x \subset \phi^c\}) \nu(W)}{\nu(\{x \in W : S(t, \alpha) + x \subset W\}) \nu(\phi^c)} = 1 - \frac{\nu(\epsilon_{S(t,\alpha)}(\phi^c)) \nu(W)}{\nu(\epsilon_{S(t,\alpha)}(W)) \nu(\phi^c)}, \quad (2.12)$$

donde $S(t, \alpha)$ es un segmento de longitud t , orientación α y con extremo izquierdo en el origen 0. Los cuatro α -valores que se usarán en el apartado 2.3 son: 0, $\frac{\pi}{4}$, $\frac{\pi}{2}$ y $\frac{3\pi}{4}$ radianes.

Un estudio mucho más detallado de los estimadores vistos en las ecuaciones 2.5, 2.6, 2.11 y 2.12 podemos encontrarlo en [74], capítulo 6 págs. 211-215, donde se proporcionan algunas propiedades teóricas junto con sus correspondientes referencias.

La \mathcal{K} -función y las distribuciones de contacto esféricas y lineales (con distintas orientaciones) se estimarán en un conjunto discreto que se especificará en la sección de experimentos. Las estimaciones de dichas funciones constituirán los descriptores asociados a una textura binaria.

2.1.2. Texturas a niveles de gris

Las texturas a niveles de gris se analizarán análogamente a las texturas binarias. La idea es simple y fue previamente considerada en [82]. La imagen original se reducirá a c distintos conjuntos de píxeles (Φ_1, \dots, Φ_c) . La imagen Φ_i la compondrán aquellos píxeles en los que la imagen original verifique cierta propiedad local. En consecuencia, se considerarán conjuntamente c propiedades distintas. Pasemos a presentar dos ejemplos de esta idea.

Ejemplo 1 La primera y más sencilla posibilidad conduciría a definir $\Phi_i = \{x \in W : \frac{256}{c}(i-1) \leq f(x) < \frac{256}{c}i\}$ para $i = 1, \dots, c$ donde asumimos que tanto c como $\frac{256}{c}$ son enteros positivos y $f(x)$ denota la intensidad en x ($0 \leq f(x) \leq 255$ si usamos un byte por píxel). De esta manera, la intensidad se fragmenta en c intervalos delimitados por puntos igualmente espaciados. Las imágenes binarias de la figura 2.3 muestran este procedimiento para la textura 2.1(b) y con $c = 4$.

Ejemplo 2 La segunda posibilidad que vamos a utilizar consiste en describir los picos y valles de la imagen por medio de las transformaciones sombrero de copa [85]. A partir de ahora las denotaremos por TH , como referencia a su nombre en inglés: top-hat.

Repasemos las definiciones de este tipo de transformaciones morfológicas. Sea T un conjunto compacto, el elemento estructurante (básico en Morfología Matemática), y sea f una imagen a niveles de gris.

La dilatación de f mediante T es la imagen definida como $f \oplus T(x) = \sup\{f(y) : y \in \check{T} + x\}$, donde $\check{T} = \{-t : t \in T\}$ es el simétrico de T respecto del origen. La erosión de f mediante T es la imagen $f \ominus T(x) = \inf\{f(y) : y \in \check{T} + x\}$. A partir de estas operaciones morfológicas básicas, definimos la apertura de f mediante el elemento estructurante T como $f \circ T = (f \ominus \check{T}) \oplus T$, mientras que el cierre es $f \bullet T = (f \oplus \check{T}) \ominus T$. La transformación TH de f con respecto a la apertura definida por T viene dada por

$$f_T^{(1)} = f - f \circ T, \quad (2.13)$$

mientras que, la transformación TH de f con respecto al cierre definido por T es

$$f_T^{(2)} = (f \bullet T) - f. \quad (2.14)$$

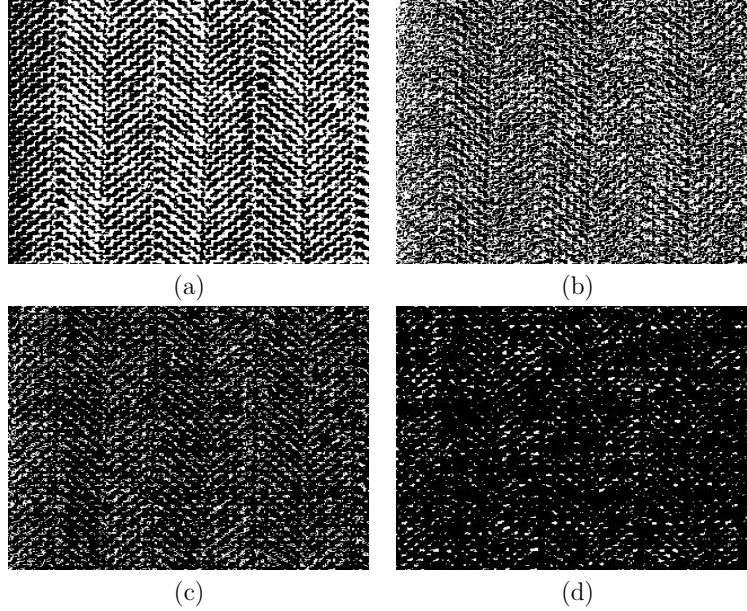


Figura 2.3: Texturas binarias obtenidas de la imagen de la figura 2.1(b) mediante el método de fragmentación de la intensidad

Se ha demostrado que estas transformaciones son una excelente herramienta para la extracción de pequeños objetos claros (respectivamente oscuros) con respecto a su fondo [85]. Algunas referencias generales sobre Morfología Matemática son [85], [87], [80] y [79].

En la tabla 2.1 se recogen los dos elementos estructurantes utilizados, T_1 y T_2 .

Tabla 2.1: Elementos estructurantes utilizados en las transformaciones TH

$$\begin{array}{ccc}
 0 & 1 & 0 \\
 1 & 1 & 1 \\
 0 & 1 & 0 \\
 T_1 & & T_2
 \end{array}$$

El conjunto aleatorio multivariante asociado con cada textura será (Φ_1, \dots, Φ_4) donde

$$\begin{aligned}
 \Phi_1 &= \{x \in W : f_{T_1}^{(1)}(x) > 0\} \\
 \Phi_2 &= \{x \in W : f_{T_2}^{(1)}(x) > 0\} \\
 \Phi_3 &= \{x \in W : f_{T_1}^{(2)}(x) > 0\} \\
 \Phi_4 &= \{x \in W : f_{T_2}^{(2)}(x) > 0\}
 \end{aligned}$$

Notemos que Φ_1 y Φ_2 corresponderían a los picos mientras que Φ_3 y Φ_4 a los valles. En ambos casos, el pico y el valle son relativos al elemento estructurante empleado. La figura 2.4 muestra las cuatro texturas binarias que se obtienen con las transformaciones TH de la figura 2.1(b).

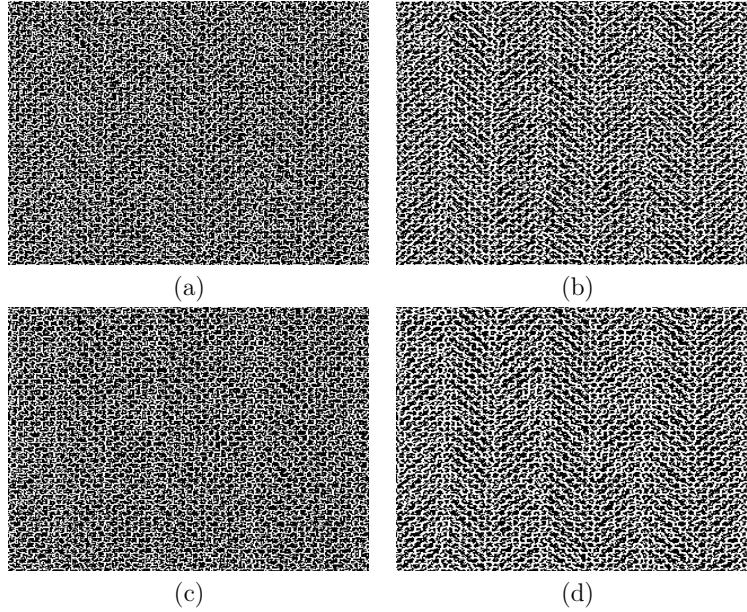


Figura 2.4: Texturas binarias obtenidas de la imagen de la figura 2.1(b) mediante las transformaciones TH

Desde un punto de vista teórico, hemos asociado un conjunto aleatorio multivariante $\Phi = (\Phi_1, \dots, \Phi_c)$ con la imagen original. Por ello, para obtener un conjunto aleatorio multivariante con los distintos métodos de segmentación, la textura original tiene que ser un proceso estocástico. (ϕ_1, \dots, ϕ_c) denotará la realización de este conjunto aleatorio multivariante, es decir, ϕ_i estará compuesta por aquellos píxeles que verifiquen la i -ésima propiedad local. La distribución de este conjunto aleatorio multivariante se describirá marginalmente tal y como se ha explicado previamente. No obstante, necesitaremos descriptores adicionales para capturar el comportamiento conjunto de estos conjuntos aleatorios. Para cada par (i, j) donde $i, j \in \{1, \dots, c\}$ podemos definir la \mathcal{K} -función cruzada:

$$\mathcal{K}_{ij}(t) = \frac{\int_{B(0,t)} C_{ij}(h) dh}{p_i p_j}, \quad (2.15)$$

donde p_i y p_j denotan la fracción de volumen de Φ_i y Φ_j y $C_{ij}(h) = P(0 \in \Phi_i, h \in \Phi_j)$ la función de covarianza cruzada. Las funciones de covarianza cruzadas nos indican la

probabilidad de que un punto pertenezca a una cierta componente y otro punto separado del anterior por el vector h pertenezca a otra determinada componente. Por su parte, $p_j \mathcal{K}_{ij}(t)$ podemos interpretarlo como el área media cubierta por el conjunto Φ_j en una distancia t desde un punto de Φ_i arbitrariamente elegido.

La función \mathcal{K}_{ij} se ha estimado con la ecuación 2.15 y el mismo método de integración numérica expuesto anteriormente (ecuaciones 2.9 y 2.10) y donde el estimador de C_{ij} es

$$\hat{C}_{ij}(h) = \frac{\nu(\phi_i \cap \phi_j - h)}{\nu(W \cap W - h)}, \quad (2.16)$$

y el estimador de la fracción de volumen p_i es como de costumbre $\hat{p}_i = \frac{\nu(\phi_i)}{\nu(W)}$. A este estimador lo denotaremos como $\hat{\mathcal{K}}_{ij}$.

Asimismo llamaremos $H_{ij}^{(s)}$ (respectivamente, $H_{ij}^{(l,\alpha)}$) a la distribución de contacto esférica (respectivamente lineal) de $\Phi_i \cup \Phi_j$. Estas funciones se han estimado aplicando las ecuaciones 2.11 y 2.12 a $\phi_i \cup \phi_j$.

En resumen, el conjunto de descriptores que usaremos con las texturas a niveles de gris será los descriptores marginales ya descritos para texturas binarias y cruzados $\hat{\mathcal{K}}_{ij}$, $\hat{H}_{ij}^{(s)}$ y $\hat{H}_{ij}^{(l,\alpha)}$ para $i > j$, con $i, j \in \{1, \dots, c\}$.

Aquí, se han mostrado dos formas simples de asociar un conjunto aleatorio multivariante a una textura. Por supuesto, se podrían proponer métodos más sofisticados que dieran distintas segmentaciones de la textura a través de, por ejemplo, diversos filtros. Posiblemente es la cuestión abierta de mayor interés en este tema.

2.1.3. Matrices de co-ocurrencia

Como un caso particular, consideremos que $\Phi_i = \{x : f(x) = i\}$, el conjunto de píxeles de la textura que toman el valor i . El número de componentes c coincide con el número de distintos valores que puede adoptar la intensidad. Una de las descripciones de texturas más populares (como ya se ha comentado en el capítulo 1), que fue propuesta por Haralick [11], son las matrices de co-ocurrencia. Dado un vector h , la matriz de co-ocurrencia se define como $P_h = [P_h(i, j)]$ donde $P_h(i, j)$ es el número de pares de píxeles x e y tales que $f(x) = i$, $f(y) = j$ donde $y = x + h$. En nuestra notación, esta condición puede describirse como $x \in \Phi_i$ e $y \in \Phi_j$. Claramente, $\hat{C}_{ij}(h)$ es proporcional a $P_h(i, j)$. La constante de normalización nos permite interpretar el valor observado $P_h(i, j)$ como una probabilidad. La elección de h es el problema básico de las matrices de co-ocurrencia. En nuestra aproximación, las \mathcal{K}_{ij} funciones son nuestra información básica (de segundo orden), que son versiones integradas de los valores $P_h(i, j)$. Obviamente, esto conduciría a demasiados conjuntos Φ_i ($c = 256$ si se usa un byte por píxel). La computación se reduciría con una mejor selección de Φ_i .

2.2. Métricas usadas para la clasificación de texturas

En la clasificación de texturas se asume que la textura a clasificar pertenece a una y sólo una clase de entre un conjunto completamente conocido de clases. Cada una de estas clases viene usualmente caracterizada por un conjunto de texturas prototipo. Un vector de características describe cada textura. Sea x_0 el vector asociado a la textura que se ha de clasificar. En cada clase i , con $i = 1, \dots, I$ (I es el número de clases), tendremos x_{ij} , las características de los prototipos, con $j = 1, \dots, n_i$ (n_i es el número de prototipos en la clase i -ésima). Tenemos un problema de clasificación. Una solución bastante común es considerar que el vector de características sigue una distribución normal (o gaussiana) multivariante con media μ_i y matriz de varianzas-covarianzas Σ_i que depende de la clase i -ésima. Usando información previa, especificamos las probabilidades a priori π_i con $i = 1, \dots, I$. Normalmente, se utiliza una distribución a priori no informativa con $\pi_i = 1/I$. Entonces definimos las probabilidades a posteriori según el teorema de Bayes

$$\pi(i | x_0) = \frac{\pi_i f_i(x_0)}{\sum_{j=1}^I \pi_j f_j(x_0)}, \quad (2.17)$$

donde

$$f_i(x_0) = |2\pi\Sigma_i|^{-1/2} \exp\left\{-\frac{1}{2}(x_0 - \mu_i)'\Sigma_i^{-1}(x_0 - \mu_i)\right\}. \quad (2.18)$$

Usualmente, los parámetros μ_i y Σ_i son desconocidos y se estiman mediante la muestra x_{ij} ($j = 1, \dots, n_i$) con estimadores máximo verosímiles: la media muestral \bar{x}_i para μ_i y la matriz de covarianzas muestral S_i para Σ_i . En consecuencia, $f_i(x_0)$ lo estimamos con

$$\hat{f}_i(x_0) = |2\pi S_i|^{-1/2} \exp\left\{-\frac{1}{2}(x_0 - \bar{x}_i)'S_i^{-1}(x_0 - \bar{x}_i)\right\}, \quad (2.19)$$

y las probabilidades a posteriori estimadas serían

$$\hat{\pi}(i | x_0) = \frac{\pi_i \hat{f}_i(x_0)}{\sum_{j=1}^I \pi_j \hat{f}_j(x_0)}. \quad (2.20)$$

La textura se clasificará en la clase con la probabilidad a posteriori estimada mayor. Al aplicarse este método, los resultados obtenidos sugirieron el uso de una matriz de covarianzas común Σ estimada por S , por tanto, se han reemplazado las matrices S_i por S . En casi todos los experimentos propuestos más adelante, se asumirá una distribución a priori uniforme, con lo cual la textura de características x_0 se clasificará en la clase en que la distancia de Mahalanobis, $(x_0 - \bar{x}_i)'S^{-1}(x_0 - \bar{x}_i)$, sea mínima.

Es importante notar que las características usadas son de hecho funciones estimadas en un conjunto discreto de puntos. Tenemos seis tipos de funciones: \mathcal{K} , distribuciones de contacto esféricas y lineales (cuatro orientaciones). Cada tipo de función tiene c versiones univariantes, además de $c(c-1)/2$ versiones cruzadas. Si d denota una métrica en el

correspondiente espacio euclídeo, entonces la medida de disimilaridad entre la textura y la clase i -ésima cuando se comparan las \mathcal{K} funciones podría ser

$$D = \sum_{j=1}^c d(\hat{\mathcal{K}}_j, \bar{\mathcal{K}}_j^{(i)}) + \sum_{j>k}^c d(\hat{\mathcal{K}}_{jk}, \bar{\mathcal{K}}_{jk}^{(i)}) \quad (2.21)$$

donde $\bar{\mathcal{K}}_j^{(i)}$ ($\bar{\mathcal{K}}_{jk}^{(i)}$) es la $\hat{\mathcal{K}}_j$ (respectivamente $\hat{\mathcal{K}}_{jk}$) media estimada con los prototipos de la clase i -ésima. Si reemplazamos las \mathcal{K} funciones por las funciones de distribución de contacto, obtendríamos la medida de disimilaridad correspondiente basada en dichas funciones. En resumen, dada d se han definido dos medidas de disimilaridad: una para las funciones \mathcal{K} y otra para las funciones H .

Se han usado tres métricas d distintas: primero, la distancia euclídea y segundo la distancia supremo. Se alcanzaron buenos resultados con la distancia supremo y las distribuciones de contacto, como se verá en la sección próxima. Sin embargo, los resultados de clasificación obtenidos con las funciones \mathcal{K} y ambas métricas no fueron tan satisfactorios. Hay que tener en cuenta que las funciones \mathcal{K} son crecientes y no acotadas. Las diferencias más grandes se observan para valores grandes de t . Para t grandes $\mathcal{K}(t)$ describe la 'tosquedad' (*coarseness*) de la textura. Por consiguiente, si empleamos la distancia euclídea o la supremo para comparar las funciones \mathcal{K} , entonces se tiene en cuenta básicamente sólo la 'tosquedad' de ambas texturas y no propiedades más locales. Para resolver esta cuestión, se ha considerado una tercera métrica para la ecuación 2.21, d_m , dada por

$$d_m(\hat{\mathcal{K}}, \bar{\mathcal{K}}^{(i)}) = \sup \frac{|\hat{\mathcal{K}}(t) - \bar{\mathcal{K}}^{(i)}(t)|}{\max\{\hat{\mathcal{K}}(t), \bar{\mathcal{K}}^{(i)}(t)\}}, \quad (2.22)$$

donde $\hat{\mathcal{K}}$ puede ser tanto una versión univariante o cruzada y $\bar{\mathcal{K}}^{(i)}$ es la media correspondiente para la clase i -ésima.

Clasificaremos una textura en la clase más cercana de acuerdo con alguna de las medidas de disimilaridad anteriores. En la siguiente sección, se presentarán diferentes experimentos que se evaluarán con el clasificador de Bayes, la disimilaridad D para las distribuciones de contacto con la distancia supremo (etiquetadas con H/sup) y la disimilaridad D para las funciones \mathcal{K} con d_m (etiquetadas con \mathcal{K}/d_m). Sólo se presentarán los resultados con estos clasificadores, pues son con los que se han obtenido los mejores resultados.

2.3. Experimentos

Los cuatro primeros experimentos comparten el mismo diseño básico. Se han usado imágenes de la base de datos de texturas de Brodatz [88]. Cada una de las imágenes de 640×640 píxeles se ha particionado en cuatro subimágenes de 320×320 píxeles: superior izquierda, superior derecha, inferior izquierda e inferior derecha. La subimagen superior izquierda será el prototipo que definirá la clase, mientras que las restantes tres subimágenes entrarán a formar parte del conjunto de prueba.

Las funciones \mathcal{K} y las distribuciones de contacto esféricas se han estimado desde 1 a 30 píxeles con un incremento de un píxel. De la misma manera, las distribuciones de contacto lineales también se han estimado de 1 a 30 píxeles con el mismo incremento (un píxel) y cuatro valores de α : 0 , $\frac{\pi}{4}$, $\frac{\pi}{2}$ y $\frac{3\pi}{4}$ radianes.

Puesto que cada clase contiene un único prototipo, la matriz de covarianzas no puede estimarse. Únicamente la clasificación basada en las medidas de disimilaridad puede utilizarse. Las siguientes cuatro subsecciones siguen este procedimiento experimental. La clasificación de texturas binarias se estudiará en la subsección 2.3.1. Las subsecciones 2.3.2 y 2.3.3 se dedicarán a las texturas a niveles de gris donde los conjuntos aleatorios multivariantes se definirán usando el método de fragmentación de la intensidad y las transformaciones TH. La influencia del ruido se contemplará en la subsección 2.3.4. Se ha considerado una alternativa para poder evaluar el clasificador de Bayes en los anteriores experimentos: en la subsección 2.3.5 se ha llevado a cabo la evaluación leave-one-out (dejar uno fuera). Obviamente, los resultados de clasificación así obtenidos no pueden compararse con los de las cuatro primeras subsecciones. Finalmente, en la subsección 2.3.6 se ha realizado una comparación global con otros descriptores clásicos de texturas como características utilizadas en la clasificación de las mismas.

2.3.1. Primer experimento: texturas binarias

En esta subsección se han usado texturas binarias generadas mediante la umbralización de texturas de Brodatz. En concreto, se han utilizado las texturas D20, D52, D64, D65, D67, D68, D75, D84, D102 y D103 (usamos la misma notación estándar de esta base de texturas), las mismas empleadas en [89]. Pueden obtenerse por ejemplo desde <http://www.ux.his.no/~tranden/>.

Se han usado dos métodos distintos para generar dos bases de texturas binarias. Primero, los píxeles con valor 1 se corresponderán con intensidades menores que 128, excepto para la D67, donde el valor 1 de la imagen binaria representa las intensidades mayores o iguales que 128 en la textura original. Desde un punto de vista subjetivo, estas texturas binarias son versiones simplificadas de las originales y conservan la estructura de dichas texturas.

En la tabla 2.2 (primera fila) se muestra el porcentaje de texturas correctamente clasificadas con esta primera base de datos. Nótese que hay 30 texturas para clasificar y 10 clases a las que asignarlas. Los mejores resultados se logran con las distribuciones de contacto esféricas y lineales.

La segunda base de texturas binarias se ha generado mediante la umbralización de las texturas originales por la intensidad media observada, en lugar de por el umbral común anterior 128. Ahora los píxeles con valor 1 tienen intensidades menores que la intensidad media de la imagen, exceptuando la imagen D67, para la cual el 1 va emparejado a los valores mayores o iguales que la intensidad media. En el apéndice puede verse esta base de texturas binarias. Los porcentajes de clasificación correcta para estas texturas binarias,

Tabla 2.2: Porcentajes de clasificaciones correctas para los cuatro primeros experimentos

		\mathcal{K}/d_m	H/sup
Primer exp.	Primera base de texturas	66.67	93.33
	Segunda base de texturas	90	100
Segundo exp.	Cuatro partes	89.73	89.73
	Ocho partes	92.31	89.73
Tercer exp.		100	100
Cuarto exp.	Binarias (0.05)	90	100
	Binarias (0.2)	83.33	100
	Grisas (25)	92.31	100
	Grisas (57)	71.77	94.85

pueden verse también en la tabla 2.2 (segunda fila). Puede apreciarse que los resultados son algo mejores para esta segunda base de texturas. Quizá sea porque la textura se preserva mejor al utilizar un nivel de umbralización dependiente de la textura en lugar de un umbral común.

Parece que las funciones \mathcal{K} son más sensibles que las funciones H . Sin embargo, los mejores resultados nuevamente son conseguidos por las funciones de distribución de contacto.

2.3.2. Segundo experimento: texturas a niveles de gris y método de fragmentación de la intensidad

Las 26 texturas de la base de texturas de Brodatz que se han elegido son: D1, D3, D6, D16, D17, D22, D25, D26, D35, D36, D53, D55, D77, D78, D79, D80, D81, D82, D83, D84, D85, D94, D95, D96, D103 y D104. En el apéndice pueden contemplarse.

Un procedimiento muy simple para construir el conjunto aleatorio multivariante sería fragmentar la intensidad en cuatro intervalos con intensidades: de 0 a 63, de 64 a 127, de 128 a 191 y por último mayor o igual que 192.

En este proceso de segmentación únicamente se usan los píxeles individualmente y no la información de los píxeles vecinos. Los porcentajes de clasificaciones correctas sobre 78 texturas aparecen en la tabla 2.2 (tercera fila titulada *Cuatro partes*). Señalemos que los resultados no son demasiado malos a pesar de la sencillez del procedimiento empleado para producir las imágenes binarias asociadas.

Por otro lado, también se han generado ocho texturas binarias fragmentando el rango de intensidades en 8 intervalos iguales. Cada uno de los intervalos anteriores, se ha fraccionado en dos mitades (el intervalo de 0 a 63 se ha subdividido en los intervalos de 0 a 31 y de 32 a 63).

Otra vez, la segmentación solamente tiene en cuenta los píxeles aisladamente y por

separado, no la estructura espacial existente. Como puede apreciarse en la tabla 2.2 (cuarta fila titulada *Ocho partes*), el trocear en ocho partes en lugar de en cuatro no supone una mejora significativa de los resultados de clasificación. Esto nos lleva a pensar que la resolución final depende en gran medida de la *bondad* de los conjuntos aleatorios multivariantes, más que del *número de componentes*. Por otra parte, no hay grandes diferencias entre las funciones \mathcal{K} y las funciones H .

2.3.3. Tercer experimento: texturas a niveles de gris y transformaciones TH

Para este experimento, se han escogido un grupo de 13 texturas de Brodatz particularmente similares: D16, D17, D53, D55, D77, D78, D79, D80, D81, D82, D83, D84 y D85. En este caso, los conjuntos aleatorios multivariantes tienen como componentes las regiones obtenidas mediante las transformaciones TH respecto a la apertura y cierre, tal y como se definieron en la sección 2.1. En la tabla 2.2 (quinta fila) se muestran los resultados obtenidos. Se ha producido una importante mejora en comparación al segundo experimento, a pesar de la dificultad (las texturas son realmente muy parecidas). Esto muestra que la construcción de las distintas texturas binarias a partir de la textura original desempeña un papel fundamental. Recordemos que en este caso un píxel pertenece a una componente determinada dependiendo tanto de su propio valor como de los valores de su vecindad.

2.3.4. Cuarto experimento: ruido

En este apartado se estudia la robustez de la metodología frente al ruido. Tanto los prototipos como las texturas se han degradado con distintos tipos de ruido y a diferentes niveles.

Primeramente, se ha añadido ruido sal y pimienta [5] a la segunda base de texturas binarias definida en la subsección 2.3.1. Este es un tipo de ruido impulsivo, que provoca que la intensidad de cada píxel difiera significativamente de la de sus vecinos. El valor de cada píxel se intercambia (de 1 a 0 o viceversa) independientemente y con la misma probabilidad, con lo que la imagen se corrompe mediante píxeles blancos y negros (de ahí su nombre). Se han usado dos probabilidades distintas, 0.05 y 0.2. Los resultados se dan en la tabla 2.2 (sexta y séptima fila).

En segundo lugar, hemos añadido ruido gaussiano [5] con desviaciones típicas 25 y 57 a las texturas del tercer experimento. Para cada uno de los píxeles originales, se ha generado un ruido de una distribución gaussiana (con media cero y desviaciones típicas ya dichas), que se ha añadido al valor original de cada píxel. La figura 2.5 muestra el cuadrante superior izquierdo de la textura D83, así como sus respectivas versiones degradadas. Los conjuntos aleatorios se han construido como en el tercer experimento. En las dos últimas filas de la tabla 2.2 se recogen los resultados obtenidos. Globalmente, las funciones H muestran un mejor desempeño que las funciones \mathcal{K} . Hay que destacar los resultados óptimos (100%)

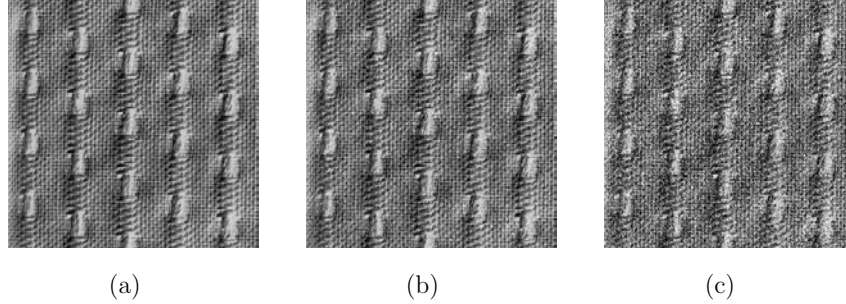


Figura 2.5: (a) Textura original D83; D83 con ruido gaussiano de desviaciones típicas 25 y 57

Tabla 2.3: Porcentajes de clasificaciones correctas para los cuatro primeros experimentos con el clasificador de Bayes

		\mathcal{K}	$\hat{H}^{(s)}$	$\hat{H}^{(l,0)}$	$\hat{H}^{(l,\frac{\pi}{4})}$	$\hat{H}^{(l,\frac{\pi}{2})}$	$\hat{H}^{(l,\frac{3\pi}{4})}$	H
Primer exp.	Primera base	80	82.5	100	90	97.5	95	100
	Segunda base	100	90	100	100	100	97.5	100
Segundo exp.	Cuatro partes	100	100	97.1	99	100	96.2	100
Tercer exp.		100	98.1	100	100	98.1	100	100
Cuarto exp.	Binarias (0.05)	97.5	85	100	97.5	100	97.5	100
	Binarias (0.2)	97.5	77.5	100	97.5	100	97.5	100
	Grises (25)	100	92.3	100	100	100	100	100
	Grises (57)	100	50	100	88.5	94.2	84.6	100

que se logran con las texturas binarias degradadas y las texturas a niveles de gris con un ruido moderado (desviación típica igual a 25). En cualquier caso, las funciones \mathcal{K} son también bastante robustas.

2.3.5. Quinto experimento: clasificación de Bayes

Como ya mencionamos previamente, si únicamente se asigna un prototipo por clase, no podemos estimar la matriz de covarianzas y diseñar el clasificador de Bayes. Por tanto, hemos adoptado una estrategia diferente: el método leave-one-out. En este procedimiento, se excluye una muestra del conjunto total de muestras disponibles, se diseña el clasificador de Bayes con las muestras restantes y la muestra excluida la empleamos para probar la clasificación. Esta operación la repetimos y así examinamos todas las muestras.

En la tabla 2.3 se presentan los porcentajes de clasificaciones correctas con este procedimiento para los cuatro primeros experimentos expuestos anteriormente.

Se han considerado siete conjuntos de características: las funciones \mathcal{K} y las funciones H ,

separada y conjuntamente. Se han conseguido buenos resultados en todos los experimentos. Nótese que se ha alcanzado un porcentaje de 100 % de clasificaciones correctas para algunas características en todos los experimentos.

El número más alto de muestras erróneamente clasificadas lo proporciona las distribuciones de contacto esféricas, en la siguiente subsección este hecho será de nuevo confirmado. Respecto al resto de las características, deben resaltarse los excelentes resultados obtenidos.

2.3.6. Sexto experimento: comparación con otras características de texturas

El objetivo de este experimento es comparar los resultados de clasificación obtenidos con nuestros descriptores con aquellos que se obtienen usando descriptores clásicos. En esta sección, se ha considerado la base de texturas que empleó Kaplan ([90]). Esta base de texturas la componen un conjunto de 77 texturas de tamaño 512×512 de las bases de Brodatz y Bonn/MIT, cuyas direcciones son <http://sipi.usc.edu/services/database/Database.html> y <http://www-dbv.cs.uni-bonn.de/image/texture.tar.gz>.

Las 77 texturas las agruparemos en 71 clases, de forma que hay 6 clases definidas por 2 texturas similares.

Cada textura original la dividiremos en 64 subtexturas (no solapadas) de tamaño 64×64 . De esta manera, 32 las usaremos como prototipos y las 32 restantes las usaremos como imágenes test. Como consecuencia, las clases compuestas por dos texturas tendrán 64 prototipos y 64 muestras de prueba. En el apéndice se muestran las texturas usadas.

Se han extraído cuatro conjuntos de características, muy conocidas y usadas en el campo de la clasificación de texturas: estimaciones de los parámetros de un campo aleatorio de Markov gaussiano (Gaussian Markov Random Field - MRF), algunos descriptores de matrices de co-ocurrencia (Gray Level Co-occurrence Matrices - LCM), de dimensión fractal (Fractal Dimension - FRD) y energías de convoluciones con filtros tipo Gabor (Gabor Convolution Energies - GCE). En [11] y en el capítulo 1 podemos encontrar una presentación detallada de estos métodos.

El primer procedimiento se basa en la suposición que la textura es una realización de un campo de Markov gaussiano. Estos modelos asumen que la intensidad de cada píxel de la imagen depende únicamente de las intensidades de los píxeles vecinos [11]. Las estimaciones de los parámetros de estos modelos estocásticos describirán las características de la textura. En este experimento, se ha usado una vecindad simétrica de 20 píxeles. Los parámetros se han calculado usando el método de mínimos cuadrados que se detalla en [13].

El segundo procedimiento es el de las matrices de co-ocurrencia [5]. Éstas se basan en la ocurrencia repetida de alguna configuración de intensidades de la textura, como se ha visto en el apartado 2.1.3. Después se suelen computar una serie de medidas. En concreto

aquí se han calculado (y posteriormente se ha obtenido la media de ellas) las matrices de co-ocurrencia para las siguientes separaciones entre píxeles: distancia de 1 y ángulos $\{0, 45, 90, 135\}$. Luego, hemos extraído cinco características (véase [91] y el capítulo 1 para los detalles): energía, entropía, correlación, momento de diferencia inversa e inercia.

En el tercer algoritmo se trata la dimensión fractal. Un fractal determinístico se define usando el concepto de autosimilaridad. Dado un conjunto acotado A de un espacio euclídeo decimos que es autosimilar cuando A es la unión de N copias distintas (no solapadas) de sí mismo, cada una de las cuales escalada una proporción r . La dimensión fractal viene dada por:

$$D = \frac{\log(N)}{\log(1/r)}. \quad (2.23)$$

Se ha mostrado que la dimensión fractal está correlacionada con una función intuitiva de 'asperidad' [14]. Se han propuestos bastantes métodos de estimación de la dimensión fractal D . Nosotros hemos usado el método *differential box-counting* explicado en [15] y se han extraído cuatro características: D de la imagen original f , D de dos transformaciones de la imagen f y la multifractal D (véase [15] y el capítulo 1).

Finalmente, el cuarto método usa filtros de Gabor. Una función de Gabor consiste de sinusoides moduladas por una ventana gaussiana y pueden generarse variando la longitud de onda, orientación, fase y desviaciones típicas de la ventana gaussiana (véase el capítulo 1). Las características se obtendrán calculando la energía (suma, sobre las fases, de los cuadrados de la convolución de la imagen con un filtro de Gabor) como en [92]. Se han considerado longitudes de onda (en píxeles) $\{2, 4, 8\}$ y orientaciones $\{0, 45, 90, 135\}$, obteniéndose un total de 12 características. G. Smith e I. Burns implementaron los cuatro métodos expuestos que pueden obtenerse en su paquete público MEASTEX [93] (<http://www.cssip.uq.edu.au/staff/meastex/>) donde además de los algoritmos en sí, puede encontrarse una explicación detallada de los mismos. Salvo pequeñas modificaciones, hemos usado aquí dicho paquete.

Como los mejores resultados fueron proporcionados con las transformaciones TH, hemos decidido usarlas en esta parte. Las funciones \mathcal{K} se han estimado de 1 a 20 con un paso de 2. Las funciones H se han estimado de 1 a 20 con un paso de 1. Nuevamente, se han considerado las cuatro orientaciones $\{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$. Éstas serán las características que se compararán con las otras opciones.

En la tabla 2.4 exponemos los porcentajes para cada uno de los métodos. Para MRF, LCM, FRD y GCE se ha usado el clasificador de Bayes con distintas matrices de covarianzas. También se ha empleado el clasificador de Bayes, pero con una matriz de covarianzas común, con nuestras características (segunda columna). La elección entre una matriz de covarianzas común o dependiente de la clase se ha realizado atendiendo a los mejores resultados de clasificación obtenidos.

Las funciones H se han usado separada y conjuntamente. Las filas tituladas \mathcal{K}/d_m y H/sup tienen el mismo significado que antes, es decir, cada textura se clasifica en la clase más cercana según la medida de disimilaridad previamente definida.

Tabla 2.4: Porcentajes de clasificación correcta para la base de texturas de Kaplan

MRF	92.2	\mathcal{K}	91.2
LCM	89	$H^{(s)}$	57.3
FRD	79.8	$H^{(l,0)}$	86.6
GCE	90.5	$H^{(l,45)}$	83.7
\mathcal{K}/d_m	69.5	$H^{(l,90)}$	83.7
H/sup	81.8	$H^{(l,135)}$	83.1
		H/Bayes	96.3

Las distribuciones de contacto lineales estimadas en unos pocos puntos tienen un poder discriminatorio similar al de los métodos clásicos. Si además, se consideran distintas orientaciones conjuntamente, obtenemos mejores resultados (96.3%).

Es realmente destacable que las funciones \mathcal{K} proporcionan un porcentaje de clasificación correcta de 91.2%, que es casi igual que el mejor estándar para esta base, MRF. Las distribuciones de contacto esféricas no se comportan bien debido a que la 'tosquedad' de la textura es demasiado fina para ser detectada con un elemento estructurante tan grande. Pensamos que para texturas más 'toscas' se obtendrían mejores resultados. Finalmente, las disimilaridades definidas en la ecuación 2.21 no dan buenos resultados. Obviamente, para aplicaciones concretas como las de los primeros experimentos con sólo un prototipo por clase, la aproximación basada en las disimilaridades de la ecuación 2.21 constituye una posibilidad útil.

Capítulo 3

Una aproximación posibilística para la clasificación de texturas

Este capítulo se ocupa del problema de la clasificación de texturas desde un punto de vista posibilístico. Aunque la metodología que presentaremos podría aplicarse a cualquier tipo de problema de clasificación sin tener que realizar ninguna modificación, nosotros nos centraremos en la clasificación de texturas.

Como ya es sabido, en un problema de clasificación de texturas, una textura ha de clasificarse en un conjunto de clases preestablecidas. Es común definir estas clases en base a una serie de texturas prototipo. De hecho, un procedimiento experimental usual en la literatura de clasificación de texturas, consiste en particionar cada textura en subimágenes y utilizar la mitad de ellas como conjunto de entrenamiento (ellas definen las clases) y la otra mitad como conjunto de prueba, para valorar la bondad de la clasificación. Por descontado, estas texturas no son modelos estocásticos distintos y bien establecidos (como podrían ser los modelos germen-grano, campos aleatorios de Markov,...), sobre todo en el caso de aplicaciones reales, con lo cual es el usuario (el experto) el que decide qué texturas constituyen una clase bien definida.

Como ya comentamos, en general, el proceso de clasificación de texturas conlleva dos pasos. El primero es la extracción de características tanto de los prototipos como de las texturas a clasificar. El segundo es la etapa de clasificación basada en las características anteriores. En la literatura de clasificación de texturas, es habitual prestar más atención a la primera fase (búsqueda de nuevas y mejores características). En cambio, el segundo paso suele resolverse en un contexto bayesiano, sin darle quizá la relevancia que merece. Es común asumir probabilidades a priori iguales para las distintas clases, así como una distribución normal (o gaussiana) para las características. De esta manera, la textura se asigna a la clase con una probabilidad a posteriori mayor. En los experimentos que siguen, mostraremos que la hipótesis de normalidad no se sustenta para las características habitualmente utilizadas en la clasificación de texturas. Esto implica que las probabilidades a posteriori obtenidas asumiendo una distribución normal no serían valores correctos. Por todo ello, proponemos una metodología que asocia a cada textura un grado de pertenencia

en cada clase, una etiqueta posibilística [56], y así cada clase será considerada como un conjunto difuso. Esta visión de la clasificación de texturas bajo una óptica difusa, podría también servir para comparar diversos procedimientos de clasificación de texturas, puesto que podríamos disponer de más medidas de bondad a la hora de evaluar las diferencias entre características de texturas, en lugar del típico porcentaje de clasificación correcta. Pasamos a presentar la metodología empleada y los distintos experimentos realizados.

3.1. Metodología

Supongamos que en un problema de clasificación de texturas, el número de clases o categorías es I . Básicamente, el problema consiste en decidir si una nueva muestra (textura) pertenece a una cierta categoría. En la práctica, cada categoría viene definida mediante un conjunto de n muestras, las muestras de entrenamiento o aprendizaje. Si cada textura la describimos mediante un vector de características x , el objetivo sería determinar si la nueva muestra, x_0 , es similar al (compatible con) conjunto de prototipos ($\{x_1, \dots, x_n\}$) que definen la clase. Desde una perspectiva difusa, la medida de similaridad puede verse como un grado de pertenencia a la categoría definida por las muestras de aprendizaje. Con el fin de determinar esta medida de similaridad, podríamos utilizar procedimientos estadísticos habituales.

Supongamos que el vector de características es un vector aleatorio y que x_1, \dots, x_n son los valores observados de una muestra aleatoria, X_1, \dots, X_n son variables aleatorias independientes e idénticamente distribuidas con función de densidad de probabilidad f .

Una opción simple sería calcular la distancia de Mahalanobis, sin embargo, información importante podría desaparecer con la transformación ([1]). En consecuencia, hemos adoptado otras aproximaciones.

3.1.1. Primera aproximación

Empezamos asumiendo una forma paramétrica para la función de densidad, $f(x | \theta)$. En la práctica, los valores verdaderos de los parámetros son desconocidos, así que deben estimarse a partir de la muestra. Una técnica popular es el método de máxima verosimilitud, que consiste en maximizar la función de verosimilitud definida por

$$L(\theta | x_1, \dots, x_n) = \prod_{i=1}^n f(x_i | \theta). \quad (3.1)$$

Si $\hat{\theta}$ denota la estimación máximo verosímil del parámetro θ , entonces podemos considerar una función de pertenencia definida por:

$$\mu_1(x_0) = f(x_0 | \hat{\theta}). \quad (3.2)$$

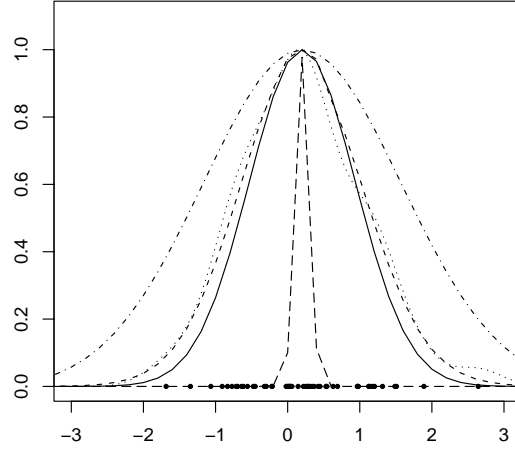


Figura 3.1: Funciones de pertenencia

Ejemplo 3 Supongamos que los prototipos $\{x_1, \dots, x_n\}$ es una muestra aleatoria de una distribución normal. Los datos usados en éste y en los ejemplos posteriores, se muestran en el eje x de la figura 3.1 y han sido generados a partir de una variable aleatoria normal de media 0 y varianza 1. Los estimadores máximo verosímiles de la media y varianza son la media muestral y la varianza muestral \bar{x} y s^2 dadas por $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$ y $s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$. Para este ejemplo, la primera función de pertenencia propuesta quedaría

$$\mu_1(x_0) = \frac{1}{\sqrt{2\pi s^2}} \exp \left[-\frac{(x_0 - \bar{x})^2}{2s^2} \right]. \quad (3.3)$$

Esta función de pertenencia normalizada, se muestra en la figura 3.1 (línea continua).

Además de la aproximación clásica previa, podríamos considerar una aproximación bayesiana. De esta manera, θ es una cantidad aleatoria descrita por la distribución a priori con densidad $\xi(\theta)$. Esta distribución se modifica con la información muestral y por el teorema de Bayes obtendríamos la llamada distribución a posteriori $\xi(\theta | x_1, \dots, x_n)$, que es proporcional a $\xi(\theta) \prod_{i=1}^n f(x_i | \theta)$. Dos funciones de pertenencia razonables podrían ser

$$\mu_2(x_0) = \int_{\Theta} f(x_0 | \theta) \xi(\theta | x_1, \dots, x_n) d\theta, \quad (3.4)$$

donde Θ es el espacio de parámetros, es decir, el conjunto de posibles valores para θ y

$$\mu_3(x_0) = f(x_0 | \tilde{\theta}), \quad (3.5)$$

donde $\tilde{\theta}$ es la estimación máximo a posteriori (MAP), es decir, $\tilde{\theta} = \operatorname{argmax}_{\theta \in \Theta} \xi(\theta | x_1, \dots, x_n)$.

Ejemplo 4 *Supongamos que nuestros prototipos, presentados en la figura 3.1, siguen una distribución normal. Bajo la aproximación bayesiana, asignamos una distribución a priori a los parámetros. Sean M y R la media y la precisión (definida como la inversa de la varianza, esto es, $R = 1/\sigma^2$). Como es habitual en el contexto bayesiano [94], hemos asumido para la distribución a priori de la precisión R una distribución gamma y una distribución normal para la media condicionada a una precisión dada, o sea,*

$$R \sim \operatorname{Ga}(\alpha, \beta), \quad (3.6)$$

y

$$M | R = r \sim N(m, \tau r). \quad (3.7)$$

Si ξ denota la densidad de la distribución a posteriori entonces

$$\xi(M, R) \propto \left\{ R^{\frac{1}{2}} \exp\left(-\frac{(\tau+n)R}{2}(M-\mu')^2\right) \right\} (R^{\alpha+\frac{n}{2}-1} e^{-\beta' r}), \quad (3.8)$$

donde \propto significa “proporcional a” y

$$\mu' = \frac{\tau m + n \bar{x}}{\tau + n}, \quad \beta' = \beta + \frac{1}{2} \sum_{i=1}^n (x_i - \bar{x})^2 + \frac{\tau n (\bar{x} - m)^2}{2(\tau n)}. \quad (3.9)$$

La segunda función de pertenencia sería

$$\begin{aligned} \mu_2(x_0) &= \int_{\Theta} f(x_0 | \theta) \xi(\theta | x_1, \dots, x_n) d\theta \propto \\ &\int_{R, M} R^{\alpha+\frac{n}{2}} \exp\left\{-\frac{R}{2}[(x_0 - M)^2 + (\tau+n)(M - \mu')^2 + 2\beta' r]\right\} dM dR. \end{aligned} \quad (3.10)$$

Si \tilde{M} y \tilde{R} denotan los estimadores máximo a posteriori (MAP) de la media y la precisión, entonces la tercera función de pertenencia vendría dada por

$$\mu_3(x_0) = \left(\frac{\tilde{R}}{2\pi}\right)^{\frac{1}{2}} \exp\left[\frac{-\tilde{R}(x_0 - \tilde{M})^2}{2}\right]. \quad (3.11)$$

Hemos elegido el caso concreto en que m , τ , α y β son iguales a 1, 0,5, 1 y 0,5 respectivamente. En la figura 3.1 están representadas μ_2 con línea discontinua (a guiones) y μ_3 con línea discontinua a rayas y puntos.

Hasta el momento, se ha asumido una forma paramétrica para la función de densidad f . Cuando no puede asumirse una estructura paramétrica, las técnicas no paramétricas podrían emplearse para estimar f . Si \hat{f} denota una estimación no paramétrica de f basada en la muestra $\{x_1, \dots, x_n\}$ [95], entonces la siguiente función de pertenencia podría considerarse,

$$\mu_4(x_0) = \hat{f}(x_0). \quad (3.12)$$

En la figura 3.1, está representada μ_4 para los datos de esta figura, utilizando una línea de puntos. Para hacer los cálculos de las funciones de pertenencia se han empleado las funciones disponibles en el programa *R* (<http://www.r-project.org/>).

Las funciones de pertenencia propuestas hasta ahora, no tienen porqué estar comprendidas entre 0 y 1. Si deseamos tener los valores convencionales pertenecientes al intervalo $[0, 1]$, éstos podrían obtenerse por medio de un proceso de normalización. Un procedimiento habitual es:

$$\mu'(x_0) = \frac{\mu(x_0) - \min_y \mu(y)}{\max_y \mu(y) - \min_y \mu(y)} \quad (3.13)$$

donde μ representa cualquiera de las funciones de pertenencia previamente definidas.

3.1.2. Segunda aproximación

Esta segunda aproximación hace uso de los procedimientos de contraste de hipótesis. El problema de contraste de hipótesis planteado es

H_0 : x_0 es un valor generado con la misma distribución que $\{x_1, \dots, x_n\}$,

H_1 : No H_0 .

Puesto que los p-valores proporcionan una medida de la evidencia (basada en los datos) de H_0 , la función de pertenencia podría definirse como el p-valor asociado con dicho test de hipótesis. En este caso, sí que obtenemos directamente valores entre 0 y 1. Tal y como hemos hecho anteriormente, trataremos el caso paramétrico en primer lugar.

Si U denota el estadístico elegido para el anterior contraste y valores grandes de U conducen al rechazo de H_0 , entonces la función de pertenencia sería

$$\mu_5(x_0) = P(U(x_0) \geq u \mid H_0), \quad (3.14)$$

donde u es el valor observado del estadístico U .

Los dos siguientes ejemplos ilustran la idea. En el primero continuamos con el caso univariante, mientras que en el segundo ejemplo se considera el caso multivariante.

Ejemplo 5 *Nuevamente, se ha asumido una distribución normal para los prototipos, es decir, X_1, \dots, X_n son independientes e idénticamente distribuidas, tales que $X_i \sim N(\mu, \sigma^2)$. El test anterior lo reformularemos como*

H_0 : $\mu = x_0$,

H_1 : No H_0 .

Si empleamos el test de la t de Student, entonces el estadístico sería

$$U = \frac{(\bar{X}_n - x_0)}{s} \sqrt{n-1} \quad (3.15)$$

y cuando H_0 es cierta,

$$U \sim t_{n-1}. \quad (3.16)$$

Finalmente

$$\mu_5(x_0) = P(U \geq u), \quad (3.17)$$

donde u es el valor observado de U . La función de pertenencia μ_5 puede apreciarse en la figura 3.1 con línea discontinua con rayas largas.

Ejemplo 6 Sea X_1, \dots, X_n una muestra aleatoria de una distribución normal multivariante con vector de medias μ y matriz de covarianzas Σ ($X_i \sim N_p(\mu, \Sigma)$). El contraste de hipótesis anterior lo reformularemos más precisamente como

H_0 : $\mu = x_0$,

H_1 : No H_0 .

Si utilizamos el test T^2 de Hotelling, entonces el estadístico sería $T_0^2 = n(\bar{x} - x_0)' S^{-1} (\bar{x} - x_0)$ (\bar{x} y S representan la media y matriz de covarianzas estimadas). Asumiendo que H_0 es cierta,

$$U = \frac{T_0^2}{n-1} \frac{n-p}{p} \sim F_{p, n-p}. \quad (3.18)$$

Finalmente,

$$\mu_5(x_0) = P(U \geq u), \quad (3.19)$$

donde u es el valor observado de U .

En muchas situaciones, una forma paramétrica para la distribución del estadístico de contraste U no puede asumirse, tal es el caso que nos ocupa. Es por ello que tendrían que emplearse procedimientos no paramétricos. Veamos un ejemplo.

Recordemos que estamos interesados en probar si la textura descrita por el vector x_0 , pertenece a la clase caracterizada por los prototipos $\{x_1, \dots, x_n\}$. Para este fin, emplearemos un test de aleatorización [96, 97, 98, 99], cuyo fundamento es, como veremos, muy sencillo. Para construir un test de aleatorización, se calcula un vector $d = (d_0, d_1, \dots, d_n)$, donde la componente d_k indica lo distinta que es la textura descrita por el vector x_k del resto de texturas ($\{x_l\}_{l=0, l \neq k}^{l=n}$). La componente d_k podría definirse como,

$$d_k = D(x_k, \bar{x}_k), \quad (3.20)$$

donde D indica una métrica en el espacio euclídeo p -dimensional y

$$\bar{x}_k = \sum_{j \neq k}^n \frac{x_j}{n}. \quad (3.21)$$

Una posible elección, utilizada en nuestros experimentos, podría ser la distancia de Mahalanobis donde

$$d_k = (x_k - \bar{x}_k)' S_k^{-1} (x_k - \bar{x}_k), \quad (3.22)$$

y S_k representa la matriz de covarianzas estimada usando todas las muestras, excepto el vector de características x_k .

Bajo la hipótesis nula H_0 de que la textura bajo estudio (cuyo valor asociado en d es d_0), pertenece a la clase descrita por dichos prototipos, todas las permutaciones de d_0, d_1, \dots, d_n son equiprobables. Sean $d_{(0)} \leq d_{(1)} \leq d_{(2)} \leq \dots \leq d_{(n)}$, los valores d_i ordenados de menor a mayor. Por consiguiente, bajo H_0 ,

$$P(d_0 = d_{(k)}) = \frac{1}{n+1} \quad \text{para } k = 0, \dots, n, \quad (3.23)$$

y el rechazo de H_0 si d_0 ocupa la k -ésima posición o mayor en la ordenación, nos da un test unilateral de tamaño $\frac{k}{n+1}$. El p-valor obtenido puede considerarse como un grado de pertenencia de la textura bajo estudio a la clase caracterizada por los prototipos ya mencionados y establecerá la función de pertenencia μ_6 .

Notemos que en lugar de una medida de disimilaridad (que ha venido inspirada por la metodología desarrollada en [99]) como la de la ecuación 3.20, podría usarse una medida de similaridad (cualquiera de las anteriores podría utilizarse) de cada textura con las restantes. En este caso, el único cambio sería que ordenaríamos las componentes d_k en orden inverso al anterior, es decir, de mayor a menor.

3.1.3. Problema de clasificación

Hemos definido distintas funciones de pertenencia que informan acerca de la compatibilidad del vector x_0 con un conjunto de prototipos. Estas funciones pueden usarse también, si nos enfrentamos al problema de clasificar x_0 en una de las I clases dadas. Supongamos que la i -ésima clase viene caracterizada por n_i muestras: $\{x_1^{(i)}, \dots, x_{n_i}^{(i)}\}$. Siguiendo cualquiera de los procedimientos propuestos, puede construirse una función de pertenencia $\mu^{(i)}$ para cada clase. Así por ejemplo, en el caso de la segunda aproximación (que es con la que trabajaremos en los experimentos), se realizarán I contrastes de hipótesis (uno para cada clase). De esta manera, conseguiremos I p-valores que indicarán el grado de pertenencia de la textura inicial a cada una de las clases. Destaquemos que estos p-valores no son probabilidades a posteriori; de hecho, no hay ninguna razón por la cual debieran sumar uno. Dicho de otro modo, dada una textura j , asociaremos a dicha textura unos grados de pertenencia (u_{1j}, \dots, u_{Ij}) , con $0 \leq u_{ij} \leq 1$ y $\sum_{i=1}^I u_{ij}$ no necesariamente igual a uno. En conclusión, los u_{ij} 's son *etiquetas posibilísticas* ([56]).

La primera opción para determinar en que clase asignar la textura x_0 , podría simplemente basarse en los valores posibilísticos $\mu^{(i)}(x_0)$ (también denotado habitualmente por u_{ix_0}). La regla de decisión podría ser clasificar x_0 en la clase con $\mu^{(i)}(x_0)$ mayor. Pero además, podríamos disponer de información previa acerca de la frecuencia de las clases. Sea π_i la posibilidad a priori de la clase i . Si reproducimos las ideas del teorema de Bayes, puede determinarse que x_0 pertenezca a la clase en la que se alcance el máximo $\pi_i \mu^{(i)}(x_0)$.

No obstante, visto en el contexto de conjuntos difusos, cobran sentido distintas alternativas tales como asignar la textura a más de una clase. Las funciones de pertenencia dan el grado de inclusión de la textura en cada una de las clases, por lo tanto, es lícito considerar un valor α en el intervalo $[0, 1]$ y determinar que la textura incluida en los α – cortes pertenezca a esas clases. Con lo cual, la textura inicial podría pertenecer a aquellas clases cuyo grado de pertenencia fuera mayor o igual que el valor α seleccionado. Recordemos que un α -corte de un conjunto difuso A (en nuestro caso, la clase i), es un conjunto (no difuso) A_α que contiene todos los elementos con un grado de pertenencia en A mayor o igual que α ([100]).

El presente clasificador, al igual que otros clasificadores supervisados, se asientan sobre el conjunto de entrenamiento. Este conjunto puede alcanzar una importante influencia sobre los grados de pertenencia finales o etiquetas finales. Obviamente, cambios en el conjunto de entrenamiento darán lugar a alteraciones de las funciones de pertenencia. Sin embargo, desconocemos hasta qué punto estas variaciones afectan a las funciones de pertenencia. Esta cuestión, nos lleva a examinar la variabilidad inherente en la selección del conjunto de aprendizaje. Supongamos que la clase i sigue consistiendo de n_i texturas, pero que ahora m_i texturas son aleatoriamente elegidas del total de las n_i y usadas con cualquiera de los procedimientos descritos previamente. En consecuencia, una selección aleatoria s , que es equiprobablemente escogida entre las $\prod_{i=1}^I \binom{n_i}{m_i}$ posibles elecciones, produce un $\mu^{(s)}$ aleatorio para cada textura, es decir, una variable aleatoria difusa [101]. Si la clasificación final se basa tan solo en una muestra, la variabilidad de esta variable aleatoria no se tendría en cuenta. Por ello, sería conveniente que el procedimiento de clasificación de texturas contemplara esta variabilidad.

A continuación se muestra en las figuras 3.2 y 3.3 un esquema de la metodología:

3.2. Experimentos y evaluación del procedimiento posibilístico

Como ya hemos explicado, un procedimiento de clasificación de texturas implica una etapa de extracción de características y otra de clasificación. La mayoría de las aproximaciones se centran en la etapa de extracción de características. Sin embargo, en nuestro caso el foco de interés está en la clasificación. Se han considerado cuatro algoritmos de extracción de características, las mismas que se usaron en el capítulo 2 para comparar

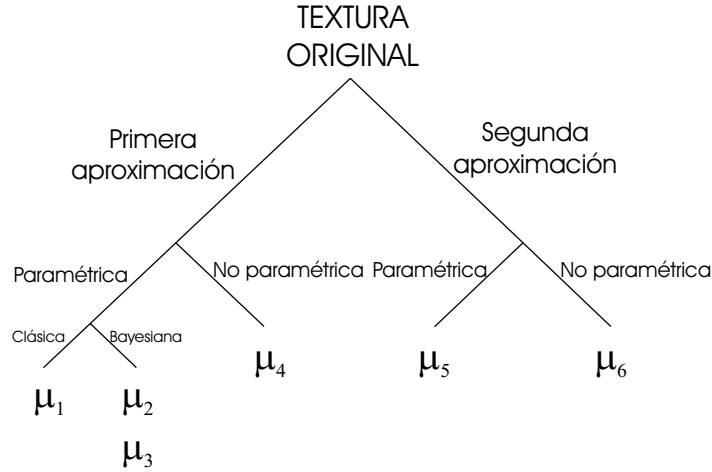


Figura 3.2: Distintas alternativas para definir la compatibilidad de un textura dada con un conjunto de prototipos (de una cierta clase)

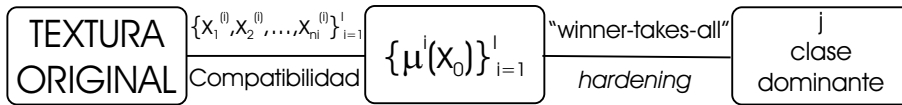


Figura 3.3: Metodología para la clasificación

la aproximación de los conjuntos aleatorios con otras aproximaciones clásicas: campos aleatorios de Markov (MRF), matrices de co-ocurrencia (LCM), dimensión fractal (FRD) y energías de Gabor (GCE). En este apartado, ilustraremos los resultados del clasificador posibilístico. Emplearemos la segunda aproximación, la basada en p-valores, y la aproximación no paramétrica, para construir las etiquetas posibilísticas. El test de aleatorización se construirá usando la distancia de Mahalanobis. Nuestra aproximación es intrínsecamente no paramétrica. No se ha asumido ninguna hipótesis en lo que respecta a la distribución de los descriptores de texturas. Esta aproximación es asimismo una metodología costosa computacionalmente.

La base de texturas con la que hemos trabajado es la usada por Kaplan [90]. Es la misma que la usada en el último experimento del capítulo 2 y el diseño experimental es también el mismo.

En la subsección 3.2.1 se da una comparación entre el clasificador de Bayes y el posibilístico y se considera el problema de la selección de prototipos que definen las clases. Varios ejemplos sintéticos se proponen en la subsección 3.2.2 para comparar un marco

probabilístico (Bayes) con el marco difuso (aleatorización). Finalmente, en la subsección 3.2.3 (quizá el aspecto más importante) se explica como usar nuestros conjuntos difusos para comparar (y quizá seleccionar) distintas características.

3.2.1. Hardening

En este apartado, hemos analizado la salida del clasificador posibilístico y la exactitud de la clasificación se ha comparado con la del clasificador de Bayes.

Como ya mencionamos antes, la medida usual para evaluar la efectividad de las características de las texturas es el porcentaje de clasificación correcta. Se requiere una partición con asignaciones a clases definidas, por lo que las etiquetas probabilísticas, difusas o posibilísticas tienen que transformarse en etiquetas *binarias*. La forma más común de convertir etiquetas *entre 0 y 1* en etiquetas *binarias* consiste en asignar la textura a la clase con el valor mayor (probabilidad a posteriori máxima en el caso del clasificador de Bayes o grado de pertenencia mayor en el caso del clasificador posibilístico) [56]. Así que llevaremos a cabo una comparación entre el clásico clasificador de Bayes y el procedimiento aquí presentado.

La tabla 3.1 presenta los porcentajes observados. Cada columna la encabeza el método de extracción de características empleado y se subdivide en dos columnas; la primera de ellas muestra los resultados del clasificador de Bayes y la segunda los del método de aleatorización. Cada fila muestra la exactitud para cada clase, dado como el porcentaje de muestras de test correctamente clasificadas. La última fila indica las medias globales.

En general, el porcentaje de clasificación correcta es menor para el clasificador propuesto que para el clasificador de Bayes. Una posible razón que explica esta circunstancia, sería el reducido tamaño muestral que define cada clase (recordemos que cada clase viene descrita por solamente 32 texturas). El efecto de este número en la estimación no es tan importante como en un test de aleatorización [99].

Para investigar la influencia de la selección de prototipos en el resultado final (lo cual no suele considerarse en la literatura de clasificación de texturas), hemos diseñado otro experimento en el que cada textura original se divide de 16 (la forma anterior también se incluye) maneras distintas en conjunto de entrenamiento y de prueba. La primera fila de la tabla 3.2 muestra el porcentaje de clasificación correcta medio, mínimo y máximo de estas 16 selecciones. Como puede apreciarse, de entre las 16 maneras de asignar prototipos y muestras test, una de las selecciones que daba peores resultados es la que aparecía en la tabla 3.1.

Otra posible consecuencia del tamaño muestral pequeño es que hay algunas texturas que no se clasifican en ninguna clase, ya que sus grados de pertenencia son cero para todas las clases. Estos casos se consideraron incorrectos a la hora de presentar los resultados de la tabla 3.1 y de la primera fila de la tabla 3.2. La segunda fila de la tabla 3.2 proporciona los porcentajes sin estas texturas, usando también las 16 selecciones.

Tabla 3.1: Porcentajes de clasificación correcta para cada clase de texturas, usando distintas características y los clasificadores de Bayes y aleatorización

Clase	MRF		LCM		FRD		GCE	
1	96.9	37.5	68.8	84.4	78.1	78.1	81.3	53.1
2	100	100	87.5	78.1	81.3	18.8	100	100
3	75	62.5	100	96.9	90.6	50	93.8	59.4
4	81.3	78.1	100	75	68.8	71.9	93.8	46.9
5	96.9	84.4	100	100	100	100	100	90.6
6	100	93.8	93.8	90.6	78.1	40.6	87.5	46.9
7	100	96.9	96.9	90.6	100	84.4	100	87.5
8	100	96.9	96.9	71.9	93.8	37.5	100	100
9	100	0	100	75	100	100	100	56.3
10	90.6	78.1	100	53.1	100	100	93.8	46.9
11	100	100	96.9	59.4	59.4	84.4	93.8	78.1
12	100	100	96.9	84.4	84.4	62.5	100	96.9
13	100	96.9	93.8	87.5	93.8	65.6	100	93.8
14	78.1	59.4	93.8	93.8	84.4	65.6	93.8	68.8
15	71.9	62.5	81.3	78.1	78.1	75	56.3	40.6
16	100	53.1	100	93.8	100	96.9	100	68.8
17	100	100	100	90.6	56.3	68.8	100	96.9
18	100	87.5	96.9	68.8	68.8	71.9	84.4	21.9
19	90.6	93.8	84.4	81.3	71.9	43.8	87.5	81.3
20	93.8	81.3	78.1	71.9	93.8	90.6	96.9	78.1
21	100	96.9	93.8	87.5	87.5	75	78.1	40.6
22	100	96.9	100	100	93.8	96.9	96.9	59.4
23	71.9	43.8	93.8	78.1	96.9	78.1	96.9	87.5
24	59.4	43.8	100	96.9	100	96.9	96.9	50
25	100	75	90.6	81.3	96.9	93.8	100	75
26	100	100	93.8	62.5	93.8	84.4	100	78.1
27	100	93.8	100	100	100	96.9	100	90.6
28	100	100	100	90.6	65.6	62.5	81.3	71.9
29	84.4	87.5	53.1	53.1	56.3	68.8	81.4	75
30	93.8	84.4	84.4	96.9	68.8	53.1	100	100
31	96.9	84.4	40.6	40.6	56.3	21.9	90.6	90.6
32	96.9	93.8	87.5	78.1	40.6	59.4	100	100
33	96.9	100	96.9	87.5	71.9	21.9	81.3	68.8
34	46.9	43.8	81.3	78.1	100	87.5	65.6	15.6
35	100	100	93.8	96.9	90.6	78.1	96.9	96.9
36	96.9	71.9	100	87.5	100	90.6	100	71.9
37	100	100	90.6	93.8	93.8	81.3	100	100
38	100	96.9	96.9	84.4	100	100	100	96.9
39	100	90.6	87.5	93.8	90.6	43.8	93.8	37.5
40	93.8	93.8	100	96.9	96.9	96.9	100	90.6
41	87.5	46.9	43.8	25	81.3	81.3	90.6	34.4
42	84.4	9.4	96.9	90.6	87.5	84.4	96.9	34.4
43	96.9	100	96.9	53.1	84.4	56.3	96.9	75
44	65.6	25	93.8	59.4	96.9	93.8	100	46.9
45	100	87.5	93.8	84.4	65.6	46.9	100	90.6
46	75	40.6	75	65.6	71.9	87.5	100	62.5
47	81.3	37.5	84.4	53.1	68.8	34.4	96.9	46.9
48	100	96.9	89.1	71.9	57.8	34.4	59.4	95.3
49	84.4	78.1	96.9	93.8	93.8	81.3	84.4	78.1
50	100	93.8	93.8	96.9	71.9	59.4	68.8	43.8
51	100	95.3	98.4	95.3	85.9	37.5	100	100
52	84.4	62.5	100	84.4	100	100	100	93.8
53	84.4	78.1	87.5	84.4	87.5	75	100	93.8
54	98.4	93.8	82.8	82.8	71.9	51.6	100	98.4
55	87.5	50	100	96.9	100	78.1	100	90.6
56	87.5	65.6	87.5	59.4	71.9	43.8	59.4	25
57	84.4	84.4	100	100	100	100	100	93.8
58	100	93.8	100	100	78.1	68.8	96.9	68.8
59	100	93.8	100	81.3	96.9	93.8	90.6	37.5
60	100	100	100	93.8	46.9	43.8	90.6	65.6
61	100	81.3	96.9	93.8	84.4	68.8	100	84.4
62	100	96.9	100	40.6	84.4	71.9	90.6	84.4
63	100	96.9	96.9	100	90.6	100	90.6	59.4
64	96.9	81.3	96.9	62.5	84.4	50	93.8	59.4
65	100	100	67.2	87.5	45.3	62.5	50	98.4
66	98.4	87.5	71.9	93.8	40.6	65.6	98.4	96.9
67	0	0	100	34.4	90.6	75	18.8	0
68	100	98.4	51.6	90.6	28.1	65.6	98.4	95.3
69	96.9	81.3	84.4	81.3	75	81.3	84.4	62.5
70	100	56.3	100	90.6	96.9	90.6	100	62.5
71	93.8	62.5	81.3	65.6	96.9	96.9	84.4	46.9
Media	92.2	77.9	89	80.6	79.8	71.5	90.5	70.9

Tabla 3.2: Media, mínimo y máximo de los porcentajes de clasificación correcta medios de las 16 selecciones (fila 1) y los mismos resultados calculados sin las texturas con grados de pertenencia iguales a cero en todas las clases para el método de aleatorización (fila 2)

MRF			LCM			FRD			GCE		
Media	Min	Max	Media	Min	Max	Media	Min	Max	Media	Min	Max
90.4	77.9	92.9	84	80.6	86.3	72.6	70.7	74.9	84.4	70.9	87.7
92.4	88.9	94.6	84.4	80.8	87.2	73.1	71.2	75	86.5	84	88.8

Un examen de la tabla 3.1 revela algunos puntos interesantes. En particular, analicemos las muestras test de la clase 9 (el caso más extremo) con MRF, cuyos porcentajes son 100 y 0 para el clasificador de Bayes y el método de aleatorización respectivamente. Las etiquetas posibilísticas para este conjunto de prueba son cero para todas las clases, mientras que el clasificador de Bayes concede un 1 a la clase 9 para todas las muestras test que pertenecían a esta clase. Por un lado, los resultados posibilísticos podríamos interpretarlos como que dichas texturas no pertenecen a ninguna clase. Por otro lado, las etiquetas probabilísticas 'confían' en la clase 9 ciegamente. La razón por la cual se dan esas probabilidades tan altas es la cercanía a las texturas que definen la clase 9, no una pertenencia real, como podemos comprobar de manera sencilla. Consideremos, por ejemplo, una textura del conjunto de prueba de la clase 9. Primeramente, calculamos las distancias de Mahalanobis entre dicha textura y cada una de las 71 clases (caracterizadas por su media y matriz de covarianzas estimadas). La distancia a la clase 9 es considerablemente más pequeña que al resto de clases (sobre 100 veces menor que para la segunda clase con distancia más pequeña). Este hecho explica el porqué de la disposición de probabilidades del clasificador de Bayes. En segundo lugar, realizamos un test para probar si la textura elegida se puede considerar como la media de la clase 9 (H_0). Tal y como se ha explicado en el ejemplo 6, bajo la suposición de que las características son gaussianas ($N_p(\mu, \Sigma)$), como nuestro clasificador de Bayes asume, podemos usar el estadístico T^2 de Hotelling [102]. De nuevo, \bar{x} y S denotarán la media y matriz de covarianzas muestrales de tamaño n (32), así

$$n(\bar{x} - \mu)'S^{-1}(\bar{x} - \mu) \sim T^2(p, n - 1), \quad (3.24)$$

donde

$$T^2(p, m) = mp/(m - p + 1)F_{p, m-p+1} \quad (3.25)$$

y $F_{p, m-p+1}$ denota la distribución F de Snedecor con p y $m - p + 1$ grados de libertad, siendo p el número de características de las que disponemos. El p -valor para la textura seleccionada es cero, lo cual conduce al rechazo de H_0 . Una ilustración más detallada de este tipo de situación se mostrará en el apartado 3.2.2.

3.2.2. Etiquetas probabilísticas versus posibilísticas : ejemplos sintéticos

Es evidente que se pierde mucha información en la operación de *hardening*, ya que no se refleja la confianza al hacer una elección. Por ejemplo, el clasificador de Bayes podría calcular que la probabilidad de una textura de estar en una cierta clase es 100% o 50%, y en ambos casos la asignación final sería la misma, aunque la confianza en la decisión del clasificador es muy distinta. Esta situación es incluso más acentuada para etiquetas posibilísticas porque las pertenencias son absolutas (es decir, no existe la restricción de que las pertenencias a lo largo de las clases deban sumar uno). Esta característica nos permite interpretar las pertenencias como un grado de tipicidad de una clase [103].

Los siguientes ejemplos simples (inspirados en el trabajo [103]) ilustran esta interpretación y muestran como un clasificador probabilístico no puede procesar ciertas instancias.

La figura 3.4 nos enseña dos muestras correspondientes a dos distribuciones normales con la misma matriz de covarianzas (la identidad) y medias $(0, 0)$ y $(6, 0)$ respectivamente. Aparecen marcados cinco puntos (con un círculo), $(-2.5, 0)$, $(2.5, 0)$, $(3, 0)$, $(3, 5)$, $(-5, 0)$. Las pertenencias, proporcionadas por el clasificador de Bayes y el método de aleatorización, se muestran en la tabla 3.3.

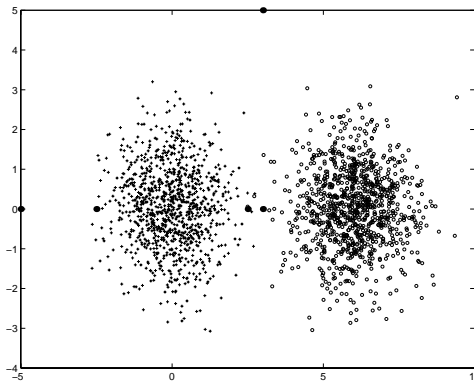


Figura 3.4: Ejemplo de dos muestras normales

Nuestro método de aleatorización produce valores de pertenencia similares (bajas) en la clase 1 para los puntos $(-2.5, 0)$ y $(2.5, 0)$, mientras que el clasificador de Bayes proporciona probabilidades diferentes para puntos igualmente típicos de la clase 1. Esto se origina debido a la proximidad de las muestras de la clase 2 y la restricción sobre las pertenencias (probabilidades), ya que han de sumar uno. A pesar de no ser puntos típicos (representativos) de la clase 1, obtienen una probabilidad muy alta de pertenecer a la clase 1. Esta circunstancia es incluso más notable en lo que respecta al punto $(-5, 0)$. Este punto no pertenece a ningún grupo, pero el clasificador de Bayes asigna toda la probabilidad a

Tabla 3.3: Pertenencias para Bayes y el método de aleatorización para el ejemplo mostrado en la figura 3.4

Puntos	Bayes		Método de aleatorización	
	Clase 1	Clase 2	Clase 1	Clase 2
(-2.5,0)	1	0	0.038	0
(2.5,0)	0.946	0.054	0.033	0
(3,0)	0.429	0.571	0.01	0.008
(3,5)	0.427	0.573	0	0
(-5,0)	1	0	0	0

la clase 1. Los otros dos puntos anómalos, (3, 0) y (3, 5), comparten la probabilidad entre ambas clases, pese a que el punto (3, 5) es un punto extremadamente atípico. Sin embargo, las etiquetas posibilísticas para estos puntos son cercanas a cero (de hecho, es cero para el punto más ruidoso).

El segundo ejemplo demuestra cómo las etiquetas probabilísticas son incapaces de distinguir un punto 'igualmente bueno'. La figura 3.5 representa una situación con dos clases circulares intersectándose. El punto *A*, la intersección de ambos clusters, es un 'buen' miembro de ambos grupos. El método posibilístico refleja este hecho, alcanzando un grado de pertenencia de 0,98 para ambos grupos. Sin embargo, el clasificador de Bayes asigna una probabilidad de 0,5 en los dos clusters.

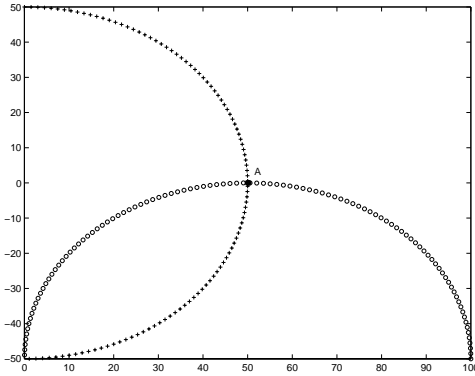


Figura 3.5: Ejemplo de dos agrupaciones circulares

3.2.3. Uso del método de aleatorización en la comparación de características de texturas

Hasta ahora, nuestros experimentos se han concentrado en la comparación entre el clasificador de Bayes y el clasificador posibilístico de una forma estándar. Hemos visto como el clasificador de Bayes proporcionaba mejores porcentajes de clasificación correcta. ¿Por qué utilizar entonces el método posibilístico? En primer lugar, basar la comparación en probabilidad a posteriori mayor frente al grado de pertenencia mayor (todo para el ganador) parece una elección demasiado simple, como se ha mostrado con los ejemplos sintéticos. En segundo lugar y más importante, las probabilidades a posteriori bayesianas no son correctas. Estos valores se basan en una suposición paramétrica muy fuerte: la de normalidad para las características de texturas. Se han realizado algunos tests de normalidad, y en todos los casos, la hipótesis nula de normalidad se rechaza con claridad. No obstante, nuestros p-valores obtenidos con la aleatorización se basan en una suposición no paramétrica y son ciertos.

A continuación, proporcionamos varios experimentos en los que comparamos distintos descriptores de texturas por medio del método de aleatorización.

Dado un valor α , consideraremos una textura correctamente clasificada si pertenece al α -corte de su propia clase. La figura 3.6 muestra, para cada α , el número de texturas correctamente clasificadas. Un comentario podría hacerse: las características fractales y de los campos aleatorios de Markov muestran un comportamiento similar que es claramente superior al de los dos restantes conjuntos de características.

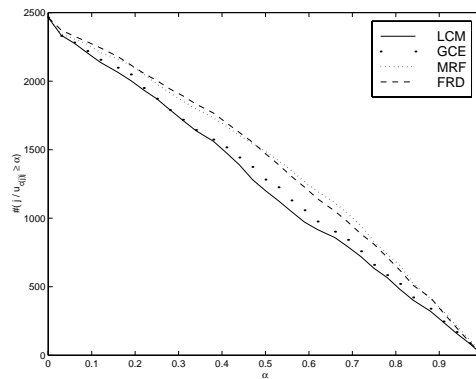


Figura 3.6: Número de clasificaciones correctas con distintas características, según el α -corte

Vamos a usar en la comparativa cuatro medidas evaluadoras distintas. Dada la textura j , la primera de estas medidas es

$$\alpha_j = u_{c(j)j}, \quad (3.26)$$

Tabla 3.4: Medias de las medidas

	FRD	LCM	MRF	GCE
α	0.4863	0.4884	0.4962	0.4944
n	1.2179	1.0503	1.0195	1.2459
d	0.2503	0.3496	0.4275	0.3441
r	0.3418	0.3957	0.4552	0.3814

donde $c(j)$ es la clase de la textura j . La segunda medida es

$$n_j = \#\{i : u_{ij} \geq \alpha_j\}, \quad (3.27)$$

es decir, el número de α_j -cortes que contienen a la textura j . Las medidas tercera y cuarta vienen definidas mediante

$$d_j = \alpha_j - \max_i \{u_{ij} : i \neq c(j)\} \quad (3.28)$$

y

$$r_j = \frac{\alpha_j}{1 + \sum_{i \neq c(j)} u_{ij}}. \quad (3.29)$$

Cuanto más elevadas sean α_j , d_j y r_j y más similares a uno las n_j , mejores son las características. En la tabla 3.4 aparecen las medias observadas. Las características MRF son las que dan los mejores resultados. Ohanian y Dubes [104] compararon las mismas características y obtuvieron malos resultados para las características MRF, en contra de sus suposiciones iniciales. Ellos lo atribuyeron al pequeño tamaño de sus imágenes (32×32 píxeles). Cada par de descriptores han sido comparados para cada medida, por medio de un test t apareado. Todos los p-valores obtenidos son casi nulos. Los valores medios son claramente diferentes.

Capítulo 4

Transformación lineal para la decorrelación estadística y perceptual en codificación de imágenes

¹ Como ya se dijo en el capítulo 1, las imágenes contienen dos tipos de redundancia: la estadística y la psicovisual. Para describir las relaciones estadísticas y perceptuales, emplearemos respectivamente las matrices de covarianzas y métricas perceptuales. Los elipsoides descritos por ambas matrices no están alineados, con lo cual debemos diagonalizar ambas matrices para tratar de eliminar los dos tipos de redundancia y obtener una buena representación. Estas ideas se presentarán en el primer apartado de este capítulo. En el siguiente apartado, se proporcionarán los detalles sobre el modelo visual que se ha asumido y que permite la definición de la matriz perceptual. Tras ello, se explicará la técnica empleada para diagonalizar simultáneamente las dos matrices. Por último, se comentarán los resultados de compresión obtenidos de la realización de diversos experimentos. Compararemos la transformación sugerida con dos representaciones estándar basadas en la matriz de covarianzas: la KLT (transformación de Karhunen-Loève o componentes principales) y la DCT, usando respectivamente cuantizadores del tipo Max-Lloyd [68] y el estándar JPEG. Estos experimentos ilustrarán la importancia de incluir a la vez las anisotropías del espacio y las características estadísticas, ya que debido a que el dominio final será euclídeo (por definición), el diseño del cuantizador se simplificará notablemente y además al mismo tiempo el uso de cuantizadores escalares quedará verdaderamente justificado (al tener una distorsión diagonal).

¹El presente capítulo constituye básicamente el trabajo '*Linear transform for simultaneous diagonalization of covariance and perceptual metric matrix in image coding*', enviado a Pattern Recognition [105].

4.1. Expresiones matriciales (estadísticas y perceptuales) y su interpretación

El propósito de la transformación es eliminar las relaciones estadísticas y perceptuales entre coeficientes, por tanto, en primer lugar dichas relaciones deben cuantificarse y formularse formalmente. Las dos matrices que definirán estas interacciones son: la matriz de covarianzas para las estadísticas y la matriz métrica perceptual para las psicovisuales.

4.1.1. Matriz de covarianzas

Las luminancias de una imagen en el dominio espacial pueden representarse mediante un vector A . Este vector podemos verlo como un conjunto de variables aleatorias. Las *desviaciones estadísticas* de un punto A_0 podemos describirlas por la matriz $\Sigma_A(A_0)$,

$$\Sigma_A(A_0) = E[(A - A_0) \cdot (A - A_0)^T] \quad (4.1)$$

donde E representa el valor esperado y el exponente T la transposición de vectores. Cuando A_0 es la media, esta matriz es la matriz de covarianzas Σ_A en el dominio (espacial) A .

El elemento de la diagonal de la matriz de covarianzas, σ_{ii}^2 , es la varianza del coeficiente i -ésimo (variable i -ésima). El elemento σ_{ij} representa la covarianza entre los coeficientes i -ésimo y j -ésimo, una relación de segundo orden.

4.1.2. Matriz métrica perceptual

Si asumimos una norma L^2 [64, 106], la *desviación perceptual* de A_0 debido a una distorsión ΔA viene determinada mediante,

$$d(A_0, A_0 + \Delta A)^2 = \Delta A^T \cdot W_A(A_0) \cdot \Delta A = \sum_i W_{ii} \Delta A_i^2 + \sum_{i \neq j} W_{ij} \Delta A_i \Delta A_j \quad (4.2)$$

donde $W_A(A_0)$ es la métrica perceptual del dominio A en el punto A_0 .

Como puede apreciarse en la ecuación 4.2, los componentes de la diagonal de la métrica perceptual representan las contribuciones de cada coeficiente a la distorsión global. Los elementos distintos de cero de fuera de la diagonal inducen contribuciones adicionales a la distorsión debido a las combinaciones de desviaciones en diferentes dimensiones, es decir, representan interacciones perceptuales entre características que modifican la distorsión. Esta es una forma bastante conveniente de representar lo que comúnmente se refiere como *enmascaramiento* [107]: la distorsión de un coeficiente podría enmascarar la distorsión subjetiva de otro coeficiente. Este tipo de interacciones normalmente no suele considerarse en la distorsión global en *transform coding*.

4.1.3. Interpretación geométrica de estas matrices

Las matrices de covarianzas y perceptuales tienen significados diferentes, ya que representan distintas relaciones. Ambas matrices son definidas positivas, con lo cual podemos asociar un elipsoide a cada una de ellas. Por un lado, Σ describe la forma de la distribución de las muestras de imágenes alrededor de A_0 , proporciona información acerca de la dispersión de los datos. Por otro lado, W describe la forma del locus (elipsoidal) de patrones perceptualmente equidistantes de A_0 (distorsión constante en la ecuación 4.2). W describe la geometría subyacente del espacio de características: los elementos de fuera de la diagonal de W representan las interacciones de segundo orden perceptuales. Es importante señalar dos hechos (véase la figura 4.1).

El primero es que los elipsoides Σ y W no están alineados con los ejes de la representación en el dominio espacial. Esto es así en las imágenes naturales porque, por un lado, los valores de la luminancia, A_i , en cada localización espacial, i , están altamente correlacionados con los valores, A_j , de las localizaciones vecinas, j [108]. Por otro lado, existe también una fuerte correlación perceptual entre píxeles vecinos [109], dando lugar a métricas altamente no diagonales en el dominio espacial [66, 71]. Estas interacciones entre coeficientes significan que la representación en el dominio espacial es bastante inadecuada para una cuantización escalar. El objetivo sería transformar a un nuevo dominio a , para eliminar estas correlaciones antes de la cuantización escalar (ese es el núcleo de *transform coding*, aunque en la aproximación estándar únicamente se busca la decorrelación estadística).

En segundo lugar, resaltemos que estos elipsoides no están alineados entre ellos. Esto significa que las transformaciones KLT o DCT basadas en la diagonalización de Σ no implican una diagonalización de W .

Como se mostrará más adelante, los métodos clásicos de diseño de cuantizadores asumen una distorsión 'diagonal' (se trabaja como si las distorsiones en un coeficiente no involucraran distorsiones en otros coeficientes). Pero si W no es diagonal en la representación elegida, dichos resultados no pueden ser estrictamente aplicables. Esta es la razón por la cual, en principio, la aproximación estándar podría mejorarse teniendo en cuenta W en la selección de la representación. En consecuencia, la representación adecuada sería aquella que no sólo diagonalizara Σ , sino también W . En este caso de diagonalización simultánea, la cuantización escalar será efectiva (ya que habremos suprimido las relaciones estadísticas y perceptuales), y los resultados clásicos de diseño de cuantizadores serán estrictamente aplicables.

4.1.4. Cambios matriciales bajo aplicaciones lineales

Es conveniente saber cómo cambian las matrices anteriores cuando aplicamos una transformación lineal. Antes de pasar a especificarlo explícitamente, sería bueno recordar que es una métrica riemanniana.

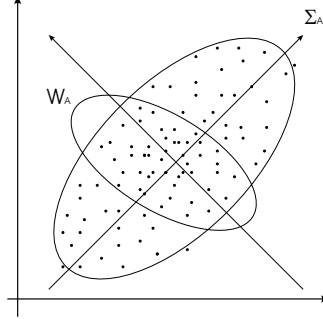


Figura 4.1: Elipsoides que describen la distribución de los datos y la geometría del espacio.

Definición 2 Una métrica riemanniana en una región del espacio \mathbb{R}^n es una forma cuadrática definida positiva, que se define sobre vectores con origen en cada punto P de la región y que depende suavemente de P .

Esta definición puede formularse más explícitamente:

Definición 3 Una métrica riemanniana en una región de un espacio, relativa a unas coordenadas (z_1, \dots, z_n) es una familia de funciones suaves $g_{ij} = g_{ij}(z_1, \dots, z_n)$, $i, j = 1, \dots, n$, con las siguientes dos propiedades:

1. la matriz (g_{ij}) es definida positiva
2. si (y_1, \dots, y_n) son nuevas coordenadas para la región, y $z_i = f_i(y_1, \dots, y_n)$, $i = 1, \dots, n$, entonces la métrica riemanniana relativa a estas nuevas coordenadas, viene representada por la familia de funciones $g'_{ij} = g'_{ij}(y_1, \dots, y_n)$, $i, j = 1, \dots, n$ dada por

$$g'_{ij} = \sum_{k=1}^n \frac{\partial f_k}{\partial y_i} \sum_{l=1}^n g_{kl} \frac{\partial f_l}{\partial y_j}. \quad (4.3)$$

La segunda propiedad puede reescribirse en forma matricial:

$$G_y = (g'_{ij}) = J^T (g_{ij}) J = J^T G_z J, \quad (4.4)$$

donde G_y sería la matriz métrica en el dominio y y con elementos g'_{ij} , G_z la matriz métrica en el dominio z con elementos g_{ij} y J denota la matriz jacobiana:

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial y_1} & \frac{\partial f_1}{\partial y_2} & \cdots & \frac{\partial f_1}{\partial y_n} \\ \frac{\partial f_2}{\partial y_1} & \frac{\partial f_2}{\partial y_2} & \cdots & \frac{\partial f_2}{\partial y_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial y_1} & \frac{\partial f_n}{\partial y_2} & \cdots & \frac{\partial f_n}{\partial y_n} \end{pmatrix} \quad (4.5)$$

La propiedad de la matriz (g_{ij}) de ser definida positiva, simplemente significa que $\zeta^T G \zeta > 0$ para vectores ζ distintos de cero, es decir, que la forma cuadrática es definida positiva.

En [110] (por ejemplo) se cubre más extensamente este concepto.

Por lo que acabamos de exponer, para un vector A_0 (cuya métrica correspondiente sería $W_A(A_0)$), si $a_0 = L \cdot A_0$, la métrica en a_0 sería (véanse la ecuación 4.4 y [110]):

$$W_a(a_0) = (L^{-1})^T \cdot W_A(A_0) \cdot L^{-1}. \quad (4.6)$$

Las matrices de covarianzas, en cambio, sufren una modificación distinta. Sea A un vector aleatorio (con matriz de covarianzas Σ_A) que transformaremos en a mediante la transformación lineal L , $a = L \cdot A$. Entonces, la matriz de covarianzas para a será [102]:

$$\Sigma_a = L \cdot \Sigma_A \cdot L^T. \quad (4.7)$$

4.2. Modelo visual y matriz métrica perceptual

El paso previo antes de la definición de la matriz métrica perceptual, es introducir el modelo perceptual que se ha asumido. El modelo estándar de análisis de imagen a bajo nivel en el sistema visual humano tiene dos etapas básicas [107, 106]. En la primera etapa, la imagen de entrada A (definida en el *dominio espacial*) se transforma en un vector, $a = T \cdot A$ (con componentes a_f , $f = 1 \dots M$), en un dominio frecuencial local (el *dominio transformado*) mediante un banco de filtros lineales, T . Tras ello, un conjunto de mecanismos responde a cada coeficiente de la señal transformada, dando lugar a una salida $r = R(a)$, que es la representación de la imagen en el *dominio respuesta*. La figura 4.2 muestra gráficamente este esquema.

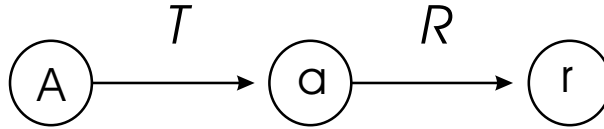


Figura 4.2: A es la imagen de entrada. Cada una de sus componentes, A_x , representa la luminancia en la posición x . A través de un conjunto de filtros pasa banda, pasamos A a un dominio frecuencial a . Aplicamos a los coeficientes transformados (a_f) una transformación no lineal para obtener la representación en el dominio respuesta (r).

La primera transformación perceptual lineal T es similar a la clase de transformaciones que se emplean en codificación de imágenes. En particular, el modelo de transformada perceptual T que hemos empleado (debido a su comportamiento similar con la transformada T [111, 112, 113], a su vez que por su testado uso como transformada previa a la aplicación de resultados perceptuales [70, 114, 115, 116, 117]) es la transformada discreta de coseno local (denotada como DCT de aquí en adelante), seguida por una normalización de cada componente por la componente continua (el conocido coeficiente DC - coeficiente(0,0) [57], que da la luminancia media del bloque), de forma que las amplitudes quedan expresadas

en contrastes [67]. Puesto que no se perciben igualmente todas las funciones base, se incluye un procesado adicional (la transformación R) para explicar estas heterogeneidades. Los modelos HVS (*sistema visual humano*) suponen que todas las componentes del vector r son igualmente importantes y que no existe ninguna interacción perceptual entre ellas [64, 107, 118], por lo tanto el dominio respuesta es euclídeo. El modelo de respuesta que hemos usado es básicamente el modelo de normalización de la energía de las referencias [119, 120, 107, 118] en el que la energía de cada coeficiente transformado (en contraste) se normaliza mediante una suma pesada de la energía de sus vecinos. La dependencia con los coeficientes vecinos viene dada por la convolución con un kernel de interacción h

$$r_i = \alpha_i \frac{|a_i|^2}{\beta_i + (h * |a|^2)_i}, \quad (4.8)$$

donde el índice i corresponde a la frecuencia espacial. En este trabajo, se ha añadido un término lineal pequeño para reproducir la CSF (función de sensibilidad al contraste [121]), así,

$$r_i = \frac{\alpha_i}{100} |a_i| + \alpha_i \frac{|a_i|^2}{\beta_i + (h * |a|^2)_i}. \quad (4.9)$$

La figura 4.3 muestra los parámetros de este modelo (no lineal) de normalización de la energía, así como un ejemplo de la respuesta para diversas funciones base de diferentes frecuencias (*cpd* denota ciclos por grado).

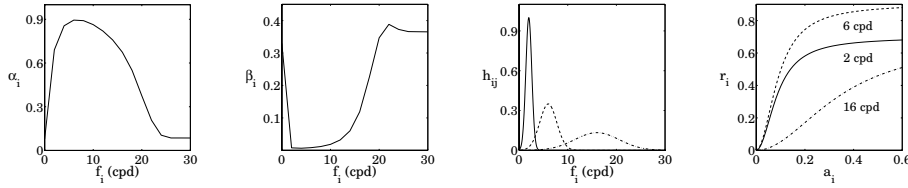


Figura 4.3: Parámetros del modelo visual y funciones no lineales de la respuesta. Los valores presentados en estas figuras asumen que la amplitud de los coeficientes está expresada en contraste (amplitud sobre luminancia media). Los ejemplos de respuesta de la última figura muestran el comportamiento básico (sigmoideal) de la ecuación 4.9, aunque no son generales puesto que la respuesta de un coeficiente depende del fondo (depende de los valores de los coeficientes vecinos). En concreto, estas curvas se calcularon para el caso particular de no contar con ningún patrón de enmascaramiento adicional (fondo cero).

Los valores de α y β se ajustaron para reproducir los umbrales de discriminación de amplitud sin enmascaramiento entre coeficientes que se midieron en el laboratorio de Óptica [122] (datos experimentales tipo Legge [123]). Se ha introducido un kernel gaussiano dependiente de la frecuencia según los resultados de las referencias [113, 64, 107, 118],

$$h_{ij} = k_i e^{-\frac{(f_i - f_j)^2}{\sigma(f_i)^2}} \quad (4.10)$$

donde f_i significaría la frecuencia espacial del coeficiente a_i , $\sigma(f_i)$ es la amplitud variable del kernel, $\sigma(f_i) = \frac{1}{3}|f_i| + 0,05$, con $|f|$ en ciclos/grados, y k_i es una constante para obtener un kernel de volumen unitario.

Suponiendo las transformaciones T y R anteriores y una métrica perceptual euclídea (identidad) en el dominio respuesta, podemos obtener la métrica perceptual en el dominio frecuencial si consideramos las propiedades de una métrica riemanniana bajo un cambio de sistema de coordenadas [110, 66] (la definición y algunas propiedades básicas de métricas riemannianas ya se han presentado en la subsección 4.1.4).

Si J denota la matriz jacobiana de la transformación R y

$$J_{ij} = \frac{\partial R_i}{\partial |a_j|} = \frac{\alpha_i}{100} \delta_{ij} + 2\alpha_i \left(\frac{|a_i|}{\beta_i + (h * |a|^2)_i} \delta_{ij} - \frac{|a_i^2 \cdot a_j|}{(\beta_i + (h * |a|^2)_i)^2} h_{ij} \right), \quad (4.11)$$

entonces la métrica en el correspondiente dominio transformado a es,

$$W_a = J^T \cdot J \quad (4.12)$$

La figura 4.4 muestra la métrica perceptual proveniente de la expresión 4.12, reemplazando J por la media de las matrices jacobianas de un conjunto de imágenes, \bar{J} .

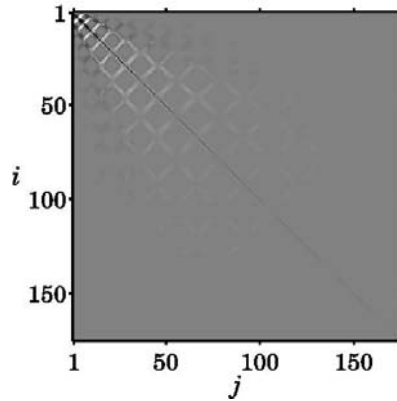


Figura 4.4: Métrica perceptual en el dominio transformado. Esta figura muestra la métrica perceptual con \bar{J} . Los píxeles oscuros y claros indican valores positivos y negativos respectivamente. Para resaltar la visibilidad de los elementos fuera de la diagonal, se le ha elevado a un cierto exponente.

El significado cualitativo de los elementos de la métrica, que dan las relaciones entre los distintos coeficientes de los vectores de características, depende de cómo se escaneen los 2D-DCT para construir los vectores de características 1D. La figura 4.5 explica el escaneo en zigzag que se ha aplicado a los 2D DCT. De acuerdo con el escaneo en zigzag,

el significado frecuencial de los elementos de la diagonal de h , J , W y Σ (cuando se definen en el dominio transformado) se incrementa progresivamente desde cero a la frecuencia de Nyquist.

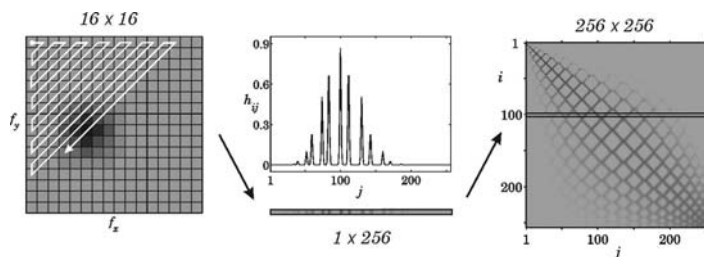


Figura 4.5: Escaneo DCT en zigzag. Una forma conveniente de escanear los 2D DCT es el sistema en zigzag que se usa en el estándar JPEG [57] porque agrupa los coeficientes de frecuencia similar. En el vector final, la frecuencia se incrementa progresivamente desde la componente continua (DC) en el primer coeficiente hasta la frecuencia Nyquist para el último coeficiente, es decir, $(32,32)$ *cpg* (*ciclos por grados*) si la frecuencia de muestreo en cada dirección es 64 *cpg*. En la figura 4.5 se ha escaneado en zigzag el kernel para el coeficiente $f_0 = (18,12)$ *cpg* y se ha introducido en su correspondiente posición en h . Según este esquema de escaneo, el coeficiente de frecuencia f_0 está en la posición $i = 100$, con lo que los valores que dan las relaciones a_{100} con a_j para $j = 1, \dots, 256$ forman un vector fila que ocupa la fila $i = 100$ para aplicarse sobre los vectores columna y dar $(h * |a|^2)_{100}$.

A partir de la figura 4.4, es claramente apreciable que la relevancia perceptual relativa de los coeficientes de la transformación depende en gran medida de la frecuencia (la diagonal de W tiene forma de pasa baja), es decir, el dominio frecuencial es perceptualmente anisotrópico. También, se aprecia que los coeficientes transformados no son perceptualmente independientes ya que W no es diagonal, o sea, las direcciones perceptualmente privilegiadas del dominio de frecuencias no están alineadas con los ejes del espacio. Esto implica que se necesita una transformación adicional para eliminar la *correlación* perceptual (así como la estadística) ente los coeficientes transformados y poder procesarlos individualmente después.

Nótese que J (y, por tanto, la métrica) cambia con la entrada y rigurosamente la transformación de decorrelación debería ser local. Sin embargo, en este capítulo asumiremos que una única métrica sirva como representante del resto de métricas. En el capítulo 5, sí que tendremos en cuenta la variación de la métrica, con lo cual incrementaremos la etapa de cálculo, pero mejoraremos los resultados obtenidos. Por el momento, los buenos resultados obtenidos en la sección 4.4 confirmarán que esta suposición podría ser válida como una primera (y más sencilla) opción. Se ha elegido la media de matrices jacobianas de un conjunto de imágenes \bar{J} , para calcular la métrica modelo que se usará en los experimentos. Se exploraron además otras opciones, tales como la matriz jacobiana de la media

de un grupo de imágenes, para resumir las métricas en una sola matriz, pero los resultados finales no variaron sustancialmente. La condensación de las distintas métricas en una única métrica, nos permitirá diseñar una transformación global (no local), lineal que diagonalice ambas matrices: la de covarianzas y la métrica perceptual. De esta manera, obtendremos un dominio en el que las características estén decorreladas estadística y perceptualmente.

4.3. Diagonalización simultánea de la métrica perceptual W_a y la matriz de covarianzas Σ_a

El procedimiento que proponemos, podría considerarse conceptualmente similar a las transformaciones de Box-Cox para conseguir normalidad [124]. Así, para poder aplicar estrictamente diversos procedimientos ampliamente estudiados, primero se obtendría la normalidad de los datos, lo que ahorraría desarrollar nuevos procedimientos. En nuestro caso, actuaremos análogamente: buscaremos cumplir las hipótesis necesarias de la teoría desarrollada. Para poder aplicar estrictamente los desarrollos incluidos en el clásico *transform coding*, llevaremos nuestros datos a un dominio euclídeo, para después aplicar la transformación estándar de Karhunen-Loève.

Sea a un vector en el ya mencionado dominio transformado, a . El proceso de diagonalización se realizará como sigue:

(1) Primeramente, W_a se *blanquea* mediante una transformación que llamaremos T_P y que vendrá definida por:

$$a^p = J \cdot (a - \bar{a}), \quad (4.13)$$

donde \bar{a} es la media de un conjunto de imágenes transformadas. De esta forma, Σ_{a^p} es $J \cdot \Sigma_a \cdot J^T$ (aplicamos la ecuación 4.7) y W_{a^p} es la identidad (se aplican las ecuaciones 4.6 y 4.12). El superíndice p de a^p , indica que estamos en un dominio euclídeo, decorrelado perceptualmente aunque no estadísticamente.

(2) En segundo lugar, aplicamos una transformación ortonormal para diagonalizar Σ_{a^p} , la transformada Karhunen-Loève (componentes principales). Esto es,

$$a^{sp} = T_K \cdot a^p \quad (4.14)$$

donde T_K^T y Λ son las matrices de vectores y valores propios de Σ_{a^p} y el superíndice sp de a^{sp} , indica que estamos en un dominio decorrelado perceptual (euclídeo) y estadísticamente. Así,

$$\Sigma_{a^p} \cdot T_K^T = T_K^T \cdot \Lambda \quad \text{y} \quad T_K \cdot T_K^T = I \quad (4.15)$$

Entonces,

$$\Sigma_{a^{sp}} = T_K \cdot \Sigma_{a^p} \cdot T_K^T = \Lambda \quad (4.16)$$

$$W_{a^{sp}} = (T_K^T)^{-1} \cdot W_{a^p} \cdot T_K^{-1} = (T_K^T)^{-1} \cdot I \cdot T_K^{-1} = I \quad (4.17)$$

Así se tiene que ambas matrices se diagonalizan. La combinación de los dos pasos nos proporciona la transformación completa,

$$a^{sp} = T_K \cdot J \cdot (a - \bar{a}) \quad (4.18)$$

La figura 4.6 muestra un ejemplo gráfico bidimensional (dos coeficientes) de este proceso.

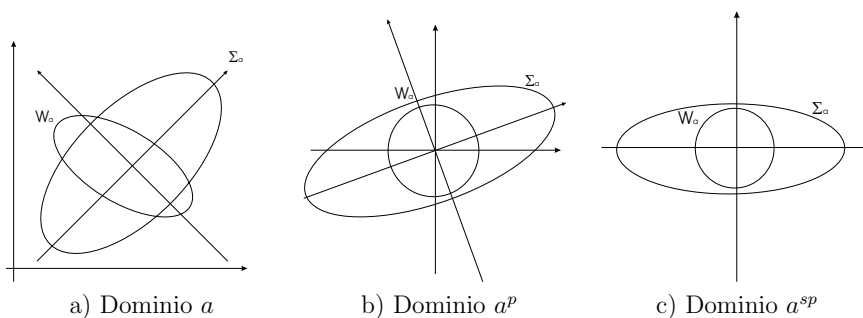


Figura 4.6: Diagonalización simultánea de W_a y Σ_a

4.4. Resultados de compresión

Se han llevado a cabo una serie de experimentos para comparar la transformación propuesta con el estándar JPEG y una aproximación completamente estadística. Hasta ahora, únicamente se ha considerado la transformación, sin embargo un codificador lo forman tres partes: transformación, cuantización y asignación de códigos. En este trabajo, se han tenido en cuenta dos aproximaciones, según la forma en la que se ha medido el error del cuantizador.

En el apartado 4.4.1 se revisarán los dos tipos de cuantizadores empleados. Los detalles de la implementación se analizarán en la subsección 4.4.2. Finalmente, en la subsección 4.4.3 presentaremos varios ejemplos de resultados de compresión sobre algunas imágenes típicas.

4.4.1. Diseño del cuantizador

Como ya hemos dicho, se han usado dos criterios a la hora de diseñar el cuantizador. La aproximación estándar para diseñar un codificador (transform coder) se basa en la minimización del error cuadrático medio (mean square error -MSE), entre la amplitud original de cada coeficiente transformado y su versión cuantizada. Para una variable aleatoria dada,

X , y un cuantizador $Q = \{y_i, R_i; i = 1, 2, \dots, N\}$, el MSE viene dado por

$$MSE = \sum_{i=1}^N \int_{R_i} (x - y_i)^2 f_X(x) dx \quad (4.19)$$

donde N es el número de niveles de cuantización, y_i es el i -ésimo nivel de cuantización, R_i es la región de cuantización correspondiente al nivel i -ésimo y $f_X(x)$ es la función de densidad de probabilidad de X . Esta distorsión (ampliamente usada) podría modificarse para considerar el significado físico del coeficiente X y así definir un error cuadrático medio perceptual (PMSE):

$$PMSE = \sum_{i=1}^N \int_{R_i} (x - y_i)^2 W(y_i) f_X(x) dx, \quad (4.20)$$

donde $W(x)$ es un peso que depende de la entrada x .

Bajo la aproximación asintótica de la cuantización (cuantizadores regulares con resolución alta: N es muy grande y las celdas de cuantización muy pequeñas [68]), la función de densidad puntual (point density function) óptima sería:

$$\lambda(x) = \frac{(W(x)f_X(x))^{1/3}}{\int (W(x)f_X(x))^{1/3} dy} \quad (4.21)$$

y la expresión asintótica para la distorsión media con esta función de densidad puntual es:

$$PMSE_{opt} = \frac{\sigma_X^2}{12N^2} \left(\int (W(\sigma_X y) \tilde{f}_X(x))^{1/3} dy \right)^3 = \frac{\sigma_X^2}{N^2} H \quad (4.22)$$

donde $\tilde{f}_X(y) = \sigma_X f_X(\sigma_X y)$ es la función de densidad normalizada con varianza unidad.

Como tenemos que cuantizar un conjunto de coeficientes, se ha de determinar el número de bits asignados para cada uno de estos coeficientes. Para este fin, suele ser muy común suponer que el error total del sistema de codificación es la suma (a veces pesada) de las distorsiones medias.

$$D = \sum_{i=1}^M PMSE_i(b_i), \quad (4.23)$$

donde $PMSE_i(b_i)$ denota el PMSE incurrido al cuantizar óptimamente el coeficiente i -ésimo con b_i bits de resolución.

Esta suposición se sostiene, cuando una *distorsión en un coeficiente no genera distorsiones en otros coeficientes*, es decir, cuando se han eliminado las interacciones entre coeficientes. Éste era el objetivo de nuestra transformación.

La asignación de bits óptima si se usan las aproximaciones de tasa alta (*high rate*) y $PMSE$ es:

$$b_i = \frac{B}{M} + \frac{1}{2} \log_2(\sigma_i^2 H_i) - \frac{1}{2M} \sum_{i=1}^M M \log_2(\sigma_i^2 H_i), \quad (4.24)$$

donde b_i es el número de bits usados para codificar el coeficiente i y B es el número total de bits disponibles. El subíndice i indica cantidades asociadas al coeficiente i -ésimo. Puede verse [71] para una descripción detallada de estos resultados.

El criterio de diseño antes visto basado en la media, no puede garantizar una realización subjetiva satisfactoria para una imagen particular. Por ello, para prevenir errores perceptuales grandes en imágenes individuales debidos a valores de los coeficientes fuera de lo normal (outliers), el error total podría evaluarse por medio de una distorsión basada en el *peor caso* [68]. De esta manera, el *error perceptual máximo* (*Maximum Perceptual Error* -MPE) en cada coeficiente quedará restringido. Un factor clave en una medida de peor caso es que los valores depende tan sólo del soporte de la función de densidad y no de la distribución [68]. Examinemos las expresiones (análogas) que adoptan las ecuaciones antes presentadas para la aproximación basada en el MPE. Los detalles pueden encontrarse en [71, 117].

Bajo la suposición tasa alta, la función de densidad puntual óptima es:

$$\lambda(x) = \frac{W(x)^{1/2}}{\int W(x)^{1/2} dy} \quad (4.25)$$

y el error máximo perceptual con la función de densidad puntual óptima es:

$$MPE_{opt} = \frac{1}{4N^2} \left(\int W(y)^{1/2} dy \right)^2. \quad (4.26)$$

Fijando la misma distorsión máxima para cada coeficiente, $MPE_{opt} = k^2$, la asignación de bits óptima viene dada por:

$$b_i = \log_2 N_i = \log_2 \left(\frac{1}{2k} \int W(y)^{1/2} dy \right), \quad (4.27)$$

donde N_i es el número de niveles de cuantización para el coeficiente i -ésimo.

El criterio *MPE* implica una distribución perceptualmente uniforme de los niveles de cuantización disponibles. Aunque todas estas aproximaciones solamente se sostienen en el caso de alta resolución, con frecuencia resultan ser una aproximación razonable para los casos de media y baja resolución [68].

4.4.2. Detalles de implementación

La transformación propuesta con los dos cuantizadores se comparó con JPEG y una aproximación completamente estadística, con lo que se ha asumido la hipótesis ampliamente admitida de que cada bloque de la imagen es la realización de un mismo modelo [57]. Para calcular la KLT (la matriz de covarianzas), se utilizó como muestra un conjunto de imágenes (8 bits/píxel) bien conocidas (la mayoría de las cuales las podemos encontrar en <http://sipi.usc.edu/services/database/Database.html>). Estas imágenes se partitionaron

en bloques de 16×16 y mediante la ordenación de los píxels dentro del bloque se obtuvieron los vectores que usamos para estimar la matriz de covarianzas. Se calculó la KLT, a pesar de que la DCT es una muy buena aproximación de esta transformación para el caso de imágenes naturales [125, 108]. Las imágenes del apartado 4.4.3 se han segmentado asimismo en bloques a los que se les ha aplicado la KLT calculada. Los coeficientes transformados se han cuantizado con un cuantizador MSE (ecuación 4.19). Nos referiremos a este procedimiento como KL-MSE.

Nuestra transformación (DCT \Rightarrow paso a contrastes $\Rightarrow T_p \Rightarrow T_K$) está también formada por una transformación KL (T_K). Ésta se calculó previamente a través de la misma muestra anterior. Por tanto, en este caso no depende de la imagen de entrada, sino que permanece fija. Como ya comentamos, se han contemplado dos opciones para el diseño del cuantizador con nuestra transformada. El primero es un cuantizador PMSE. El peso $W(x)$ que aparece en la ecuación 4.20 representa el elemento de la diagonal de la matriz métrica perceptual que corresponde con el coeficiente que está siendo cuantizado. Puesto que la matriz métrica perceptual en el dominio a^{sp} es la identidad, el PMSE se reduce simplemente al MSE con nuestra transformación. Por otra parte, el cuantizador MPE también se simplifica enormemente. Se convierte en uno de los cuantizadores escalares más habituales: el cuantizador uniforme. En lo que respecta a la asignación de bits, la distribución del número de niveles de cuantización depende exclusivamente del rango de cada coeficiente. Nos referiremos a estas dos opciones como T-MSE y T-MPE respectivamente.

Los resultados que se han visto en el apartado 4.4.1 son estrictamente aplicables bajo la aproximación de tasa alta [68, 126]. Para diseñar los cuantizadores MSE con baja resolución (para KL-MSE y T-MSE), se ha empleado el método LBG o algoritmo de Lloyd [68, 127]. Los resultados asintóticos se usaron como inicialización del algoritmo iterativo. Los cuantizadores finales fueron bastante consistentes con las suposiciones asintóticas. Como hemos dicho anteriormente, para T-MPE se emplea un cuantizador uniforme (el paso es constante), en lugar de un cuantizador de Lloyd-Max. La asignación de bits para todas las alternativas se determinó por medio de un algoritmo de asignación voraz (restringido a enteros) [68] basado en la asignación secuencial de un bit al coeficiente con mayor distorsión en cada iteración. Las distorsiones vienen dadas por las ecuaciones 4.22 y 4.26, según el caso. En todos los esquemas, el coeficiente DC se codifica por separado usando DPCM (*differential pulse code modulation*, [57]).

4.4.3. Imágenes decodificadas

Para evaluar la destreza de estos esquemas, vamos a mostrar los resultados de compresión de tres imágenes estándar (no incluidas en el conjunto de entrenamiento). Todos los esquemas se usaron con la misma tasa de compresión (0.5 bpp). Las imágenes se muestran en la figura 4.7.

La bondad de los distintos esquemas se ha evaluado tanto cualitativa como cuantitativamente. Las figuras 4.8 y 4.9 muestran las imágenes decodificadas, con las distintas aproximaciones.

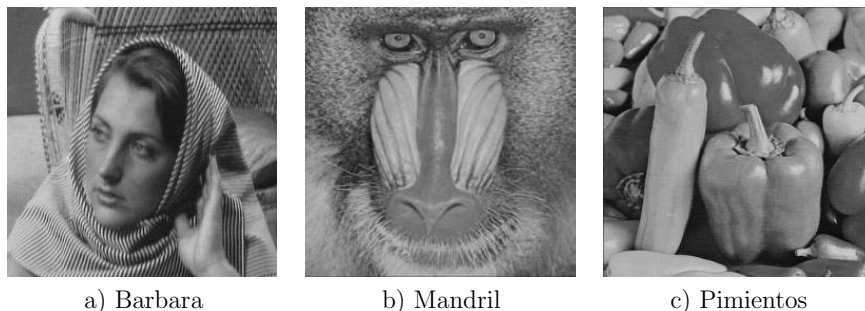


Figura 4.7: Imágenes usadas para evaluar la bondad de las aproximaciones: (un primer plano de) Barbara, mandril (baboon) y pimientos (peppers)

Estas imágenes nos permiten apreciar la correspondiente calidad subjetiva. A efectos de comparación, también se han empleado algunas medidas de calidad. Una medida objetiva de calidad que se utiliza popularmente es el cociente señal-ruido (*peak-to-noise-ratio* -PSNR [69]). Se define como sigue

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \quad (4.28)$$

donde MSE es el error cuadrático medio (mean square error), definido como:

$$MSE = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (f(x, y) - g(x, y))^2, \quad (4.29)$$

con M y N las dimensiones de las imágenes (en las direcciones horizontal y vertical) y f y g las dos imágenes a comparar.

La interpretación de esta medida es que cuanto mayor sea el PSNR, mejor será la calidad de la imagen procesada.

No obstante, esta medida de fácil computación no siempre es una medida adecuada de la calidad subjetiva [69]. Como vimos en la sección 4.2, la percepción visual del sistema visual humano no es tan simple. Por tanto, se han considerado igualmente dos medidas objetivas basadas en la percepción visual humana. Estas distancias perceptuales entre imágenes se describieron en [66]. En este trabajo se adoptaron los valores $\beta = 2$ y $\beta = \infty$ para el llamado índice del sumatorio. Estas distancias se denotarán como D_2 y D_∞ . Cuanto más cercanas a cero sean, mejor será la calidad. Los resultados se presentan en la tabla 4.1.

Pasemos a analizar los resultados. Las figuras 4.8 y 4.9 muestran varios comportamientos. Las imágenes JPEG están borrosas debido al truncamiento de coeficientes de altas frecuencias. Los detalles de frecuencias altas, de alto contraste (la ropa y sillón de Bárbara son ejemplos claros) se han perdido. Estas observaciones pueden percibirse también en la figura 4.11 (el original puede verse en la figura 4.10), en la que se muestra el hocico del mandril.

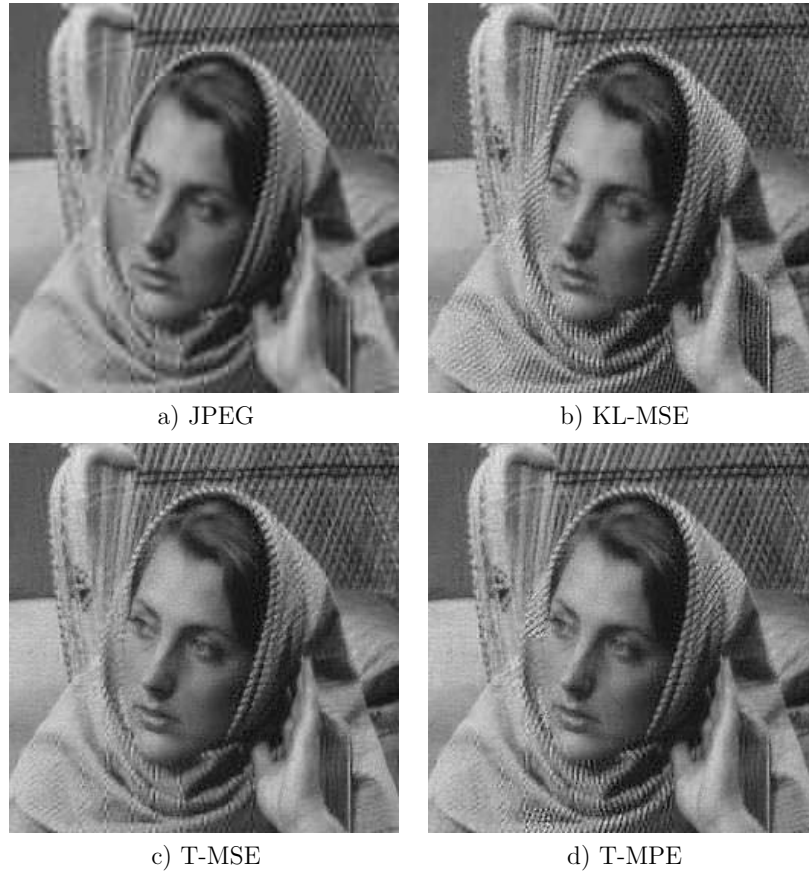


Figura 4.8: Barbara a 0.5 bits/pixel

Como puede verse, los bigotes están tan suavizados que no pueden distinguirse. Además, el efecto de los bloques es bastante notable (véanse las figuras 4.8 y 4.11). Por otro lado, las imágenes con KLT aparecen granuladas (obsérvese las zonas suaves de los pimientos o la mano de Bárbara). Esto podría deberse a la cuantización poco fina de algunos coeficientes.

Con respecto a la transformación aquí presentada, no hay diferencias significativas entre el uso del cuantizador MSE o el cuantizador MPE, aunque bien es cierto que el criterio MPE presenta una ligera mejora de la calidad subjetiva (véanse las ropas de Bárbara). En general, tanto T-MSE y T-MPE dan lugar a imágenes decodificadas con una calidad superior a aquella que se obtiene con JPEG o KL-MSE. Los detalles principales se preservan y los artefactos producidos por los bloques no se perciben tan fácilmente. Aparece una ligera granulosidad, pero no es tan apreciable como la que aparecía en el esquema KL-MSE. Estas apreciaciones subjetivas (basadas en la observación de las imágenes de-

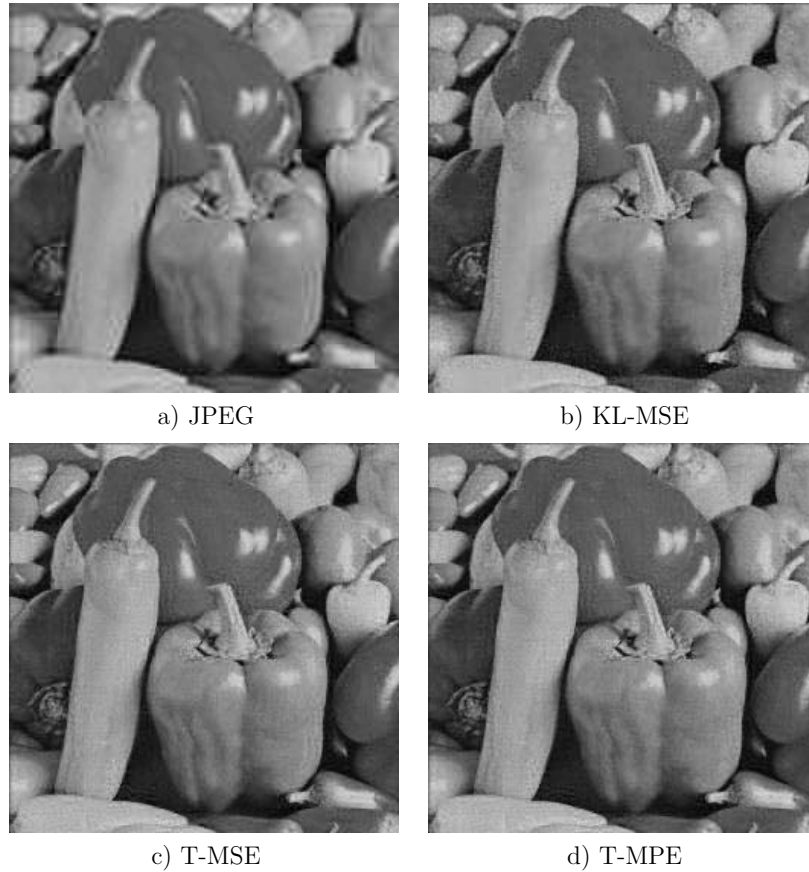


Figura 4.9: Pimientos a 0.5 bits/pixel

codificadas) son corroboradas por su lado por las medidas objetivas de la tabla 4.1. Los valores más altos (mejor calidad) para el PSNR se alcanzan con la aproximación T-MPE (excepto para Bárbara). Estos valores no son muy diferentes de los de la aproximación T-MSE. Las distancias perceptuales (D_2 and D_∞), nuevamente validan los comentarios previos. Para todas las imágenes y la medida PSNR, JPEG proporciona los resultados más pobres, mientras que si consideremos las distancias perceptuales, lo sustituye el KL-MSE. T-MPE es el mejor con las distancias perceptuales.

Tabla 4.1: Medidas de calidad para las distintas aproximaciones a 0.5 Bits/Pixel

Imagen	Medida	JPEG	KL-MSE	T-MSE	T-MPE
Barbara	PSNR	23.48	25.87	25.07	25.48
	D_2	29.85	64.17	13.91	13.78
	D_∞	6.42	24.18	3.87	3.76
Baboon	PSNR	23.46	25.34	26.93	27.16
	D_2	59.12	54.77	14.27	12.49
	D_∞	6.55	19.89	3.74	3.71
Peppers	PSNR	29.37	32.45	33.41	33.54
	D_2	22.2	35.07	9.22	8.04
	D_∞	5.43	13.42	3.3	2.47

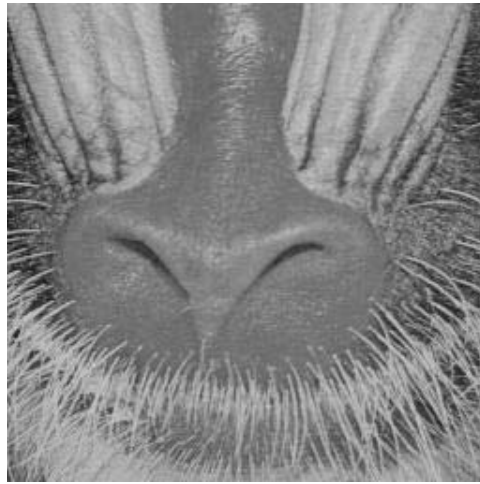


Figura 4.10: Hocico del mandril

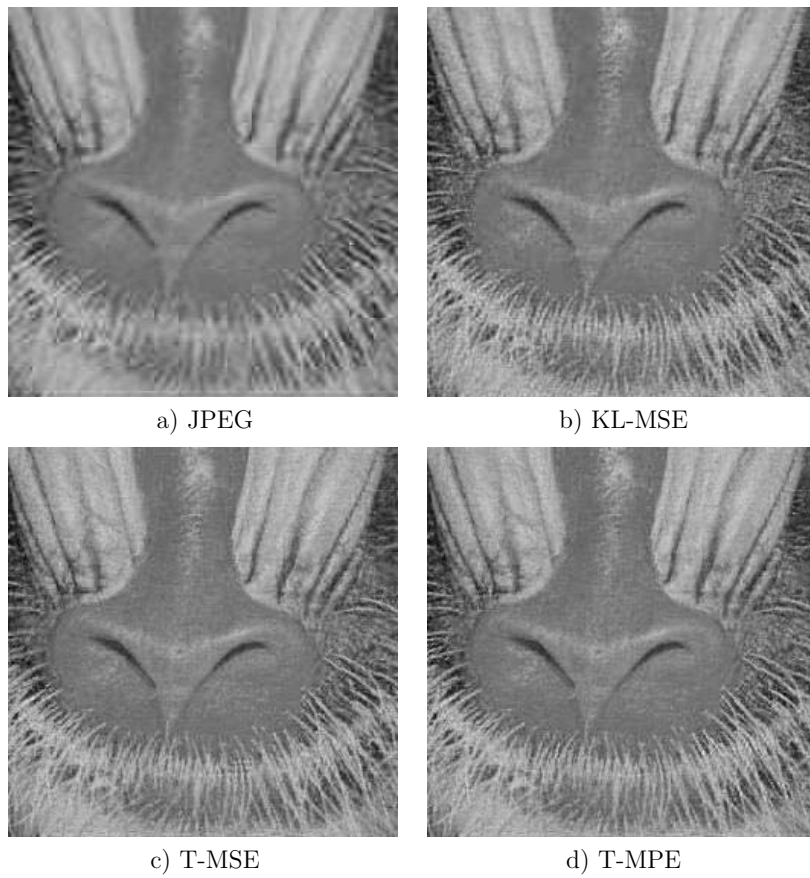


Figura 4.11: Hocico del mandril a 0.5 bits/pixel

Capítulo 5

Transformación no lineal para la decorrelación estadística y perceptual

¹ En el capítulo 4, se asumió una matriz perceptual igual para todos los puntos. Los resultados obtenidos fueron buenos, sin embargo, realmente las métricas perceptuales dependen del punto (imagen) que estemos tratando, puesto que el comportamiento incremental del sistema visual humano varía en función del estímulo de entrada. En este capítulo, se empleará una transformación no lineal (inspirada en un modelo perceptual) cuyo objetivo será la reducción (eliminación) de las correlaciones estadísticas y perceptuales, permitiendo así un procesado escalar de los coeficientes así como una mejor interpretación cualitativa de los vectores de características para su uso en diversas aplicaciones de análisis de imágenes. Se mostrará su mejor comportamiento frente a diversas alternativas en lo que respecta al campo de la compresión de imágenes. Para este último punto, se hace necesario conocer la inversa de la transformación. Para este fin, presentaremos un método para invertirla que ha resultado muy eficaz. Este método se basa en la resolución numérica de sistemas de ecuaciones diferenciales ordinarias. Aunque únicamente se tratará aquí la aplicación de la transformación en *transform coding*, podría emplearse en otras aplicaciones que trabajasen con imágenes naturales (*retrieval* [129], fusión [130], ...).

5.1. Relaciones estadísticas y perceptuales entre coeficientes

En el capítulo 4, se presentaron dos matrices, la matriz de covarianzas y la matriz métrica perceptual que cuantificaban en cierta manera las relaciones estadísticas y per-

¹Una parte importante del presente capítulo, constituye el artículo '*Non-linear Invertible Representation for Joint Statistical and Perceptual Feature Decorrelation*', publicado en *Lecture Notes in Computer Science*, 2000 [128].

ceptuales entre coeficientes. Se usó una única métrica perceptual que resumía en una sola matriz todas las posibles métricas. No obstante, la métrica depende en gran medida del punto (imagen), como mostraremos seguidamente. La matriz métrica (usando la matriz jacobiana media como ya hicimos anteriormente) y de covarianzas en el dominio a del capítulo 4, calculadas con las imágenes de 4.4.2, nos servirán como referente. Las denotaremos respectivamente \overline{W}_a y \overline{S}_a . Para otro conjunto de imágenes, se calculó la matriz de covarianzas para cada imagen (S_a) y la métrica perceptual (W_a) para cada bloque (puede obtenerse aplicando la ecuación 5.2) en el dominio a (el dominio de partida de la transformación lineal del capítulo 4). Se calcularon unas medidas de dispersión de estas matrices. Para poder comparar los resultados para las matrices métricas y las de covarianzas, hemos normalizado por la norma de la matriz referente respectiva, a modo de coeficiente de variación. Las medidas que usaremos para cuantificar la variación de estas matrices serán:

$$\text{Para una imagen } \frac{\|S_a - \overline{S}_a\|}{\|\overline{S}_a\|}$$

$$\text{Para un bloque } \frac{\|W_a - \overline{W}_a\|}{\|\overline{W}_a\|}$$

donde $\|\cdot\|$ denota la norma 2 de matrices [131]. En la tabla 5.1 aparecen las medias y medianas de las medidas anteriores.

Tabla 5.1: Medias y medianas de la dispersión de las matrices de covarianzas y métricas en el dominio a

	$\ S_a - \overline{S}_a\ /\ \overline{S}_a\ $	$\ W_a - \overline{W}_a\ /\ \overline{W}_a\ $
Media	0.2295	9.2597
Mediana	0.2189	1.9989

Según los resultados de la tabla 5.1, las matrices métricas varían mucho más que las de covarianzas, en el dominio a . Esto quiere decir que en realidad no existen unos únicos ejes propios perceptuales, sino que varían localmente. Por lo tanto, es necesaria una transformación no lineal para alcanzar una decorrelación perceptual. La transformación lineal presentada en el capítulo 4, podría verse como una aproximación del caso general no lineal.

En este capítulo, como ya hemos dicho, tendremos en consideración la mayor variabilidad de la métrica y emplearemos una transformación no lineal para llegar a una representación de la imagen de inspiración perceptual en la que se reducirán simultáneamente la correlación estadística y perceptual entre características, mejorando de esta manera la transformación propuesta en el capítulo 4. Tal y como se vio en el capítulo 4, el objetivo de la transformación sería encontrar los ejes propios de cada elipsoide (los definidos por las matrices de covarianzas y métrica) y diagonalizarlos simultáneamente. En el contexto de *transform coding*, la eficiencia decorreladora de una transformación de extracción de características suele referirse a la naturaleza diagonal de la matriz de covarianzas resultante. Nosotros empleamos también, como complemento a la matriz de covarianzas, la matriz

métrica y puesto que los elementos de esta matriz fuera de la diagonal indican interacciones entre distintas dimensiones del espacio de características, propondremos las mismas medidas usadas para la matriz de covarianzas a la hora de evaluar las transformaciones desde el punto de vista perceptual.

De esta manera, dada una matriz M que describe las relaciones (estadísticas o perceptuales) entre características, puede definirse una medida escalar η [132] (la interacción estadística, η_s o la interacción perceptual, η_p) que compare las magnitudes de los coeficientes fuera de la diagonal con las magnitudes de los coeficientes de la diagonal,

$$\eta = \frac{\sum_{i \neq j} |M_{ij}|}{\sum_i |M_{ii}|}. \quad (5.1)$$

Una vez definidas estas medidas, el objetivo de la transformación estaría en la minimización de η_s y η_p . En este capítulo, trabajaremos con el modelo perceptual de normalización de la energía que ya se presentó en el capítulo 4. Con lo cual, tras su aplicación a la representación de la imagen en el dominio DCT, llegaremos a un dominio con $\eta_p = 0$ para cada imagen de entrada, ya que obteníamos un dominio euclídeo. Más allá del efecto decorrelador perceptual que obviamente se obtiene con esta transformación, se mostrará que también tiene interés estadístico (quizá el sistema visual humano, adaptado al análisis de imágenes naturales, realice con esta transformación no lineal una especie de transformación independizadora [118]). Debido a su estructura particular, la transformación también reduce las correlaciones estadísticas entre coeficientes en el dominio DCT, disminuyendo enormemente η_s . Así, ambas medidas, η_s y η_p , se minimizarán con una única transformación adaptativa que podremos invertir adecuadamente, sin necesidad de almacenar funciones bases ad hoc. Para el cálculo de η_p es necesario conocer la matriz métrica en cada dominio. Tal y como se vio en el capítulo 4, la métrica en el dominio *respuesta* es euclídea (matriz identidad, I) y por las propiedades que también se vieron en ese capítulo, es fácil obtener la métrica en otros dominios: basta con multiplicar por la matriz jacobiana de la transformación correspondiente,

$$W_a = J^T \cdot I \cdot J, \quad (5.2)$$

donde J representa la matriz jacobiana de la transformación aplicada para ir del dominio a al dominio *respuesta*.

5.2. Modelo no lineal y su efecto decorrelador

La transformación que va a emplearse a lo largo del presente capítulo, es el modelo no lineal de normalización de la energía que se presentó en el capítulo 4:

$$r_i = \frac{\alpha_i}{100} |a_i| + \alpha_i \frac{|a_i|^2}{\beta_i + (h * |a|^2)_i} \quad (5.3)$$

donde i indicaba frecuencia espacial y a_i era el coeficiente en el dominio *transformado* (usando DCT local). El significado del resto de los parámetros ya se explicó en el capítulo 4.

Veamos el porqué de usar este modelo. En primer lugar, transforma el dominio DCT en un espacio *perceptualmente* euclídeo. En segundo lugar, su estructura lo convierte en una forma especial de codificador predictivo, con lo que la salida r debiera mostrar una relación estadística menor que la entrada DCT.

La idea básica de la codificación predictiva (DPCM) [57] es eliminar de cada coeficiente la parte que pueda predecirse a partir de los coeficientes vecinos. Si descontamos de la señal original una predicción de cada coeficiente, se reducirá de alguna manera la relación entre los coeficientes vecinos. En DPCM más común, ese descuento consiste en substraer la predicción de la señal de entrada [57].

En [118] se ha mostrado que la normalización de la ecuación 4.9 puede ajustarse para maximizar la decorrelación de la salida (en este trabajo, no se ha perseguido este objetivo y los valores de los parámetros responden a valores empíricos que se ajustan a los experimentos, como ya se explicó en el capítulo 4). Esta normalización mediante una suma pesada de los coeficientes vecinos podría interpretarse como un DPCM divisivo (véase la figura 5.1): si el punto central del kernel gaussiano adopta el valor cero (es decir, si el coeficiente a_i no se tiene en cuenta en $(h * |a|^2)_i$ como sucede en [118]), la convolución en el denominador podría verse como una predicción de la energía de cada coeficiente a partir de las energías de los vecinos. La división sería una forma distinta de descontar la predicción del coeficiente de entrada.

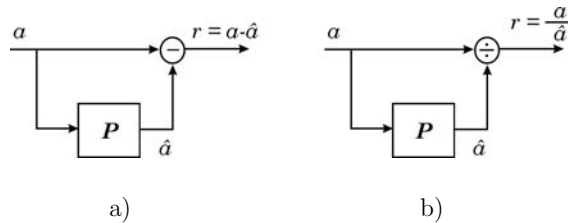


Figura 5.1: Distintos esquemas DPCM. La figura 5.1a) muestra el clásico DPCM. La figura 5.1b) muestra la interpretación en forma de DPCM del modelo de normalización. En este caso, la parte que se predice de los coeficientes vecinos se descuenta dividiendo la señal de entrada por la predicción

De hecho, la etapa predictor de este modelo no lineal es similar al esquema predictor que se ha utilizado con éxito en [133] para explotar las probabilidades condicionadas de los coeficientes transformados y codificarlos de manera más eficiente. Todo ello sugiere que el esquema de interacción del modelo de normalización no lineal, podría ciertamente reducir la correlación estadística entre los coeficientes de r . Notemos nuevamente que los parámetros de la ecuación 4.9 que usaremos en el trabajo, son empíricos y no han sido específicamente optimizados para mejorar la decorrelación de un conjunto de entrenamiento

dado.

5.3. Método diferencial para la inversión del modelo de normalización no lineal

Para su aplicación en codificación de imágenes, se hace necesario poder invertir la transformación (tras haber cuantizado los coeficientes en el dominio respuesta). El kernel de interacción (y predicción) del modelo de normalización no lineal, hace que la transformación no sea inmediatamente invertible. Así, tendremos un sistema de ecuaciones no lineales:

$$R(a_b) = r_b, \quad (5.4)$$

donde r_b es conocido y a_b es el vector solución buscado. Puesto que el jacobiano de la transformación es distinto de cero, por el teorema de la función inversa, podemos asegurar que existe la inversa en un cierto abierto.

A pesar de que la versión actual del modelo perceptual (ecuación 5.3), es suficientemente sencilla como para obtener una inversa analítica tras una apropiada serie de manipulaciones ², en este apartado propondremos un método más general para la inversión de este tipo de problemas. El interés del método reside en que, aun no siendo estrictamente necesario en el caso del modelo actual, podrá ser aplicado en otras ocasiones.

El método que presentaremos será alternativo a otra serie de métodos numéricos, cuyo objetivo es también la obtención de la inversa planteada como la búsqueda de la raíz de un sistema no lineal ($R(a_b) - r_b = 0$). Existen diversos métodos para este fin, como el método de Newton-Raphson [86]. Notemos que la tarea de encontrar una raíz multidimensional es, con frecuencia, más difícil que la de minimizar funciones de varias variables y resaltemos que no son problemas equivalentes (en las págs. 382-383 de [86] se explica con precisión esta distinción). Si, por ejemplo, tratásemos de minimizar $\|R(a_b) - r_b\|$, por medio de métodos como el del descenso por gradiente [86], nos podríamos ver atrapados en mínimos locales. Esto es precisamente lo que ha sucedido en las pruebas realizadas, partiendo de 3 puntos iniciales diferentes, como luego comprobaremos. Debido a esta circunstancia, y a la necesidad de que en el método de Newton-Raphson hemos de partir de un punto inicial suficientemente bueno para conseguir converger a la raíz, hemos optado por el método que seguidamente presentaremos y con el que hemos alcanzado buenas soluciones, desde diversos puntos iniciales.

La base del método propuesto, es expresar el problema de la inversión como un problema de valores iniciales y reducirlo a la integración de unas ecuaciones diferenciales. En un problema de valores iniciales, conocida una solución a_a en unas ciertas condiciones iniciales, r_a , y el comportamiento local del sistema (descrito mediante las ecuaciones diferenciales),

²J. Malo y E. Simoncelli, 'Fast Inversion of the Divisive Normalization', *Tuesday Seminar of the Lab. of Computational Vision, NYU*, julio 2001

se busca la solución, a_b , en unas condiciones distintas, r_b , integrando las ecuaciones diferenciales desde r_a hasta r_b . Para que exista la solución, es necesario que el comportamiento del sistema sea suave (que la matriz jacobiana está acotada en el camino de r_a a r_b). Pasemos a presentarlo explícitamente.

Por el teorema del valor medio sabemos que

$$R^{-1}(r_b) - R^{-1}(r_a) = \bar{J}(\xi)(r_b - r_a) \quad (5.5)$$

siendo $\bar{J}(\xi)$ la matriz jacobiana de R^{-1} con respecto a r , y la barra indica que cada fila de la matriz jacobiana se evalúa en un valor medio distinto. Cada uno de los valores medios, es un punto interno del segmento que une r_b y r_a . Por supuesto, aunque desconocemos R^{-1} , sí que es posible calcular su matriz jacobiana, puesto que es la inversa de la matriz jacobiana de R respecto a a . Si llamamos $dr = r_b - r_a$, la ecuación 5.5 podemos expresarla como:

$$a_b = R^{-1}(r_b) = R^{-1}(r_a) + \bar{J}(\xi)dr = a_a + \bar{J}(\xi)dr. \quad (5.6)$$

Esta ecuación representaría la evolución local de la respuesta inversa.

Consideremos primero el caso unidimensional. Supongamos que tenemos el problema de valores iniciales

$$\begin{cases} y' = f(x, y) \\ y(x_0) = y_0 \end{cases} \quad (5.7)$$

y queremos conocer $y(z)$ (en nuestro caso y' sería la derivada de la inversa de R). Integrando podríamos encontrar el valor de y en z . Existe una abundante literatura sobre métodos para integrar ecuaciones diferenciales, uno de los más populares [134] es el método de Runge-Kutta de cuarto orden. Se construye a partir de una secuencia de 4 incrementos de Euler que requieren la evaluación de 4 veces el lado derecho de la ecuación 5.7, imitando la estructura de la fórmula de Simpson de integración numérica. En concreto, se generan aproximaciones y_n a $y(x_0 + nh)$ para un paso h fijo y para $n = 0, 1, 2, \dots$, usando la fórmula de recurrencia:

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (5.8)$$

donde

$$\begin{aligned} k_1 &= f(x_n, y_n)h \\ k_2 &= f\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right)h \\ k_3 &= f\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right)h \\ k_4 &= f(x_n + h, y_n + k_3)h \end{aligned}$$

Este método tiene la ventaja de autoiniciarse, ser estable y dar una buena precisión. Este método puede adaptarse fácilmente si tenemos sistemas de ecuaciones diferenciales con valores iniciales (capítulo 5 de [135]), y es esto lo que hemos empleado en nuestro

problema. Partiendo de unas condiciones iniciales conocidas (r_a, a_a) se ha propagado la solución desde r_a hasta el punto deseado del que queríamos conocer su inversa, r_b , empleando Runge-Kutta de cuarto orden haciendo uso de que conocemos la matriz jacobiana de la inversa. De esta manera, obtenemos una aproximación a la inversa de r_b, a_b . Abusando de la notación en algunos puntos, expresémoslo mediante las ecuaciones correspondientes.

$$\begin{cases} y' = JR^{-1}(r_b) = (JR(a_b))^{-1} = F(y) \\ y(r_a) = a_a \end{cases} \quad (5.9)$$

donde JR denota la matriz jacobiana de R con respecto a a y análogamente JR^{-1} .

Partiendo de nuestro valor inicial (r_a, a_a) , y con vector de paso h , la recurrencia sería como la ecuación 5.8, siendo ahora

$$\begin{aligned} k_1 &= F(y_n)h \\ k_2 &= F\left(y_n + \frac{k_1}{2}\right)h \\ k_3 &= F\left(y_n + \frac{k_2}{2}\right)h \\ k_4 &= F(y_n + k_3)h \end{aligned}$$

Para comprobar el funcionamiento de esta forma de calcular la inversa de R , así como su velocidad y robustez, presentaremos algunos de los experimentos realizados. Se han transformado diversas imágenes naturales bien conocidas (Barbara, Einstein, Barcos (*Boats*), Pimientos (*Peppers*), Lena, Casas (*Golden Hill*) según el modelo de normalización no lineal y después se han invertido usando el algoritmo de Runge-Kutta de cuarto orden, partiendo de distintos puntos iniciales y variando también el tamaño del paso (y como consecuencia el número de aproximaciones, n).

Las figuras 5.2 y 5.3 muestran los errores de reconstrucción medios para las distintas imágenes, según el número de aproximaciones n y distintos puntos iniciales. En la figura 5.2 mostramos los errores de reconstrucción medios usando hasta 25 aproximaciones. La línea de puntos es la media del error obtenido si partimos de un espectro DCT de la forma $1/f$, la curva en línea discontinua representa el error al partir de la DCT media sobre el conjunto de imágenes consideradas y por último la curva en línea continua indica el error medio si partimos de un espectro de la DCT plano. Las barras verticales son indicativas de la dispersión. Pensemos que bajo el valor -4 , los errores son visualmente despreciables.

En la figura 5.3, se muestra en la parte superior de la gráfica los errores cometidos empleando la minimización por el método del descenso por gradiente, mientras que en la parte inferior aparecen los errores de nuestro método. Fijémonos que a pesar de utilizar un elevado número de iteraciones (hasta 100), en el método del descenso por gradiente, nos quedamos atrapados en mínimos locales, alejados de la solución buscada. Ahora la línea continua indica que partimos de un espectro medio, la línea discontinua de un espectro $1/f$ y la línea discontinua con rayas y puntos de un espectro plano.

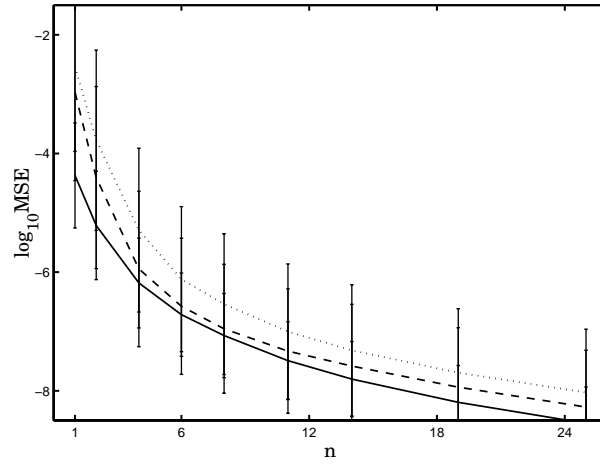


Figura 5.2: Errores de reconstrucción medios, partiendo de tres condiciones iniciales distintas, para el método propuesto

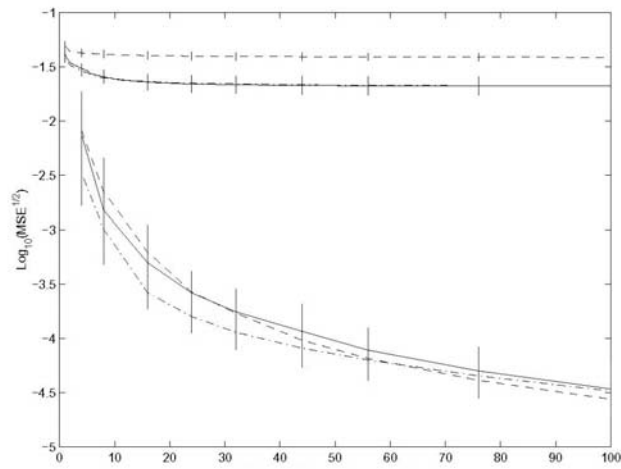


Figura 5.3: Errores de reconstrucción medios, partiendo de tres condiciones iniciales distintas, para el método propuesto (parte inferior) y el método de descenso por gradiente (parte superior)

Además de con estas gráficas, podemos apreciar la convergencia del método propuesto, observando directamente la inversión conseguida de las imágenes, variando el número de aproximaciones (n): 1, 2, 4, 6, 8, 14, 19 y 25. En las figuras 5.4, 5.5, 5.6, 5.7, se muestra la inversión con algunas imágenes bien conocidas, partiendo de la respuesta a una DCT media.



Figura 5.4: Inversión de Bárbara con distinto número de aproximaciones

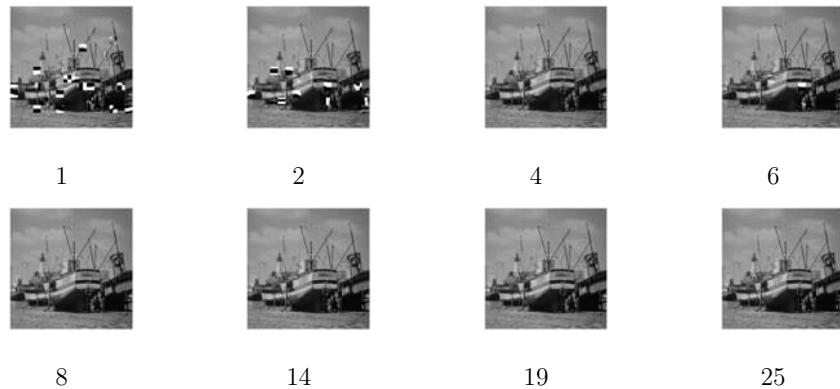


Figura 5.5: Inversión de Barcos con distinto número de aproximaciones

En base a los experimentos anteriores, podemos concluir que el método propuesto alcanza la inversa (con una distorsión insignificante) para todos los bloques, sin importar las condiciones iniciales y en un número pequeño de aproximaciones. Para la mayoría de los bloques, se consigue una inversa aceptable en muy pocas aproximaciones (sobre 4), sin importar las condiciones iniciales. Unos pocos bloques, en cambio, requieren un número mayor, pero en ningún caso se ha superado el 25. Así pues, el método no parece sensible

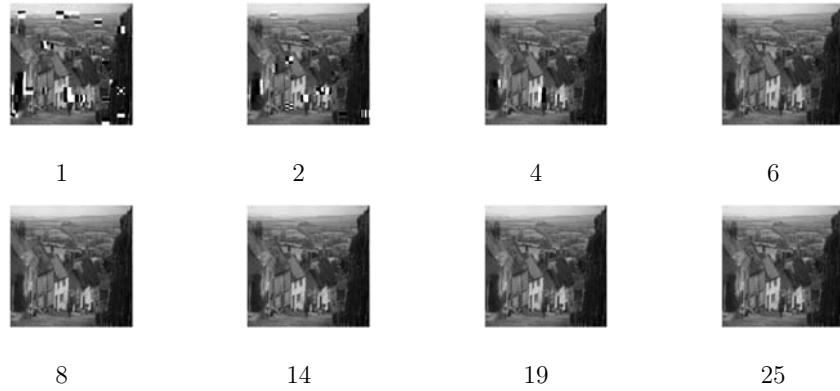


Figura 5.6: Inversión de Casas con distinto número de aproximaciones

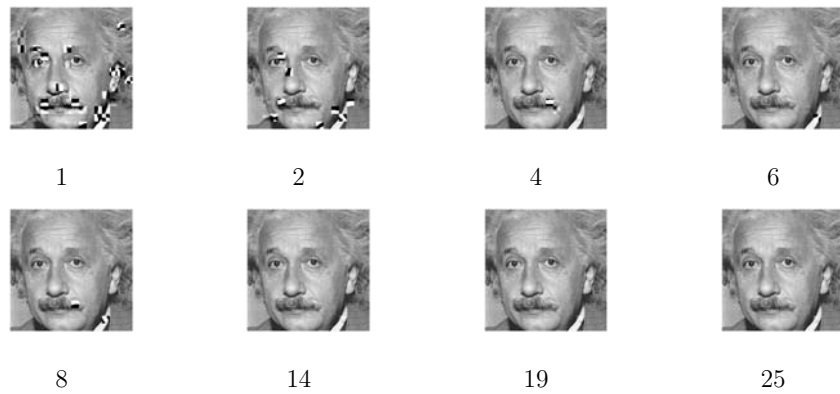


Figura 5.7: Inversión de Einstein con distinto número de aproximaciones

a los valores iniciales, la clave sobre la distorsión final se encuentra en la naturaleza del bloque.

5.4. Comparación entre distintos dominios de representación

En este apartado, mostraremos las dos distintas medidas de interacción que se presentaron en 5.1, en los dominios a (transformado) y r (respuesta) y así veremos el poder decorrelador de la transformación no lineal. Las matrices para las que calcularemos η_s y η_p , son las previamente consideradas, \overline{W}_a y \overline{S}_a y por otro lado la matriz de covarianzas

estimada en el dominio r para las mismas imágenes, \bar{S}_r y $W_r = I$ (identidad). En la tabla 5.2 se presentan dichas medidas.

Tabla 5.2: Medidas de interacción para dos dominios de representación

	Transformado (a)	Respuesta (r)
η_s	3.9	0.5
η_p	1.5	0.0

En la figura 5.8, aparecen las matrices anteriores, que muestran claramente las ventajas de trabajar en el dominio respuesta, r , pues la métrica es euclídea y la matriz de covarianzas se acerca a la diagonalidad.

5.5. Imágenes decodificadas

En esta sección, compararemos la transformación no lineal frente a otras alternativas. Usaremos, igual que hicimos en el capítulo 4, dos diseños para el cuantizador: uno basado en el MSE y el otro en el MPE. Denotaremos como R-MSE y R-MPE las aproximaciones que usan la transformación no lineal R propuesta y cada uno de los dos cuantizadores. El diseño de los cuantizadores se ha realizado tal y como se expuso en 4.4.2. Se han utilizado 25 aproximaciones para la inversión. Compararemos nuestra aproximación con dos cuantizadores en dominio DCT: el estándar JPEG y con la aproximación que utiliza un cuantizador MPE con un modelo de respuesta (r) no lineal sin interacciones [71] (No lineal MPE-DCT). JPEG podría verse como una aproximación con un cuantizador MPE para un modelo de respuesta (r) lineal [71].

En la figura 5.9, se muestra un ejemplo de imagen decodificada usando estas aproximaciones, Barbara con una tasa de compresión para todos los casos de 0.27 bits/píxel.

Puede apreciarse como JPEG elimina detalles (de altas frecuencias) importantes, dando lugar a una reconstrucción suave. La aproximación con el cuantizador MPE no lineal sin interacciones en DCT, conserva estos detalles, pero en cambio incrementa notablemente el efecto de los bloques. Para R-MSE se vuelve a reducir el efecto bloque, pero no es muy superior a JPEG. El mejor resultado se obtiene con R-MPE, pues se preservan los detalles y domina bastante bien el efecto bloque. Nuevamente queda demostrada la superioridad de la aproximación basada en MPE frente a la PMSE (recordemos que en nuestro caso concreto como W_r es la identidad, sería directamente MSE) [71]. Pero lo importante es que la transformación propuesta con el cuantizador adecuado, realmente mejora las diversas alternativas y por sí solo se obtiene una imagen de 'buena' calidad para la reducida tasa de compresión considerada (0.27 bits/píxel).

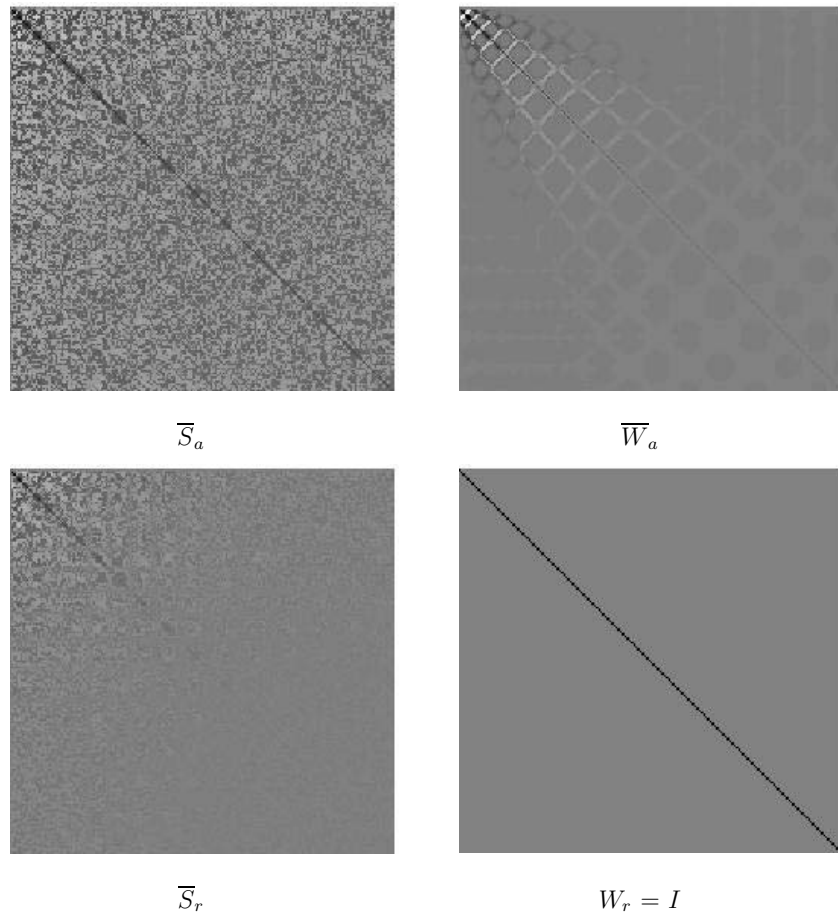


Figura 5.8: Matrices de covarianzas y métricas en distintos dominios. Sólo se muestra el cuadrante superior izquierdo, una submatriz 176×176 , por claridad. Como ya explicamos en el capítulo 4, se muestran en escaneo zig-zag



a) JPEG



b) No lineal MPE-DCT



c) R-MSE



d) R-MPE

Figura 5.9: Barbara a 0.27 bits/píxel

Capítulo 6

Conclusiones y trabajo futuro

En primer lugar, en los capítulos 2 y 3 se ha tratado el problema de la clasificación de texturas. En la resolución de este tipo de problemas los pasos básicos que se contemplan son: 1) extracción de vectores de características con potencia discriminadora entre las clases; 2) procedimiento de clasificación basado en las características anteriores.

El capítulo 2 se centra en el primer paso y constituye principalmente el trabajo [72]. Hemos asociado distintos conjuntos aleatorios multivariantes a una textura dada (considerada como una función aleatoria) y a partir de ellos, podemos usar distintas características para clasificar dicha textura (aleatoria). Las características de texturas que se han propuesto son los descriptores distribucionales usuales de este tipo de modelos, tomados de la Geometría Estocástica: funciones de distribución de contacto y funciones \mathcal{K} . Notemos que se ha adoptado una aproximación no paramétrica, es decir, no se ha asumido ningún modelo paramétrico para obtener los descriptores probabilísticos de las texturas analizadas. Los distintos experimentos que se han llevado a cabo indican que podemos obtener porcentajes de clasificación correcta realmente elevados (100 % en algunos experimentos). Las funciones de distribución de contacto muestran un mejor desempeño que las funciones \mathcal{K} . Además, se han alcanzado resultados comparables a otras características de texturas ampliamente usadas como las extraídas de campos aleatorios de Markov, matrices de co-ocurrencia, dimensión fractal y filtros de Gabor. En presencia de ruido también se consigue una buena precisión, lo cual sugiere que estas características son bastante robustas en un ambiente ruidoso.

Existen algunos puntos adicionales que podrían estudiarse. Primero, el número de puntos en los que las funciones H y \mathcal{K} deberían estimarse. Segundo, el número de orientaciones de las funciones de distribución de contacto lineales permanece como una cuestión abierta. Tercero, aquí hemos usado las distribuciones de contacto circulares y lineales. Esta es una opción natural cuando no disponemos de ninguna información previa sobre nuestros datos, pero podría ser interesante diseñar métodos para seleccionar distintas distribuciones de contacto asociadas con otros elementos estructurantes convexos y compactos. Cuarto, diseñar procedimientos de segmentación (construcción de los conjuntos aleatorios) adaptados a los problemas específicos para resaltar las propiedades de las texturas. Aunque, sólo

se ha estudiado para texturas binarias y a niveles de gris, la metodología podría extenderse con pequeñas modificaciones a imágenes a color o multiespectrales. Por último, en lugar de usar conjuntos aleatorios, se podrían utilizar otros elementos de la Geometría Estocástica (procesos de puntos o líneas) para describir las texturas y clasificarlas o segmentarlas.

En el capítulo 3, en cambio nos hemos centrado en el segundo paso. Se presenta un procedimiento de clasificación de texturas basado en tests de aleatorización y en ideas de la teoría de conjuntos difusos, dando lugar a etiquetas posibilísticas que indican la pertenencia de la textura a cada clase, pudiendo ir de no ser cubierta por ninguna clase a tener una mezcla de las clases, ya que estas etiquetas no tienen porqué sumar uno, como las etiquetas probabilísticas. Con este procedimiento podemos plantear distintas medidas para comparar (o seleccionar) características de texturas, más allá del clásico porcentaje de clasificaciones correctas. Se han dado algunos ejemplos usando características de texturas ampliamente usadas. Aunque esta metodología la hemos presentado y utilizado con vectores de características, puede extenderse a cualquier tipo de características (funciones, variables discretas, características difusas, ...), siempre que podamos definir una distancia (o similaridad) entre las características. Un punto interesante sería un estudio comparativo de distintas características en segmentación de texturas. Podrían representarse gráficamente y en lugar de asignar un color determinado (*crisp*) a la clase dominante, se podría asignar un color 'difuso' a cada píxel mezclando los colores de las clases en proporción a su pertenencia.

En segundo lugar, en los capítulos 4 y 5 hemos tratado el problema de seleccionar una representación adecuada para su aplicación en compresión de imágenes estáticas (en el contexto de *transform coding*). Para ello, hemos tenido en cuenta factores estadísticos y perceptuales, puesto que las imágenes naturales cuentan con ambos tipos de redundancia: estadística y psicovisual.

En el capítulo 4 se introduce una transformación lineal para eliminar las relaciones de segundo orden estadísticas y perceptuales. La correlación perceptual entre coeficientes se ha formalizado a través de la matriz métrica perceptual, de la misma forma como se representa la correlación estadística en la matriz de covarianzas. Debido a su acción decorrelante, la transformación propuesta facilita y justifica la aplicación de una cuantización escalar y el error total del sistema puede ser determinado con precisión a través de la suma de las distorsiones medias de cada coeficiente. A la vista de las figuras y los resultados de distorsión, una transformación que decorrelacione únicamente estadísticamente (KLT) no es suficiente por sí sola. Se han ilustrado los beneficios de considerar además el funcionamiento del sistema visual humano en la definición de la representación de la imagen. La decorrelación estadística y perceptual conjunta da lugar a mejores resultados en imágenes naturales. Aquí, sólo hemos tratado la aproximación de segundo orden desde el punto de vista estadístico y perceptual. La idea podría extenderse a interacciones de más alto orden, estadísticas (momentos de mayor orden) y perceptuales (términos de mayor orden en la expansión de la respuesta no lineal del sistema visual humano). Por supuesto, a pesar de que obtenemos buenos resultados con esta transformación, pueden mejorarse utilizando la transformación no lineal presentada en el capítulo 5.

Como ya dijimos, la suposición de una única matriz métrica no es cierta, ya que el sistema visual humano otorga distinta importancia a diferentes texturas, escalas o colores. En el capítulo 5 (buena parte del cual aparece en [128]), sí que se contempla la variación de la métrica según la imagen. Utilizamos una transformación no lineal, completamente inspirada en factores psicovisuales (es una transformación que trata de explicar el comportamiento del sistema visual humano en su primera etapa, concretamente en el área V1 del córtex). Debido a la naturaleza no lineal de esta transformación, hemos presentado un método de integración numérica de ecuaciones diferenciales para invertirla. A través de varios experimentos hemos comprobado su convergencia desde distintos valores iniciales. En cuanto a los resultados de compresión, los experimentos realizados sugieren la conveniencia del empleo de esta transformación. En este trabajo sólo se ha contemplado su aplicación en *transform coding* con imágenes estáticas, sin embargo podría resultar de utilidad en otros contextos en los que el ser humano sea el usuario final, como, por ejemplo, compresión de vídeo (en el que ya se usó un modelo de respuesta más sencillo [117]).

Apéndice A

Texturas usadas en los capítulos 2 y 3

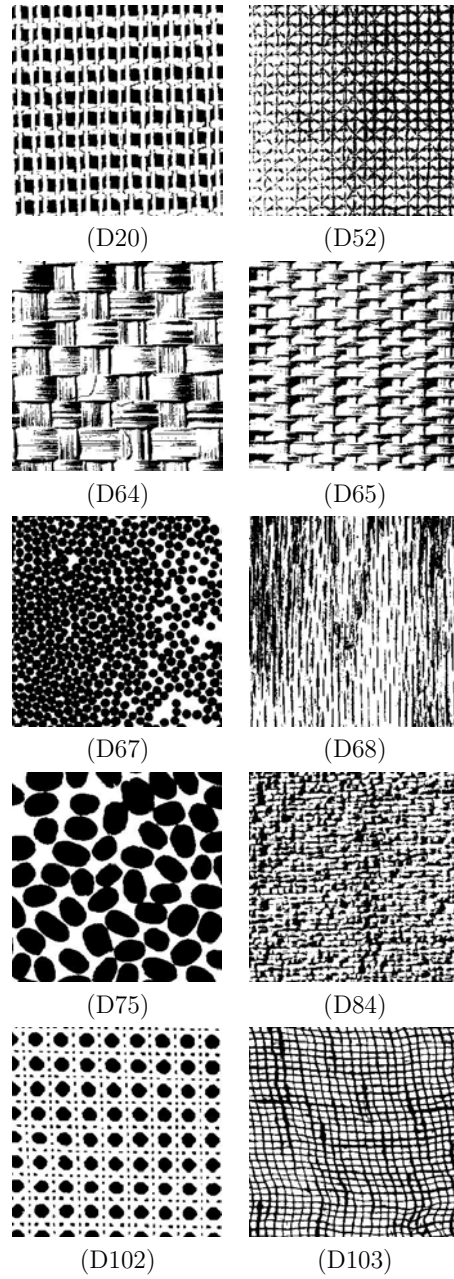


Figura A.1: Texturas binarias usadas en el primer experimento del capítulo 2 (segunda base)

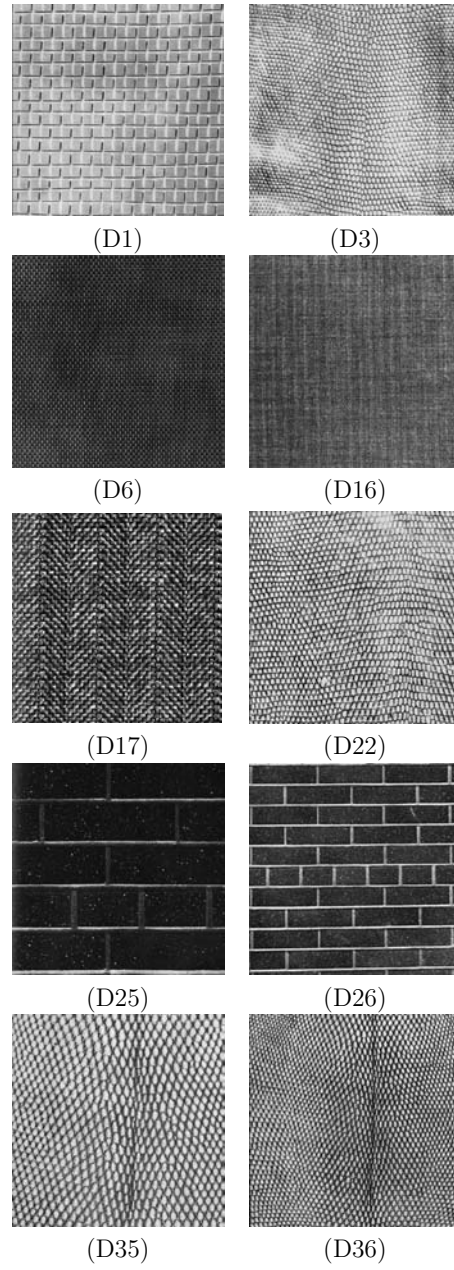


Figura A.2: Texturas de Brodatz usadas en los experimentos segundo y tercero del capítulo 2

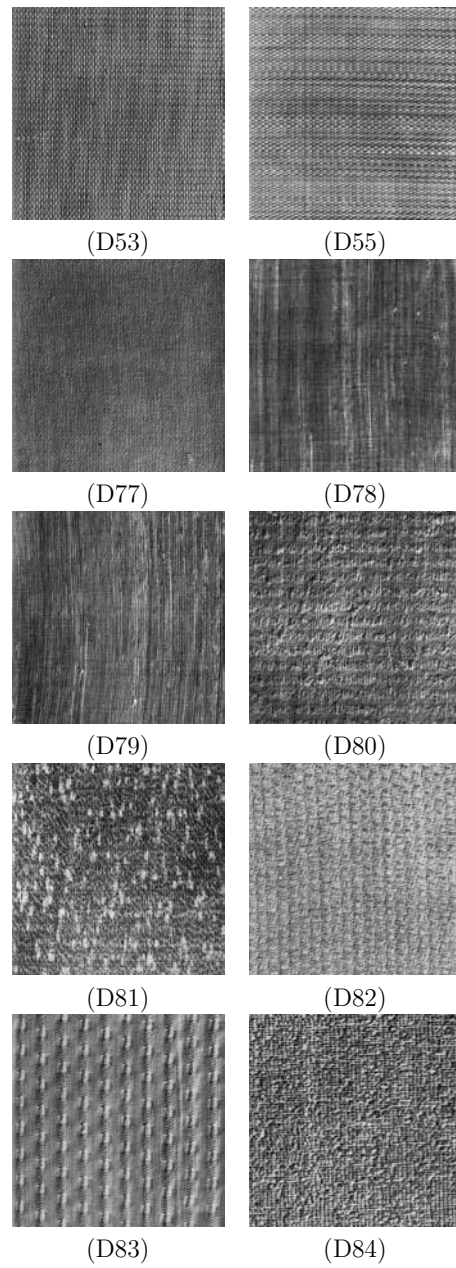


Figura A.3: Texturas de Brodatz usadas en los experimentos segundo y tercero del capítulo 2 (continuación)

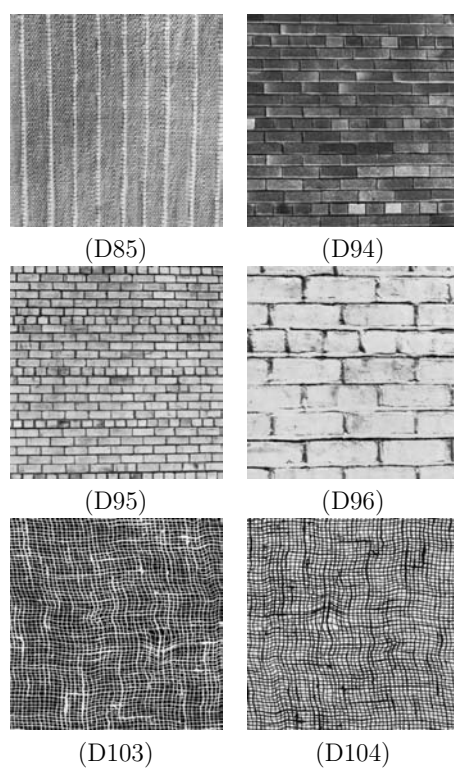
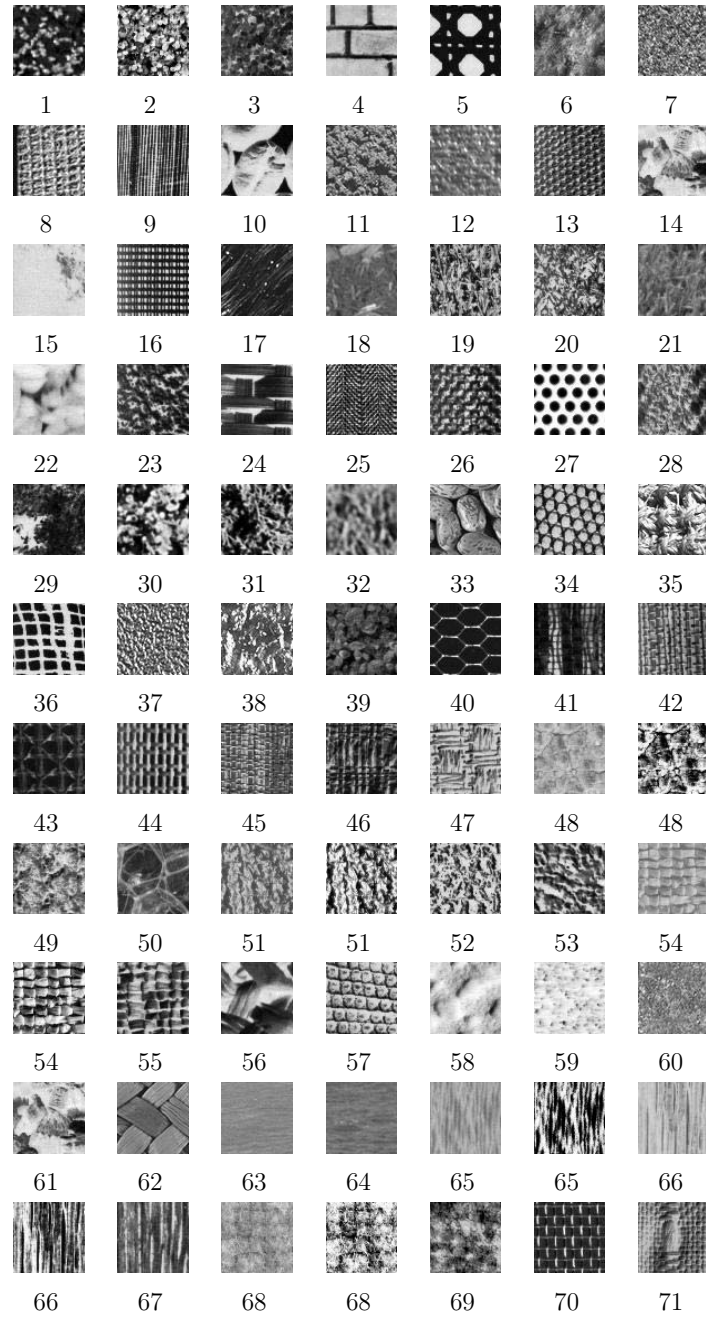


Figura A.4: Texturas de Brodatz usadas en los experimentos segundo y tercero del capítulo 2 (continuación)

Figura A.5: Base de texturas (64×64) de Kaplan [90]

Bibliografía

- [1] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, Boston, 1990.
- [2] A. Hyvärinen, J. Karhunen, y E. Oja, *Independent component analysis*, Wiley, New York, 2001.
- [3] J. Portilla y E.P.Simoncelli, “Texture modeling and synthesis using joint statistics of complex wavelet coefficients,” en *Proceedings of the IEEE Workshop on Statistical and Computational Theories of Vision, Forth Collins, CO*, 1999.
- [4] J. M. Coggins, *A Framework for Texture Analysis Based on Spatial Filtering*, Ph.D. thesis, Michigan State University, 1982.
- [5] M. Sonka, V. Hlavac, y R. Boyle, *Image Processing, Analysis and Machine Vision*, Chapman and Hall, 1993.
- [6] R.M. Haralick, “Statistical and structural approaches to texture,” *Proceedings IEEE*, vol. 67, núm. 5, págs. 786–804, 1979.
- [7] K. Karu, A.K. Jain, y R.M. Bolle, “Is there any texture in the image?,” *Pattern Recognition*, vol. 29, núm. 9, págs. 1437–1446, 1996.
- [8] B. Julesz, “Textons, the elements of texture perception, and their interactions,” *Nature*, vol. 290, págs. 91–97, 1981.
- [9] H. Voorhees y T.A. Poggio, “Detecting textons and texture boundaries in natural images,” en *1st Intl. Conf. on Computer Vision, London, England*, 1987.
- [10] S. Ando, “Texton finders based on Gaussian curvature of correlation with an application to rapid texture classification,” en *Proc. of the 1988 IEEE Intl. Conf. on Systems, Man and Cybernetics, China*, 1988.
- [11] M. Tuceryan y A.K. Jain, *Texture analysis*, capítulo 2.1, págs. 235–276, Handbook of Pattern Recognition and Computer Vision. World Scientific Publishing Company, 1993.
- [12] M. Tuceryan y A.K. Jain, “Texture segmentation using Voronoi polygons,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, págs. 211–216, 1990.

- [13] R. Chellappa y S. Chatterjee, "Classification of textures using Gaussian Markov random fields," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, págs. 959–963, 1985.
- [14] A.P. Pentland, "Fractal-based description of natural scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, págs. 661–672, 1984.
- [15] B.B. Chaudhuri, N. Sarkar, y P. Kundu, "Improved fractal geometry based texture segmentation technique," en *IEE Proceedings. IEE*, 1993, vol. 140, págs. 233–241.
- [16] J.J. Gibson, *The Perception of the Visual World*, Houghton Mifflin, Boston, MA, 1950.
- [17] K.I. Laws, *Textured Image Segmentation*, Ph.D. thesis, University of Southern California, 1980.
- [18] M. Unser y M. Eden, "Nonlinear operators for improving texture segmentation based on features extracted by spatial filtering," *IEEE Trans. Syst. Man Cybern.*, vol. 20, págs. 804–815, 1990.
- [19] J.M. Coggins y A.K. Jain, "A spatial filtering approach to texture analysis," *Pattern Recognition Letters*, vol. 3, págs. 195–203, 1985.
- [20] D. Gabor, "Theory of communication," *J. Inst. Elect. Eng.*, vol. 93, págs. 429–457, 1946.
- [21] J.G. Daugman, "Uncertainty relation for resolution in space, spatial frequency and orientation optimized by two-dimensional visual cortical filters," *Journal of the Optical Society of America*, vol. 2, núm. 7, págs. 1160–1169, 1985.
- [22] M.R. Turner, "Texture discrimination by Gabor functions," *Biol. Cybern.*, vol. 55, págs. 71–82, 1986.
- [23] M. Clark y A.C. Bovik, "Texture segmentation using Gabor modulation /demodulation," *Pattern Recognition Letters*, vol. 6, págs. 261–267, 1987.
- [24] S.G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, núm. 7, págs. 674–693, 1989.
- [25] A.K. Jain y F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, vol. 24, págs. 1167–1186, 1991.
- [26] T. Chang y C.C.J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Transactions on Image Processing*, vol. 2, págs. 429–441, 1993.
- [27] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Transactions on Image Processing*, vol. 4, págs. 1549–1560, 1995.

-
- [28] J. Portilla y E.P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelets coefficients," *International Journal of Computer Vision*, vol. 40, núm. 1, págs. 49–71, 2000.
- [29] P. Dewaele, P. Van Gool, y A. Oosterlinck, "Texture inspection with self-adaptative convolution filters," en *Proc. 9th Int. Conf. on Pattern Recognition, Rome, Italy*, 1988, págs. 56–60.
- [30] D. Chetverikov, "Detecting defects in texture," en *Proc. 9th Int. Conf. on Pattern Recognition, Rome, Italy*, 1988, págs. 61–63.
- [31] J. Chen y A.K. Jain, "A structural approach to identify defects in textured images," en *Proc. IEEE Int. Conf. on Systems, Man and Cybernetics, Beijing*, 1988, págs. 29–32.
- [32] R.W. Connors, C.W. McMillin, K. Lin, y R.E. Vasquez-Espinosa, "Identifying and locating surface defects in wood: Part of an automated lumber processing system," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 5, págs. 573–583, 1983.
- [33] L.H. Siew, R.M. Hodgson, y E.J. Wood, "Texture measures for carpet wear assessment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, págs. 92–105, 1988.
- [34] A.K. Jain, F. Farrokhnia, y D.H. Alman, "Texture analysis of automotive finishes," en *Proc. of SME Machine Vision Applications Conf., Detroit, MI*, 1990, págs. 1–16.
- [35] E.J. Wood, "Applying Fourier and associated transforms to pattern characterization in textiles," *Textile Research Journal*, vol. 60, núm. 4, págs. 212–220, 1990.
- [36] R. Sutton y E.L. Hall, "Texture measures for automatic classification of pulmonary disease," *IEEE Trans. Comput.*, vol. 21, págs. 667–676, 1972.
- [37] H. Harms, U. Gunzer, y H.M. Aus, "Combined local color and texture analysis of stained cells," *Comput. Vision Graph. Image Process.*, vol. 33, págs. 364–376, 1986.
- [38] G.H. Landeweerd y E.S. Gelsema, "The use of nuclear texture parameters in the automatic analysis of leukocytes," *Pattern Recognition*, vol. 10, págs. 57–61, 1978.
- [39] M.F. Insana, R.F. Wagner, B.S. Garra, D.G. Brown, y T.H. Shawker, "Analysis of ultrasound image texture via generalized rician statistics," *Opt. Engin.*, vol. 25, págs. 743–748, 1986.
- [40] C.C. Chen, J.S. Daponte, y M.D. Fox, "Fractal feature analysis and classification in medical imaging," *IEEE Trans. Medical Imaging*, vol. 8, págs. 133–142, 1989.
- [41] A. Lundervold, "Ultrasonic tissue characterization- a pattern recognition approach," Tech. Rep., Norwegian Computing Center, 1992.
- [42] P. Miller y S. Astley, "Classification of breast tissue by texture analysis," *Image and Vision Computing*, vol. 10, núm. 5, págs. 277–282, 1992.

- [43] D.L. Toulson y J.F. Boyce, "Segmentation of MR images using neural nets," *Image and Vision Computing*, vol. 10, núm. 5, págs. 324–328, 1992.
- [44] D. Wang y S.N. Srihari, "Classification of newspaper image blocks using texture analysis," *Comput. Vision Graph. Image Process*, vol. 47, págs. 327–352, 1989.
- [45] F.M. Wahl, K.Y. Wong, y R.G. Casey, "Block segmentation and text extraction in mixed text / image documents," *Comput. Vision Graph. Image Process*, vol. 20, págs. 375–390, 1982.
- [46] J.A. Fletcher y R. Kasturi, "A robust algorithm for text string from mixed text / graphics images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, págs. 910–918, 1988.
- [47] T. Taxt, P.J. Flynn, y A.K. Jain, "Segmentation of document images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, págs. 1322–1329, 1989.
- [48] A.K. Jain y S.K. Bhattacharjee, "Text segmentation using Gabor filters for automatic document processing," *Mach. Vision and Appl.*, vol. 5, págs. 169–184, 1992.
- [49] J. Puzicha, T. Hofmann, y J.M. Buhman, "Non-parametric similarity measure for unsupervised texture segmentation and image retrieval," en *Computer Vision and Pattern Recognition*, 1997, págs. 262–272.
- [50] R.M. Haralick, K. Shanmugam, y I. Dinstein, "Textural features for image classification," *IEEE Trans. Syst. Man Cyberm.*, vol. 3, págs. 610–621, 1973.
- [51] E. Rignot y R. Kwok, "Extraction of textural features in SAR images: Statistical model and sensitivity," en *Proc. Int. Geoscience and Remote Sensing Symp., Washington, DC*, 1990.
- [52] A.H. Schistad y A.K. Jain, "Texture analysis in the presence of speckle noise," en *Proc. IEEE Geoscience and Remote Sensing Symp., Houston, TX*, 1992.
- [53] L.J. Du, "Texture segmentation of SAR images using localized spatial filtering," en *Proc. Int. Geoscience and Remote Sensing Symp., Washington, DC*, 1990.
- [54] J.H. Lee y W.D. Philpot, "A spectral textural classifier for digital imagery," en *Proc. Int. Geoscience and Remote Sensing Symp., Washington, DC*, 1990.
- [55] E. Monjoux y J.P. Rudant, "Texture segmentation in aerial images," en *Image Processing Algorithms and Techniques II, San Jose*, 1991.
- [56] J.C. Bezdek, L.O. Hall, M. Clark, D. Goldgof, y L.P. Clarke, "Segmenting medical images with fuzzy models: An update," en *Fuzzy Information Engineering. A Guided Tour of Applications*, D. Dubois, H. Prade, y R.R. Yager, Eds., capítulo 4. Wiley, 1997.

-
- [57] A.M. Tekalp, *Digital Video Processing*, Prentice Hall, Upper Saddle River, NJ, 1995.
- [58] R.O. Duda y P.E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, 1973.
- [59] A.J. Bell y T.J. Sejnowski, "The independent components of natural scenes are edge filters," *Vision Res.*, vol. 37, núm. 23, págs. 3327–3338, 1997.
- [60] D. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *Journal of the Optical Society of America A*, vol. 4, núm. 12, págs. 2379–2394, 1987.
- [61] R.J. Clarke, *Transform Coding of Images*, Academic Press, New York, 1985.
- [62] B.A. Olshausen y D. Field, "Emergence of simple-cell receptive field properties by a learning a sparse code for natural images," *Nature*, vol. 381, págs. 607–609, 1996.
- [63] E.P. Simoncelli y B.O. Olshausen, "Natural image statistics and neural representation," *Annu. Rev. Neurosci.*, vol. 24, págs. 1193–1216, 2001.
- [64] P.C. Teo y D.J. Heeger, "Perceptual image distortion," *Proc. of the First IEEE Intl. Conf. Im. Proc.*, vol. 2, págs. 982–986, 1994.
- [65] J. Malo, A.M. Pons, y J.M. Artigas, "Subjective image fidelity metric based on bit allocation of the human visual system in the DCT domain," *Image & Vision Computing*, vol. 15, núm. 7, págs. 535–548, 1997.
- [66] A.M. Pons, J. Malo, J.M. Artigas, y P. Capilla, "Image quality metric based on multidimensional contrast perception models," *Displays*, vol. 20, págs. 93–110, 1999.
- [67] S. Winkler, "Issues on vision modeling for video quality assessment," *Sig. Process.*, vol. 4, núm. 12, págs. 2401–2417, 1999.
- [68] A. Gersho y R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Press, Boston, 1992.
- [69] Y.Q. Shi y H. Sun, *Image Compression for Multimedia Engineering: fundamentals, algorithms and standards*, CRC Press, Boca Raton, 2000.
- [70] G.K. Wallace, "The JPEG still picture compression standard," *Communications of the ACM*, vol. 34, núm. 4, págs. 31–43, 1991.
- [71] J. Malo, F. Ferri, J. Albert, J.Soret, y J.M. Artigas, "The role of perceptual contrast non-linearities in image transform coding," *Image & Vision Computing*, vol. 18, núm. 3, págs. 233–246, 2000.
- [72] I. Epifanio y G. Ayala, "Random set view of texture classification," *IEEE Transactions on Image Processing*, aceptado para su publicación.

- [73] G. Matheron, *Random Sets and Integral Geometry*, Wiley, London, 1975.
- [74] D. Stoyan, W.S. Kendall, y J. Mecke, *Stochastic Geometry and its Applications*, Wiley, Berlin, segunda edición, 1995.
- [75] J. Goutsias, “Modeling random shapes: An introduction to random closed set theory,” Tech. Rep. JHU/ECE 90-12, Department of Electrical and Computer Engineering, Image Analysis and Communications Laboratory, The John Hopkins University, Baltimore, MD 21218, Abril 1998.
- [76] D.J. Nott y R.J. Wilson, “Multi-phase image modelling with excursion sets,” *Signal Processing*, vol. 80, págs. 125–139, 2000.
- [77] G. Ayala y A. Simó, “Bivariate random closed sets and nerve fibre degeneration,” *Advances in Applied Probability*, vol. 27, págs. 293–305, 1995.
- [78] G. Ayala y A. Simó, “Stochastic labelling of biological images,” *Statistica Neerlandica*, vol. 52, núm. 2, págs. 141–152, 1998.
- [79] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, 1982.
- [80] L. Vincent, *Stochastic Geometry Likelihood and Computation*, vol. 80 of *Monographs on Statistics and Applied Probability*, capítulo 6, págs. 199–283, Chapman & Hall/CRC, 1999.
- [81] I. Molchanov, *Statistics of the Boolean Model for Practitioners and Mathematicians*, John Wiley and Sons, Chichester, 1997.
- [82] P. García, M. Petrou, y S. Kamata, “The use of Boolean model for texture analysis of grey images,” *Computer Vision and Image Understanding*, vol. 74, núm. 3, págs. 227–235, June 1999.
- [83] P. García y M. Petrou, “Classification of binary textures using the one-dimensional Boolean model,” *IEEE Transactions on Image Processing*, vol. 8, núm. 10, págs. 1457–1462, 1999.
- [84] P. García y M. Petrou, “Analysis of irregularly shaped texture regions,” *Computer Vision and Image Understanding*, vol. 84, págs. 62–76, 2001.
- [85] P. Soille, *Morphological Image Analysis. Principles and Applications*, Springer-Verlag, 1999.
- [86] W.H. Press, B.P. Flannery, S.A. Teulosky, y W.T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press, Cambridge, 1992.
- [87] J. Goutsias y S. Batman, *Handbook of Medical Imaging. Volume 2. Medical Image Processing and Analysis*, capítulo 4, págs. 175–272, SPIE Optical Engineering Press, 2000.

-
- [88] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*, Dover, New York, 1966.
- [89] E.R. Dougherty, J.T. Newell, y J.B. Pelz, "Morphological texture-based maximum-likelihood pixel classification based on local granulometric moments," *Pattern Recognition*, vol. 25, núm. 10, págs. 1181–1198, 1992.
- [90] L.M. Kaplan, "Extended fractal analysis for texture classification and segmentation," *IEEE Transactions on Image Processing*, vol. 8, núm. 11, págs. 1572–1585, 1999.
- [91] R.W. Connors y C.A. Harlow, "A theoretical comparison of texture algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, págs. 204–222, 1980.
- [92] S. Fogel y D. Sagi, "Gabor filters as texture discriminators," *Journal of Biological Cybernetics*, vol. 61, págs. 103–113, 1989.
- [93] G. Smith y I. Burns, "Measuring texture classification algorithms," *Pattern Recognition Letters*, vol. 18, págs. 1495–1501, 1997.
- [94] M.H. DeGroot, *Optimal statistical decisions*, McGraw-Hill, New York, 1970.
- [95] B.W. Silverman, *Density estimation for statistics and data analysis*, Champan and Hall, London, 1992.
- [96] B. Manly, *Randomization, Bootstrap and Monte Carlo Methods in Biology*, Chapman and Hall, second edition edición, 1997.
- [97] P. Good, *Permutation Tests. A Practical Guide to Resampling Methods for Testing Hypotheses*, Springer-Verlag, 1994.
- [98] A.C. Davison y D.V. Hinkley, *Bootstrap Methods and their Application*, Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1997.
- [99] P.J. Diggle, *Statistical Analysis of Spatial Point Patterns*, Academic Press, London, 1983.
- [100] G.J. Klir y T.A. Folger, *Fuzzy Sets, Uncertainty and Information*, Prentice Hall, 1988.
- [101] M.L. Puri y D.A. Ralescu, "Fuzzy random variables," *J. Math. Anal. Appl.*, vol. 114, págs. 409–422, 1986.
- [102] K.V. Mardia, J.T. Kent, y J.M. Bibby, *Multivariate Analysis*, Academic Press, 1989.
- [103] R. Krishnapuram y J.M. Keller, "A possibilistic approach to clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, núm. 2, págs. 98–110, mayo 1993.

-
- [104] P.P. Ohanian y R.C. Dubes, "Performance evaluation for four classes of textural features," *Pattern Recognition*, vol. 25, núm. 8, págs. 819–833, 1992.
- [105] I. Epifanio, J. Gutiérrez, y J. Malo, "Linear transform for simultaneous diagonalization of covariance and perceptual metric matrix in image coding," *enviado a Pattern Recognition*.
- [106] H.R. Wilson, "Pattern discrimination, visual filters and spatial sampling irregularities," en *Computational Models of Visual Processing*, M.S. Landy y J.A. Movshon, Eds., Massachussets, 1991, págs. 153–168, MIT Press.
- [107] A.B. Watson y J.A. Solomon, "A model of visual contrast gain control and pattern masking," *Journal of the Optical Society of America A*, vol. 14, págs. 2379–2391, 1997.
- [108] R.J. Clarke, "Relation between the Karhunen-Loève and cosine transforms," *Proc. IEE, Pt. F*, vol. 128, núm. 6, págs. 359–360, 1981.
- [109] D. Kersten, "Predictability and redundancy of natural images," *Journal of the Optical Society of America A*, vol. 4, núm. 12, págs. 2395–2400, 1987.
- [110] B. Dubrovin, S. Novikov, y A. Fomenko, *Modern Geometry: Methods and Applications*, Springer Verlag, New York, 1982.
- [111] A.J. Ahumada y H.A. Peterson, "Luminance-model-based DCT quantization for color image compression," en *Proceedings of the SPIE*, 1992, vol. 1666, págs. 365–374.
- [112] A.B. Watson, "DCT quantization matrices visually optimized for individual images," en *Human Vision, Visual Processing and Digital Display IV*, B.E. Rogowitz, Ed., 1993, vol. 1913-14 of *Proceedings of the SPIE*.
- [113] J.A. Solomon, A.B. Watson, y A.J. Ahumada, "Visibility of DCT basis functions: effects of contrast masking," en *Proceedings of Data Compression Conference, Snowbird, Utah*, 1994, IEEE Computer Society Press, págs. 361–370.
- [114] D. LeGall, "MPEG: A video compression standard for multimedia applications," *Communications of the ACM*, vol. 34, núm. 4, págs. 47–58, 1991.
- [115] J. Malo, A.M. Pons, y J.M. Artigas, "Bit allocation algorithm for codebook design in vector quantization fully based on human visual system non-linearities for suprathreshold contrasts," *Electronics Letters*, vol. 31, núm. 15, págs. 1229–1231, 1995.
- [116] J. Malo, F. Ferri, J. Albert, y J. Soret, "Comparison of perceptually uniform quantization with average error minimization in image transform coding," *Electronics Letters*, vol. 35, núm. 13, págs. 1067–1068, 1999.

- [117] J. Malo, J. Gutiérrez, I. Epifanio, F. Ferri, y J.M. Artigas, “Exploiting perceptual feed-back in multigrid motion estimation using an improved DCT quantization,” *IEEE Transactions on Image Processing*, vol. 10, núm. 10, págs. 1411–1427, 2001.
- [118] E.P. Simoncelli y O. Schwartz, “Modeling surround suppression in V1 neurons with a statistically-derived normalization model,” en *Adv. in Neural Inf. Proc. Syst.*, M.S. Kearns, Ed. 1999, vol. 11, MIT Press.
- [119] D.J. Heeger, “Normalization of cell responses in cat striate cortex,” *Visual Neuroscience*, vol. 9, págs. 181–198, 1992.
- [120] M. Carandini y D. Heeger, “Summation and division by neurons in visual cortex,” *Science*, vol. 264, págs. 1333–1336, 1994.
- [121] F.W. Campbell y J.G. Robson, “Application of Fourier analysis to the visibility of gratings,” *Journal of Physiology*, vol. 197, págs. 551–566, 1968.
- [122] A.M. Pons, *Estudio de las Funciones de Respuesta al Contraste del Sistema Visual*, Ph.D. thesis, Dpt. d’Òptica, Facultat de Física, Universitat de València, Julio 1997.
- [123] G.E Legge y J.M. Foley, “Contrast masking in human vision,” *Journal of the Optical Society of America*, vol. 70, págs. 1458–1471, 1980.
- [124] G.E. Box y D.R. Cox, “An analysis of transformations. (With discussion),” *J. Royal Statistical Society. Serie B*, vol. 26, págs. 211–252, 1964.
- [125] W. Niehsen y M. Brunig, “Covariance analysis of motion-compensated frame differences,” *IEEE Trans. Circ. Syst. Video Tech*, vol. 9, núm. 4, págs. 536–539, 1999.
- [126] W.R. Bennett, “Spectra of quantized signals,” *Bell Syst. Tech. J.*, vol. 27, págs. 446–472, 1948.
- [127] Y. Linde, A. Buzo, y R.M. Gray, “An algorithm for vector quantizer design,” *IEEE Transactions on Communications*, vol. 28, núm. 1, págs. 84–95, 1980.
- [128] J. Malo, R. Navarro, I. Epifanio, F.J. Ferri, y J.M. Artigas, “Non-linear invertible representation for joint statistical and perceptual feature decorrelation,” *Lecture Notes in Computer Science*, vol. 1876, págs. 658–667, 2000.
- [129] A. Bimbo, *Visual Information Retrieval*, Morgan Kaufman Publishers, NY, 1999.
- [130] H. Li, B.S. Manjunath, y S.K. Mitra, “Multisensor image fusion using the wavelet transformation,” *Graph. Mod. Im. Proc.*, vol. 57, núm. 3, págs. 235–245, 1995.
- [131] G.H. Golub y C.F. Van Loan, *Matrix computations*, Johns Hopkins University Press, Baltimore, 1996.
- [132] A.N. Akansu y R.A. Haddad, *Multiresolution Signal Decomposition*, Academic Press, Boston, 1992.

- [133] R.W. Buccigrossi y E.P. Simoncelli, "Image compression via joint statistical characterization in the wavelet domain," *IEEE Trans. Im. Proc.*, vol. 8, núm. 12, págs. 1688–1701, 1999.
- [134] S.D. Conte y C. De Boor, *Análisis numérico elemental: un enfoque algorítmico*, McGraw-Hill, México, 1977.
- [135] J.D. Lambert, *Numerical methods for ordinary differential systems. The initial value problem*, Wiley, 1991.