

# UN FILTRO PARA LA ELIMINACIÓN DE RUIDO MEDIANTE CORRELACIÓN ESPACIAL EN IMÁGENES

Por

Carlos Damián Estrada Chávez

Tesis sometida para el cumplimiento parcial de  
los requisitos para obtener el grado de

Maestro en Ciencias con especialidad en Computación y Matemáticas  
Industriales

Director de tesis: Dr. Mariano J. J. Rivera Meraz

Co-director: Dr. Johan J. Van Horebeek



CENTRO DE INVESTIGACIÓN EN MATEMÁTICAS  
A.C.

Julio 2012

## Nota

Sobre los enlaces a páginas WEB contenidos en este documento.

La finalidad de las referencias a páginas web, es la de señalar términos clave y facilitar la búsqueda de estos al lector. La fuente de consulta referida debe considerarse como opcional y, dado el carácter dinámico de la misma, su validez queda a juicio propio del lector.

Así mismo, la información contenida en las páginas web referidas, no sirven de apoyo a los argumentos aquí vertidos y, por lo tanto, su validez tampoco se ve comprometida por el contenido o la inexistencia de las fuentes enlazadas.

Las citas a páginas web se encontrarán como notas al pie de página, mientras que el material bibliográfico en el que se apoya esta tesis se encuentra en la sección de *Bibliografía*.

# Resumen

En la presente tesis se propone un algoritmo novedoso para el filtrado de imágenes en niveles de gris y color. El concepto central sobre el cual se basa dicho algoritmo es una medida de correlación espacial: diremos que dos colores están correlacionados espacialmente si estos aparecen cercanos en la imagen con mucha frecuencia, con respecto a los otros pares de colores. En el caso discreto, dicha medida de correlación espacial se representa mediante una matriz de Adyacencia, la cual puede calcularse de manera eficiente. Se presenta primero un filtro iterativo con un esquema simple de 4 vecinos y promedios locales pesados. Posteriormente, se plantea un filtro no iterativo y se plantea una estrategia para estimar sus parámetros (el filtro propuesto es autocalibrable). Dada su similitud en implementación con un filtro bilateral (FB) (también se implementa mediante la iteración de promedios locales pesados) se realizan comparaciones con este tipo de filtros; usando una técnica del estado del arte para estimar los parámetros en el FB. Se puede constatar que el filtro propuesto en general produce mejores resultados que el FB.

Términos clave: filtro de imágenes, correlación espacial, autocalibración, filtro bilateral.

## *Dedicatoria*

*A mis padres, hermanos, amigos y demás familiares,*

*a Judith*

*y a Marla...*

*Gracias.*



*“Alegoría de la caverna de Platón”*

# Agradecimientos

Primeramente agradezco a todo el personal de CIMAT y especialmente al área de Ciencias Computacionales por aceptarme como estudiante de maestría. Gracias también a las autoridades de CONACyT por su labor en la administración de becas, y gracias por supuesto, a todos los ciudadanos que pagan sus impuestos y que de esta forma nos patrocinan financieramente a tantos alumnos de posgrado. Doy las gracias a todos mis compañeros de maestría, quienes solidariamente supieron resolver dudas y animar los momentos más extenuantes. Por último agradezco a mis profesores, especialmente a mis directores de tesis: el Dr. Johan Van Horebeek y el Dr. Mariano Rivera Meraz; y a mis sinodales: Dr. Jean-Bernard Hayet y el Dr. José Luis Marroquín. Mi mas sincero agradecimiento a todos ellos, por lo que me han ofrecido durante estos años.

# Tabla de contenido

Resumen .....	2
Prólogo .....	6
<b>Capítulo 1</b> Correlación espacial en imágenes .....	8
1.1 Prerrequisito: la imagen de etiquetas .....	9
1.2 Matriz de adyacencia (MA).....	11
1.3 Matriz de transición (MT) .....	17
<b>Capítulo 2</b> Correlación espacial en el problema de eliminación de ruido en imágenes.....	20
2.1 Heurísticas para la reducción de ruido .....	20
2.2 La correlación espacial en la inferencia. ....	21
<b>Capítulo 3</b> El filtro ‘4-conectado’.....	24
3.1 Algoritmo.....	24
3.2 Suavizado de imágenes .....	25
3.3 Eliminación de ruido.....	29
<b>Capítulo 4</b> Adaptabilidad al ruido .....	31
4.1 El filtro bilateral como punto de referencia .....	31
4.2 Similitud ‘contextual’.....	34
4.3 Cercanía y tamaño de la vecindad .....	36
<b>Capítulo 5</b> Comentarios finales y conclusiones .....	45
5.1 Resumen de resultados .....	45
5.2 Conclusiones .....	45
Anexo I Código fuente (Matlab).....	58
Filtro 4-conectado (scor4filt). ....	58
Filtro por correlación espacial (scorfilt). ....	60
Inferencia de las intensidades mediante el espectro. ....	63
Anexo II Resultados .....	65
Bibliografía .....	67

# Prólogo

Considérese la siguiente imagen contaminada con ruido:



---

Imagen *'boats'* contaminada con ruido.

A pesar de que el nivel de ruido es alto, seguramente el lector alcanzará a distinguir algunas de las figuras que contendría la imagen original. En el presente trabajo

abordaremos el tema de eliminar el ruido en imágenes digitales. Se asumirá que el lector posee conocimientos básicos sobre procesamiento de imágenes.

El problema de eliminar el ruido de imágenes digitales ha sido abordado desde varias perspectivas (Morel, 2005) y se continúan proponiendo nuevos algoritmos, obteniéndose cada vez mejores resultados (Egiazarian, 2009). En la práctica, sin embargo, filtros tradicionales como el bilateral (Tomasi & Manduchi, 1998) continúan siendo de uso común debido a su sencillez y relativa eficacia para eliminar el ruido.

Con este trabajo intentamos introducir un nuevo enfoque motivado en una heurística simple: en una imagen, los píxeles que apreciamos sin ruido, corresponden a los que poseen colores más “correlacionados” entre sí. Apoyados en la imagen anterior, uno puede reflexionar si la heurística propuesta tiene algún sentido. De forma más específica, definiremos una medida de correlación espacial, la cual nos indicará cuando la presencia de un color implique la presencia de otro a una distancia determinada.

La anterior medida tendría una justificación estadística: los datos relevantes son los más correlacionados. Pero desde el punto de vista de visión computacional, o de manera más general, de la percepción visual (de como los humanos procesan la información visual), será necesario justificar el uso de una medida estadística en la descripción de una característica perceptual.

## Capítulo 1

# Correlación espacial en imágenes

El propósito de este capítulo es definir una medida de correlación espacial en imágenes digitales e introducir algunos conceptos que se usarán en los capítulos posteriores. Representaremos a una *imagen digital*  $I$  mediante una función  $I : \Omega \rightarrow \Phi$  en donde tanto el dominio  $\Omega$  de *coordenadas espaciales* como el conjunto de *valores*  $\Phi$  son finitos y discretos. Llamaremos *posición* a cada elemento de  $\Omega$  e *intensidad* a cada elemento de  $\Phi$ , siempre y cuando este sea unidimensional; es decir, cuando se haga referencia a una imagen en escala de gris. De lo contrario, diremos que cada elemento de  $\Phi$  es un *color*, debido a que en una imagen digital a color, este valor *codifica* el color de manera tal que pueda ser interpretado y realizado por algún dispositivo.

Una manera de codificar el color, es hacer referencia a un *espacio de color*, de manera que los valores de  $\Phi$  representen coordenadas dentro de dicho espacio. De esta forma es posible calcular una *distancia entre colores* en el espacio de color en el cual están representados. Así pues, en este trabajo, hablaremos de una medida entre colores en un sentido muy distinto al anterior; nos interesará medir el grado de asociación entre los colores en base a su distribución espacial en la imagen, lo cual llamaremos *correlación espacial entre colores*.

La medida de correlación que se presenta es nada rebuscada: *diremos que dos colores están correlacionados espacialmente si estos aparecen cercanos en la imagen con mucha frecuencia, con respecto a los otros pares de colores*.

Aquí debemos resaltar los siguientes puntos:

- La medida de correlación espacial entre colores es relativa a una imagen en particular.

- Puede entenderse desde un punto de vista enteramente estadístico, y es aplicable a cualquier señal digital o serie de tiempo.

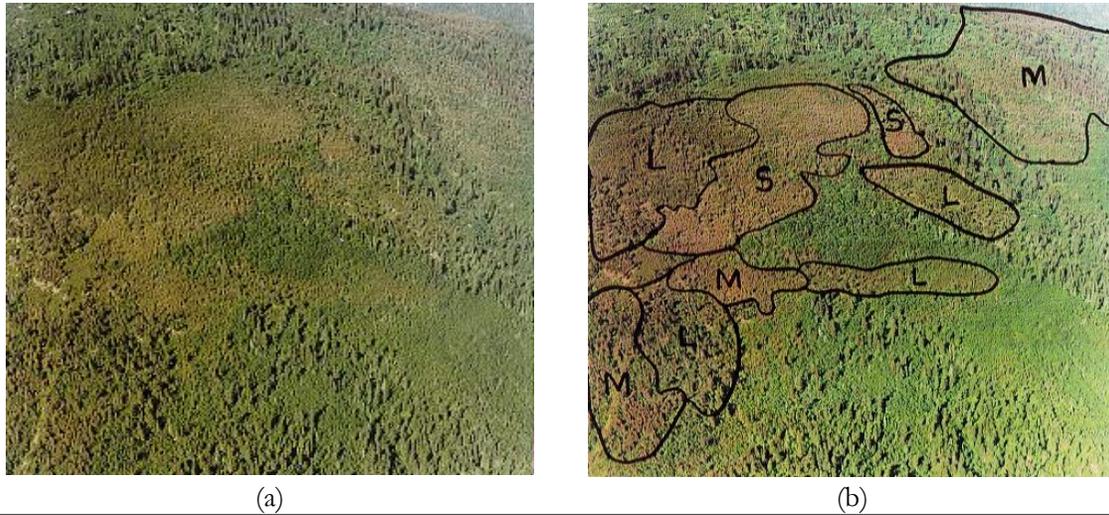


Fig. 1-1. Clasificación subjetiva de regiones en una imagen a color<sup>1</sup>. (a) Fotografía aérea de zona boscosa afectada por plaga. (b) Identificación y clasificación del daño en ligero (L), moderado (M) y severo (S). Las distintas tonalidades de color están correlacionadas espacialmente.

## 1.1 Prerrequisito: la imagen de etiquetas

Para poder calcular la correlación espacial entre cada par de colores en una imagen, es necesario realizar una tarea implícita: identificar todas las parejas de colores posibles. Muchas veces no será posible identificar todos los pares de colores, debido a que el número de maneras en que se pueden combinar es muy grande e implica un problema de capacidad de procesamiento. Por ejemplo, para una imagen a color típica en formato RGB, los canales 'R', 'G' y 'B' son digitalizados comúnmente con 8 bits cada uno, esto es  $256^3 = 16,777,216$  posibles colores distintos, por lo tanto tendríamos  $256^6/2$  pares no ordenados de colores.

<sup>1</sup> Fuente de las imágenes: "Forest Health Aerial Overview Survey Standards for British Columbia", Apéndice I.

<http://www.ilmb.gov.bc.ca/risc/pubs/teveg/foresthealth/aerial-11.htm>

B.C. Ministry of Forests, Forest Practices Branch and Canadian Forest Service, Junio 2000.

Es necesario entonces que el número de parejas de colores sea adecuado a las capacidades de procesamiento. Para reducir el número de colores identificamos los colores similares asignándoles una sola *etiqueta*; a este procedimiento de reducir el número de colores en la imagen, se le conoce en la literatura como *cuantización de color*<sup>2</sup>. Nos referiremos con *etiquetas* a un conjunto de índices (ej.  $\{1,2,3,4,\dots\}$ ) que identifican la clase o estado de un pixel después de la cuantización.

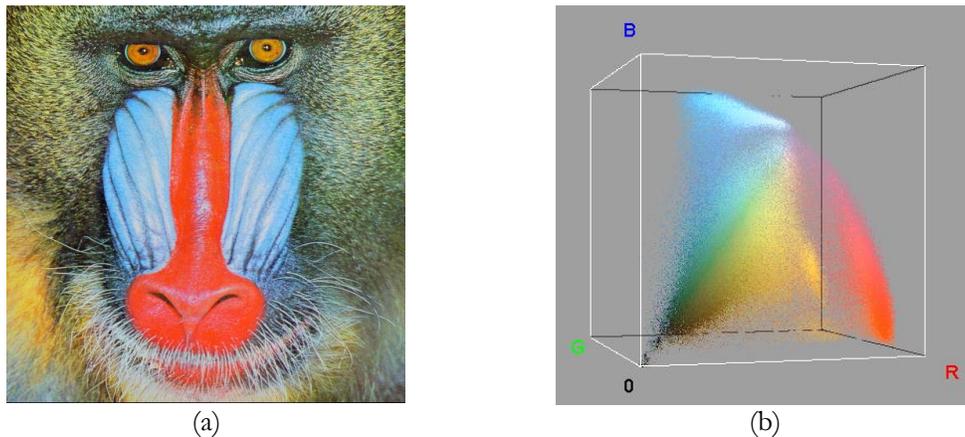


Fig. 1-2: Imagen ‘baboon’ (a) la cual tiene 230,427 colores distintos y su representación en el espacio de color RGB (b) obtenida con *Color Inspector 3D*.

Llamaremos *imagen de etiquetas* correspondiente a una imagen, a un arreglo bidimensional con el mismo alto y ancho de la imagen pero cuyos ‘valores’ son etiquetas que identifican a un color, o a varios colores similares. En otras palabras cambiamos colores por etiquetas. Nótese que una imagen típica en tonos de gris, donde la imagen consiste de un solo canal, es por sí misma una imagen de etiquetas.

<sup>2</sup> Cuantización de color: [http://en.wikipedia.org/wiki/Color\\_quantization](http://en.wikipedia.org/wiki/Color_quantization)

Resumen de algunas técnicas: [http://web.cs.wpi.edu/~matt/courses/cs563/talks/color\\_quant/CQindex.html](http://web.cs.wpi.edu/~matt/courses/cs563/talks/color_quant/CQindex.html)

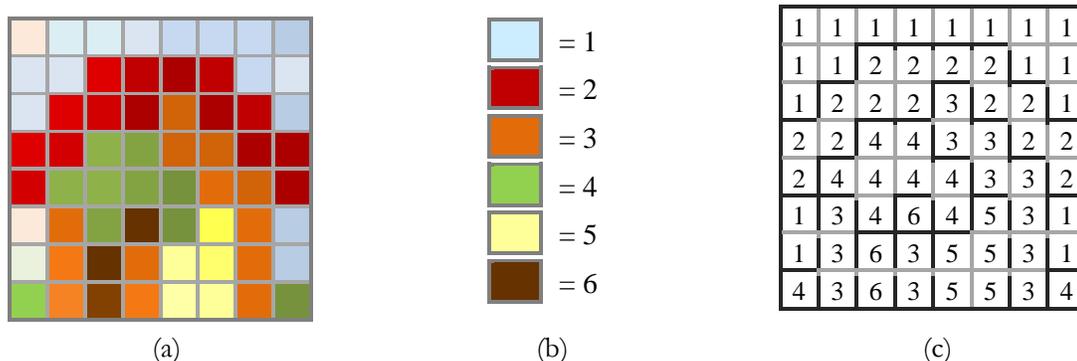


Fig. 1-3. Ejemplo minimalista de generación de una *imagen de etiquetas*.

- a) representación de una imagen de 8x8 píxeles.
- b) mapeo de cada color a una etiqueta (paleta de color).
- c) imagen de etiquetas resultante.

Esta manera de codificar una imagen mediante etiquetas es conocida como *indexado de color*<sup>3</sup>. Para cualquier función  $L$  predefinida de color indexado (que por lo general requerirá de cuantizar la imagen), dada la imagen  $I$ , obtenemos la *imagen de etiquetas*  $\tilde{I}$ , i.e.:

$$\tilde{I} = L(I) \tag{1.1}$$

Denotaremos con  $\Lambda$  al conjunto de las posibles etiquetas, es decir  $\Lambda = \text{Im}(\tilde{I})$ .

## 1.2 Matriz de adyacencia (MA)

Como se discutía en la introducción, la medida de correlación propuesta se apoyaría en la idea intuitiva de contar las veces que dos colores aparecieran en píxeles cercanos en la imagen. Así pues, dada una *función de cercanía*  $f$  que evalúa si la distancia entre dos píxeles se encuentra dentro de cierta escala espacial, y una *función de similitud*  $g$  que

<sup>3</sup> Indexado de color: [http://en.wikipedia.org/wiki/Indexed\\_color](http://en.wikipedia.org/wiki/Indexed_color)

indica el grado de semejanza entre dos valores del rango<sup>4</sup> de una señal  $s$  (que denotaremos como  $u$  y  $v$ ), definimos la *función de correlación espacial*  $\mathbf{U}$  de la señal  $s$  para el caso continuo como:

$$\mathbf{U}(u, v) = \int_{x \in \mathbb{R}^2} g(s(x) - u) \int_{z \in \mathbb{R}^2} f(x - z) g(s(z) - v) dz dx \quad (1.2)$$

donde  $u, v \in \text{Im}(s)$ , y  $f$  y  $g$  son funciones pares monótonas decrecientes en  $(0, \infty)$  y con rango en  $(0, 1)$ .

Alternativamente, realizando el cambio de variable  $t = x - z$ , escribimos:

$$\mathbf{U}(u, v) = \int_{x \in \mathbb{R}^2} g(s(x) - u) \int_{t \in \mathbb{R}^2} f(t) g(s(x - t) - v) dt dx \quad (1.3)$$

La función  $\mathbf{U}$  puede leerse, *grosso modo*, como la certeza de que en la señal  $s$ , las posiciones con valores similares a  $u$ , se encuentren cercanas a posiciones cuyos valores sean similares a  $v$ . De forma análoga, para  $I$  de dominio discreto tenemos:

$$\mathbf{U}(u, v) = \sum_{\forall p \in \Omega} g(I_p - u) \sum_{\forall q \in \Omega} f(p - q) g(I_q - v) \quad (1.4)$$

Es de particular interés obtener la correlación espacial sobre una imagen de etiquetas, lo cual es posible si  $g$  está definida por la *función indicadora*  $\mathbf{1}$ . Así para las etiquetas  $a$  y  $e$  tendríamos:

$$\mathbf{U}_{a,e} = \sum_{\forall p \in \Omega} \left( \mathbf{1}[\tilde{I}_p = a] \sum_{\forall q \in \Omega} f(p - q) \mathbf{1}[\tilde{I}_q = e] \right) \quad (1.5)$$

Aquí consideraremos solamente el caso más económico de calcular, en donde contaríamos solamente el número de veces que dos colores están en pixeles contiguos

---

<sup>4</sup> Nos referiremos con *rango* de una función cualquiera  $F$  al conjunto  $\text{Im}(F)$ .

en la imagen<sup>5</sup>. Entonces, dada la imagen de etiquetas  $\tilde{I}$ , con rango  $\Lambda = \{1, 2, \dots, N\}$ , su *matriz de adyacencia (MA)*  $\mathbf{A}_{N \times N}$ , es aquella donde cada una de sus entradas  $\mathbf{A}_{a,e}$ ,  $a, e \in \Lambda$ , toma el valor del conteo de coincidencias entre los pixeles contiguos etiquetados con los números  $a$  y  $e$ , i.e.:

$$\mathbf{A}_{a,e} = \sum_{\forall p \in \Omega} \mathbf{1}[\tilde{I}_p = a] \sum_{k=1}^4 \mathbf{1}[\tilde{I}_{(p-t_k)} = e] \quad (1.6)$$

donde  $t_1 = (1, 0)$ ,  $t_2 = (0, 1)$ ,  $t_3 = (-1, 0)$ ,  $t_4 = (0, -1)$ .

1	1	1	1	1	1	1	1
1	1	2	2	2	2	1	1
1	2	2	2	3	2	2	1
2	2	4	4	3	3	2	2
2	4	4	4	4	3	3	2
1	3	4	6	4	5	3	1
1	3	6	3	5	5	3	1
4	3	6	3	5	5	3	4

Fig. 1-4. La matriz de adyacencia se puede formar considerando una vecindad de *4-conectado*, esto es, contando los enlaces del pixel del centro contra el de arriba, abajo, izquierda y derecha.

El siguiente algoritmo muestra lo económico que resulta calcular la MA.

<sup>5</sup> Procedimientos similares son conocidos como *conteo de uniones (join count)* dentro de la bibliografía relacionada con estadística espacial o análisis espacial.

---

**Algoritmo 1-1. Generación de la matriz de adyacencia.**

---

*Entrada:* imagen de etiquetas  $\tilde{I}$ , número de etiquetas  $N$ .

*Salida:* matriz de adyacencia  $A_{N \times N}$ .

Definir:

$m, n$ : alto y ancho de  $\tilde{I}$

$$\mathbf{A} = \mathbf{0}_{N \times N}$$

Para cada  $i=1, \dots, m$

  Para cada  $j=1, \dots, n$

    Si  $i > 1$

$$\mathbf{A}(\tilde{I}_{(i,j)}, \tilde{I}_{(i-1,j)}) = \mathbf{A}(\tilde{I}_{(i,j)}, \tilde{I}_{(i-1,j)}) + 1$$

    Fin

    Si  $j > 1$

$$\mathbf{A}(\tilde{I}_{(i,j)}, \tilde{I}_{(i,j-1)}) = \mathbf{A}(\tilde{I}_{(i,j)}, \tilde{I}_{(i,j-1)}) + 1$$

    Fin

  Fin  $j$

Fin  $i$

$$\mathbf{A} = \mathbf{A} + \mathbf{A}^T$$


---

1	1	1	1	1	1	1	1
1	1	2	2	2	2	1	1
1	2	2	2	3	2	2	1
2	2	4	4	3	3	2	2
2	4	4	4	4	3	3	2
1	3	4	6	4	5	3	1
1	3	6	3	5	5	3	1
4	3	6	3	5	5	3	4

(a)

	1	2	3	4	5	6
1	34	14	4	2	0	0
2	14	30	7	5	0	0
3	4	7	20	7	6	5
4	2	5	7	16	2	4
5	0	0	6	2	10	0
6	0	0	5	4	0	2

(b)

---

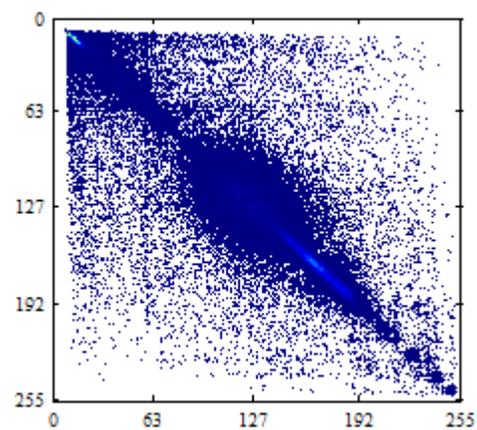
Fig. 1-5: (a) Imagen de etiquetas de la Fig. 1-3 y (b) su correspondiente matriz de adyacencia. La MA es simétrica por construcción.

**Nota:** Un conteo manual indicará que los pixeles vecinos del mismo color deben contarse 2 veces. Tómese por ejemplo el par de seis.

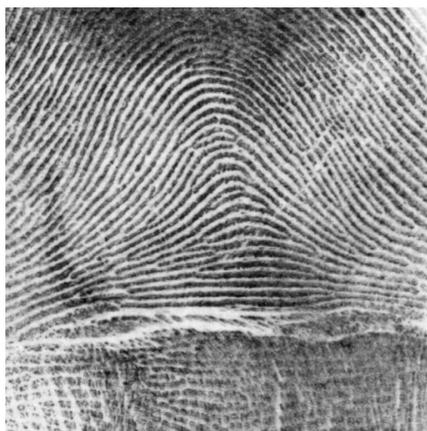
La primera observación que se hace sobre la MA, es que generalmente las etiquetas correspondientes a colores similares resultan estar correlacionadas espacialmente; esto es debido a que, en una imagen comúnmente encontraremos regiones de pixeles con colores similares entre sí. Esto se aprecia en la siguiente figura en donde se toma la imagen en tono de gris como la imagen de etiquetas. Las graficas a la derecha de cada imagen, resultan después de ordenar los renglones y las columnas de la MA con respecto a su tono de gris asociado.



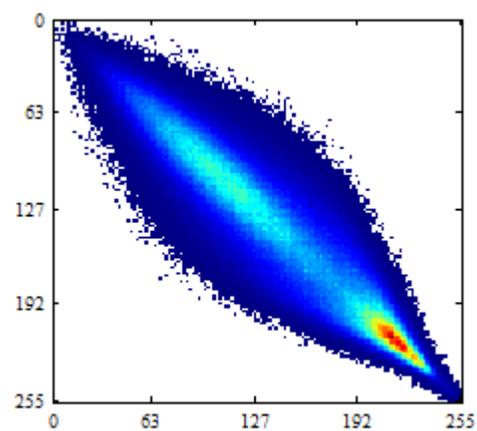
(a.1)



(a.2)



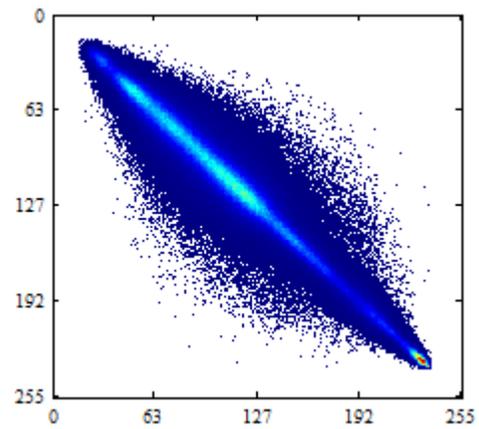
(b.1)



(b.2)



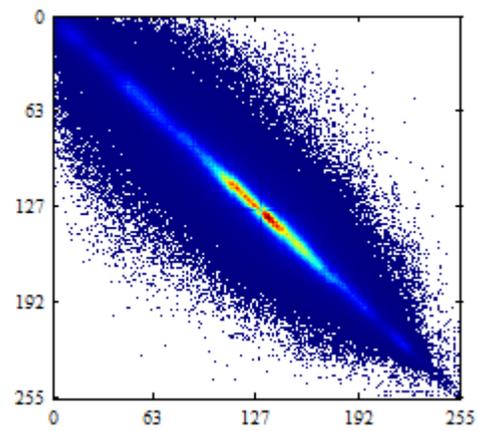
(c.1)



(c.2)



(d.1)



(d.2)

Fig. 1-6. Imágenes en tonos de gris y gráfica de su correspondiente MA (ver texto previo).

Para imágenes en tonos de gris, la MA (con las columnas y renglones ordenadas con respecto a la intensidad) nos proporciona una visualización de las variaciones en el rango de la imagen, esto es, los cambios suaves de color se encuentran cercanos a la diagonal mientras que las variaciones abruptas de color se encontraran en las bandas más lejanas. De hecho, la correlación espacial puede considerarse por si misma como una función de similitud entre dos etiquetas.

### 1.3 Matriz de transición (MT)

A partir de la MA podemos formar el *grafo de etiquetas* asociado, asignando cada etiqueta a un nodo, y cada entrada no nula de la MA a una arista que conecte a las etiquetas correspondientes. Dado que cada entrada de la MA puede interpretarse como la ‘longitud de la frontera’ entre dos etiquetas, entonces los pesos en las aristas del grafo de etiquetas poseen un significado análogo al de ‘capacidades de flujo’<sup>6</sup> entre etiquetas.

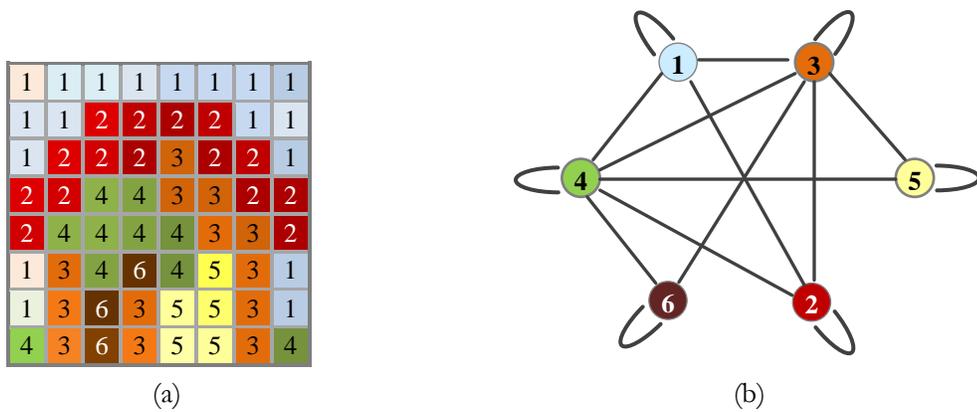


Fig. 1-7. (a) Representación de una imagen etiquetada y su (b) grafo de etiquetas (sin pesos).

Sera de especial interés, considerar la probabilidad de trasladarse de un nodo a otro, lo cual se obtienen al normalizar la MA de manera que obtengamos una *matriz estocástica*<sup>7</sup>. Llamaremos a esta, *matriz de transición* (MT)  $\mathbf{P}_{N \times N}$ .

$$\mathbf{P}_{a,e} = \frac{\mathbf{A}_{a,e}}{\sum_{i=1}^N \mathbf{A}_{a,i}} \quad (1.7)$$

donde  $\mathbf{P}_{a,e}$  es la probabilidad de pasar de un nodo con etiqueta  $a$  a otro con etiqueta  $e$ .

<sup>6</sup> El termino *capacidad* es usado comúnmente en problemas de *flujo en redes*.

<sup>7</sup> Una matriz estocástica es una matriz cuadrada con entradas reales no negativas, con cada renglón sumando 1.

	1	2	3	4	5	6
1	34	14	4	2	0	0
2	14	30	7	5	0	0
3	4	7	20	7	6	5
4	2	5	7	16	2	4
5	0	0	6	2	10	0
6	0	0	5	4	0	2

(a)

	1	2	3	4	5	6
1	0.6296	0.2593	0.0741	0.0370	0	0
2	0.2500	0.5357	0.1250	0.0893	0	0
3	0.0816	0.1429	0.4082	0.1429	0.1224	0.1020
4	0.0556	0.1389	0.1944	0.4444	0.0556	0.1111
5	0	0	0.3333	0.1111	0.5556	0
6	0	0	0.4545	0.3636	0	0.1818

(b)

Fig. 1-8. Matriz de adyacencia y matriz de transición correspondientes a la Fig. 1-5.

La matriz  $\mathbf{P}$  en general no será simétrica, por lo que el grafo de etiquetas con pesos dados por la MT sería *dirigido*. Una propiedad de la MT es que en una caminata sobre el grafo, con probabilidades de transición dadas por  $\mathbf{P}$ , visitaría cada nodo un número de veces proporcional al valor del histograma correspondiente a su etiqueta, al dar un número infinito de pasos. Lo anterior se deriva del hecho de que el vector de distribución estacionario de  $\mathbf{P}$  (el eigenvector izquierdo unitario correspondiente al eigenvalor 1) es igual al *histograma de la imagen de etiquetas* normalizado. Este hecho se puede comprobar directamente:

Empezaremos diciendo que la suma por columnas o por renglones de la matriz de adyacencia nos da aproximadamente el histograma de la imagen de etiquetas. La razón es porque por cada pixel, hemos contado los enlaces a sus 4 vecinos y después dividido entre 4, con excepción de los pixeles de los bordes de la imagen. Denotaremos al *histograma aproximado* de la imagen de etiquetas como:

$$\hat{\mathbf{h}} = \frac{1}{4} \mathbf{A} [1 \ 1 \ \dots \ 1]^T \quad (1.8)$$

Denotemos el *histograma aproximado normalizado* de la imagen de etiquetas.

$$\bar{\mathbf{h}} = \frac{\hat{\mathbf{h}}}{s} \quad (1.9)$$

donde,  $s = \sum_{i=1}^N \hat{\mathbf{h}}_i$  (aprox. el número total de pixeles en la imagen)

y la matriz diagonal  $\mathbf{D}$ , donde cada elemento de su diagonal es la suma por renglones de  $\mathbf{A}$ , i.e.:  $\mathbf{D} = \text{Diag}(4\hat{\mathbf{h}})$ .

Ahora calculamos,

$$\bar{\mathbf{h}}^T \mathbf{P} = \bar{\mathbf{h}}^T (\mathbf{D}^{-1} \mathbf{A}) = (\bar{\mathbf{h}}^T \mathbf{D}^{-1}) \mathbf{A}$$

donde,

$$\left[ \bar{\mathbf{h}}^T \mathbf{D}^{-1} \right]_i = \bar{\mathbf{h}}_i \left( \frac{1}{4\hat{\mathbf{h}}_i} \right) = \frac{\hat{\mathbf{h}}_i}{s} \frac{1}{4\hat{\mathbf{h}}_i} = \frac{1}{4s}$$

$$\bar{\mathbf{h}}^T \mathbf{P} = \frac{1}{4s} [1 \quad 1 \quad \dots \quad 1] \mathbf{A} = \frac{1}{s} \hat{\mathbf{h}}^T = \bar{\mathbf{h}}^T$$

esto es,

$$\bar{\mathbf{h}}^T \mathbf{P} = \bar{\mathbf{h}}^T \tag{1.10}$$

Lo cual implica que el histograma normalizado  $\bar{\mathbf{h}}$  es el eigenvector de  $\mathbf{P}$  asociado al eigenvalor igual a 1.

## Capítulo 2

# Correlación espacial en el problema de eliminación de ruido en imágenes

Aun y cuando no se haga explícito, todo algoritmo de filtrado de imágenes está fundado en supuestos heurísticos. Este hecho suele quedar oscurecido por los formalismos, de manera que el fenómeno natural subyacente -la percepción visual- suele pasar a segundo plano. Preguntas como, ¿Qué es el ruido?, ¿Cómo es un pixel que tiene ruido con respecto a los que no?, ¿Se relacionan de alguna forma conceptos como color, tamaño de las figuras en la imagen, y ruido? Todas estas preguntas, deberán ser relevantes para formar un filtro de imágenes con características ‘perceptuales’. Así pues, sobre el filtro propuesto, ha sido de especial interés encontrar una justificación consistente empíricamente.

### 2.1 Heurísticas para la reducción de ruido

Dada una imagen contaminada por ruido, ¿Cómo estimar la imagen original? Es claro que no sería posible inferir el verdadero valor de un pixel sin conocer los valores de algunos otros pixeles en la imagen; en efecto, es necesario establecer una *relación* entre el valor de cada pixel y el de otros pixeles. Considérese el siguiente ejemplo de inferencia del verdadero valor de un pixel:

1. *(Heurística)* Un pixel no tiene ruido si y solo si su valor es igual a la de los pixeles contiguos.
2. *(Observación)* En la imagen  $I$ , el pixel  $x$  con valor  $u$  tiene únicamente pixeles contiguos con valor  $v$ .

Luego,

3. *El pixel  $x$  tiene ruido.*

Y

4. *El verdadero valor del pixel  $x$  es  $v$ .*

Como se puede ver, se parte de una sentencia *heurística* que tomamos como regla, la cual *relaciona* los pixeles contiguos de la imagen. Ahora bien, ¿Es correcta la segunda conclusión del razonamiento anterior?, la respuesta es *no*, debido a que pudo darse el caso en que los pixeles contiguos al pixel  $x$  fueran alterados a un mismo valor distinto al que tenían originalmente; aun así, la conclusión es una buena hipótesis sobre el verdadero valor del pixel en cuestión<sup>8</sup>. Tenemos entonces que el nuevo valor se ha inferido a partir de la heurística planteada, aunque no en un sentido deductivo, por lo cual, nada impide que pudiera haber más de una manera de inferir el valor de un pixel bajo esta misma heurística (cosa que, de hecho, haremos más tarde). La heurística mencionada tiene un sentido *local*, es decir, describe lo que de forma intuitiva esperaríamos de una *región pequeña* de la imagen ‘sin ruido’.

## 2.2 La correlación espacial en la inferencia.

Considérese el siguiente argumento, el cual hace referencia a la MA y MT del capítulo anterior:

1. *(Heurística) Un pixel no tiene ruido si y solo si su valor es igual a la de los pixeles contiguos.*
2. *(Observación) En la imagen de etiquetas  $\tilde{I}$ , los pixeles con etiqueta  $a$  son contiguos  $\mathbf{A}_{a,a}$  veces con pixeles con etiqueta  $a$ ,  $\mathbf{A}_{a,e}$  veces con pixeles con etiqueta  $e$ ,  $\mathbf{A}_{a,i}$  veces con pixeles con etiqueta  $i$ , etc.*

---

<sup>8</sup> Esta forma de razonamiento inferencial es conocido como *abductivo*, y se caracteriza precisamente por plantear conjeturas que surgen al observar los efectos de una regla.

3. Luego, la probabilidad de que la verdadera etiqueta de un pixel con valor  $a$  sea  $a$  es  $\mathbf{P}_{a,a}$ , que sea  $e$  es  $\mathbf{P}_{a,e}$ , que sea  $i$  es  $\mathbf{P}_{a,i}$ , etc.
4. (Observación) El pixel  $x$  tiene etiqueta  $a$ .
5. Luego, de 3 y 4, la probabilidad de que la verdadera etiqueta de  $x$  sea  $a$  es  $\mathbf{P}_{a,a}$ , que sea  $e$  es  $\mathbf{P}_{a,e}$ , que sea  $i$  es  $\mathbf{P}_{a,i}$ , etc.

De la heurística se desprende que en una imagen sin ruido los valores de dos pixeles contiguos *serán iguales*. Si nos referimos a una imagen de etiquetas, la MA es el conteo de las veces que cada par de etiquetas *son iguales* y contiguas, mientras que la MT es la *probabilidad de que una etiqueta sea igual a otra, en pixeles contiguos*. Tomemos por ejemplo el caso de una imagen con una sola fila como la representada en la siguiente figura y supongamos que nos interesa estimar el verdadero valor de un pixel etiquetado con  $a$ , como se aprecia en la siguiente figura.

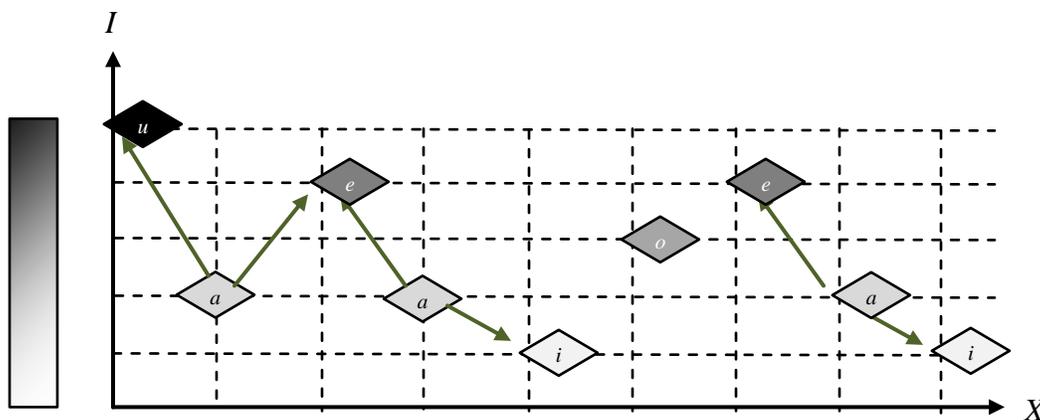


Fig. 2-1. Representación de una imagen 1-D, i.e. con una única fila o columna. Las flechas marcan las adyacencias de los pixeles etiquetados con  $a$  (ver párrafo anterior).

Podemos decir que con la correlación espacial se logra ‘ampliar’ el *histograma local*, pues para la estimación se tomarán en cuenta todos los vecinos de los pixeles del mismo color.

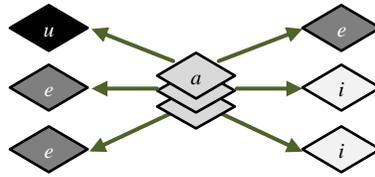


Fig. 2-2. Vecindad ampliada. Los pixeles con etiqueta  $a$  de la figura anterior, agrupados de tal forma que compartan su vecindad.

Así pues, tomando como referencia la figura anterior, podemos decir que ‘es mas *probable* que la etiqueta  $a$  sea *igual* a la etiqueta  $e$ , que a la etiqueta  $i$ ’, debido al número de vecindades contiguas que comparten. Tomaremos entonces esta ‘*probabilidad de que dos etiquetas sean iguales*’ como una *función de similitud* entre etiquetas contiguas, digamos:

$$g(a, e) := \mathbf{P}_{a,e} \quad (2.1)$$

Por ultimo, también podemos pensar en la MA como un ‘histograma’ categorizado por etiquetas. Lo siguiente sería ‘combinar’ este *histograma global* (la MA) con el *histograma local* buscando un balance entre la ‘información’ global y local. Algunas estrategias específicas se detallan en los siguientes capítulos. En el Apéndice A se muestran otras características de la correlación espacial que pueden servir de complemento a las ideas presentadas en este capítulo.

## Capítulo 3

### El filtro ‘4-conectado’

En el capítulo anterior se mencionaba que la correlación espacial podía ser utilizada para ‘ampliar’ el histograma local. Para ilustrar esto, en el presente capítulo se muestran los resultados de un filtro que estima el valor de cada pixel con base en los valores de los cuatro pixeles adyacentes, de manera que se esperaría que la escasa información local sea compensada al emplear los pesos calculados mediante la MA.

#### 3.1 Algoritmo

Con base en lo visto el capítulo anterior, definimos el siguiente algoritmo:

---

**Algoritmo 3-1. Filtrado de imágenes con vecindad 4-conectado.**

---

*Entrada:* imagen  $I$

*Salida:* imagen filtrada  $\hat{I}$

Calcular imagen de etiquetas  $\tilde{I}$  (ver sección 1.1)

Calcular matriz de adyacencia  $\mathbf{A}$  (Algoritmo 1-1)

Para cada  $p \in \Omega$ :

$$\hat{I}_p = \frac{\sum_{q \in V_p^4} \mathbf{A}_{a,e} I_q}{\sum_{q \in V_p^4} \mathbf{A}_{a,e}} \quad (3.1)$$

Donde,  $a = I_p, e = I_q$  y  $V_p^4$  son las posiciones correspondientes al pixel en  $p$  junto con su vecindad 4-conectado.

*Nota:* No existe diferencia si se intercambia  $\mathbf{A}$  por  $\mathbf{P}$  en la ecuación (3.1).

---

El algoritmo anterior bien puede aplicarse utilizando una ventana de cualquier tamaño, tema que abordaremos en el próximo capítulo. Por ahora, mostraremos el efecto de aplicar este filtrado con vecindad ‘4-conectado’ de forma iterativa.

La implementación se realizó en Matlab. Para generar la imagen de etiquetas correspondientes a las imágenes de color, se tomó la función `rgb2ind`<sup>9</sup> con la opción ‘`nodither`’, el cual implementa el algoritmo de *cuantización de Wu*<sup>10</sup> (Wu, 1991)(cuantización de mínima varianza). Se pudo comprobar visualmente que dicho algoritmo de cuantización se producen mejores resultados que con la cuantización uniforme. La cuantización de Wu divide el cubo de color RGB en ‘cajas’ (cada caja representa una etiqueta), asignando mas etiquetas en regiones del cubo (del histograma de color) donde existe una mayor concentración de colores. Para las pruebas se utilizó un máximo de 255 etiquetas.

## 3.2 Suavizado de imágenes

Aparte de los pixeles alterados por el ruido, los bordes de la imagen serán una forma de discontinuidad perceptual. Lo primero que se pudo observar al suavizar imágenes es que en general el filtro propuesto respeta los bordes después de varias iteraciones.

---

<sup>9</sup> Documentación de `rgb2ind` en Matlab: <http://www.mathworks.com/help/toolbox/images/f8-15484.html>

<sup>10</sup> Código disponible en la página web de Xiaolin Wu: <http://www.ece.mcmaster.ca/~xwu/cq.c>



Fig. 3-1. Aplicado iterativo del filtro '4-conectado'. De izquierda a derecha y de arriba abajo: imagen original e imágenes resultantes después de aplicar el filtro 3, 10 y 25 veces respectivamente.

Una explicación a esto es que la cantidad de píxeles en algún borde, por lo general será pequeña relativamente a la cantidad de píxeles en regiones suaves y que por lo tanto la correlación espacial entre colores diferentes será baja en relación a la correlación con colores similares.



Fig. 3-2. A la izquierda la imagen original, a la derecha filtrada después de 25 iteraciones. En general el filtro respeta los bordes.

Como consecuencia de lo anterior, las imágenes con superficies granulares o con texturas presentan un mayor emborronamiento como se muestra en las siguientes imágenes.



Fig. 3-3. Efecto del filtrado después de 10 iteraciones en imágenes con superficies granulares o texturas finas.

Una característica que pudo observarse, es que los colores poco auto-correlacionados en la imagen tienden a ‘desaparecer’.



Fig. 3-4. Efecto en colores poco auto-correlacionados. De izquierda a derecha: imagen original, imagen con ruido de alta frecuencia e imagen filtrada (3 iteraciones).

De esto dependerá también, que detalles finos en la imagen se vean afectados.



Fig. 3-5. Suavizado ligero. Los detalles finos en la imagen original (arriba izquierda) permanecen en la imagen filtrada después de 3, 10 y 25 iteraciones.

### 3.3 Eliminación de ruido

En una imagen con ruido los colores espurios no estarán correlacionados con algún otro color y por lo tanto tenderán a desaparecer. En la siguiente figura se muestran ejemplos de imágenes contaminadas artificialmente con ruido junto con su correspondiente imagen filtrada aplicando el filtro 4-conectado.



(a) AWGN<sup>11</sup>,  $\sigma_n = 100$



(b) AWGN,  $\sigma_n = 70$

---

Fig. 3-6. A la izquierda imagen contaminada con ruido, a la derecha imagen filtrada después de 3 iteraciones. Posteriormente se mejoraran estos resultados.

Algunas observaciones referentes al efecto que tiene el ruido de la imagen en la MA se presentan en el Apéndice B.

---

<sup>11</sup> Additive White Gaussian Noise

## Capítulo 4

# Adaptabilidad al ruido

Los *parámetros de configuración* permiten adecuar el filtro a las diversas características de la imagen y adaptarlo a diversas aplicaciones o propósitos. Un valor adecuado para estos parámetros puede inferirse sólo después de observar la imagen, por lo tanto, la efectividad del filtrado queda sujeta a la capacidad visual y experiencia de quien calibra la imagen. Desde luego, sería interesante tener un filtro *auto-calibrable*, es decir, que se ajuste sin supervisión a las condiciones de la imagen. Esto implicaría tener una hipótesis o un *modelo de como el observador percibe una imagen*, para que a partir de este modelo, sean derivados los valores adecuados de los parámetros. Con *modelo* podemos referirnos a supuestos heurísticos simples (como se discutió en el capítulo 2), así por ejemplo, con el filtro de media local asumimos que toda región local es percibida ‘plana’, i.e. de un solo color, de manera que solo es necesario calibrar el tamaño de la región local bajo la cual este supuesto es válido.

Con el presente capítulo se busca cumplir con los siguientes objetivos:

- Mejorar los resultados del filtro 4-conectado presentado en el capítulo anterior.
- Presentar un filtro auto-calibrado el cual tome una región local más amplia.
- Medir la efectividad del filtro resultante comparándolo contra el filtro bilateral.

### 4.1 El filtro bilateral como punto de referencia

El *filtro bilateral* (Tomasi & Manduchi, 1998) dadas la función de cercanía  $f$  y la función de similitud  $g$  es:

$$\hat{I}_p = \frac{\sum_{q \in \Omega} f(p, q) g(I_p, I_q) I_q}{\sum_{q \in \Omega} f(p, q) g(I_p, I_q)} \quad (4.1)$$

La forma usual de definir  $f$  y  $g$  es mediante Gaussianas:

$$g_{bil}(I_p, I_q) = e^{-\frac{\|I_p - I_q\|_2^2}{2\sigma_r^2}} \quad (4.2)$$

$$f_{bil}(p, q) = e^{-\frac{\|p - q\|_2^2}{2\sigma_d^2}} \quad (4.3)$$

donde los parámetros  $\sigma_r$  y  $\sigma_d$  controlan la apertura de la gaussiana en el rango y en el dominio respectivamente.

El par de parámetros (mas el tamaño de ventana) que controlan el comportamiento del filtro bilateral logran que este filtro sea muy flexible; sin embargo, de acuerdo a (Zhang & Gunturk, 2008) no existía a la fecha, un análisis teórico de como seleccionar de manera óptima los parámetros  $\sigma_d$  y  $\sigma_r$ . En el mismo trabajo citado se propone en base a pruebas realizadas, que una buena elección del valor de  $\sigma_d$  parece estar en el rango [1.5, 2.1], mientras que  $\sigma_r$  parece estar relacionado linealmente con  $\sigma_n$  (desviación estándar del ruido AWGN).

Para comparar nuestros resultados contra el filtro bilateral usaremos la misma configuración utilizada en (Zhang & Gunturk, 2008):  $\sigma_d = 1.8$ ,  $\sigma_r = 2 \times \sigma_n$  con tamaño de ventana de  $11 \times 11$ . Adicionalmente, se obtuvo la *configuración óptima* por nivel de ruido mediante la función `fminsearch` de Matlab. El valor óptimo obtenido para los parámetros  $\sigma_d$  y  $\sigma_r$  para un ancho de ventana de  $11 \times 11$  se muestra en la siguiente tabla.

$\sigma_n$	$\sigma_d$	$\sigma_r$
10	1.054225	0.109223
15	1.170315	0.169392
20	1.258260	0.232878
25	1.331346	0.302320
30	1.401339	0.376674
35	1.467100	0.460825
40	1.530953	0.561410
50	1.669907	0.822967
60	1.831355	1.159397
80	2.197954	2.002180
100	2.596337	3.022389

---

Tabla 4.1. Configuración óptima para el filtro bilateral (ver texto). Los valores  $\sigma_r$  se refieren a la intensidad de la imagen en el rango [0, 1].

Para las pruebas se usaron 11 imágenes en tonos de gris descargadas de (Foi). Con respecto a la cuantización de las imágenes en tonos de gris, se efectuó una cuantización uniforme usando 255 bins sobre la imagen con el *histograma ecualizado*<sup>12</sup>.

En la siguiente grafica se muestran los resultados del filtro bilateral bajo las dos configuraciones descritas junto con los resultados del filtro 4-conectado con 5 iteraciones.

---

<sup>12</sup> Con la ecualización del histograma en una imagen en tono de gris se busca distribuir los bins de manera similar que en la cuantización de mínima varianza empleada en imágenes a color.  
[http://en.wikipedia.org/wiki/Histogram\\_equalization](http://en.wikipedia.org/wiki/Histogram_equalization)

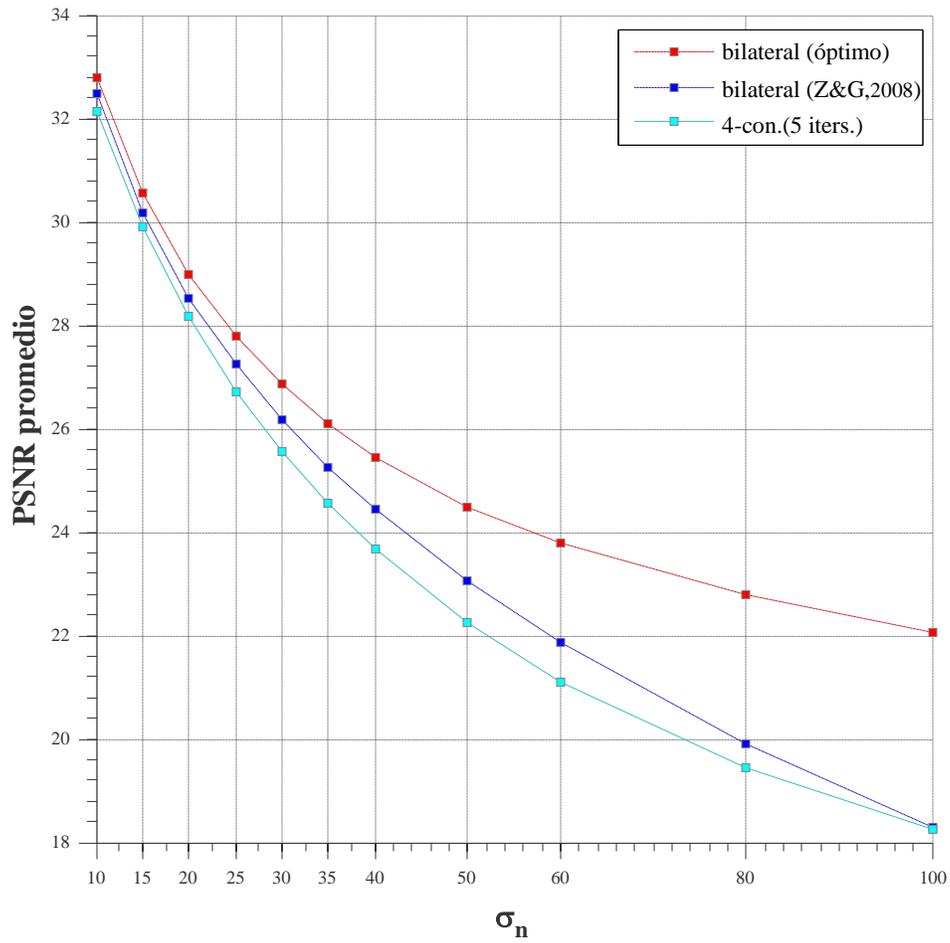


Fig. 4-1. Filtro 4-conectado vs filtro bilateral.

Puede notarse que el filtro bilateral configurado en base a (Zhang & Gunturk, 2008) arroja mejores resultados que el filtro 4-conectado definido en el capítulo anterior, aunque a niveles bajos de ruido ( $\sigma_n$  menor a 20) el PSNR de ambos es similar.

## 4.2 Similitud 'contextual'

Como se vio en la sección 1.3, una manera de visualizar las relaciones entre los colores de una imagen es a través de un grafo, donde cada etiqueta corresponde a un nodo y donde las aristas representan la *probabilidad de transición* de una etiqueta a otra, definidas por la MT. Nos gustaría entonces, medir la similitud entre dos etiquetas tomando en

cuenta su *contexto*, es decir, incluir la influencia de trayectorias indirectas que conecten a los correspondientes nodos en el grafo de etiquetas.

Con este propósito, considérese la matriz:

$$\mathbf{Q} = \left( \frac{\mathbf{P} + \mathbf{P}^T}{2} \right) \quad (4.4)$$

Cada entrada de la matriz  $\mathbf{Q}^t$  es entonces, la probabilidad de trasladarse en  $t$  pasos entre los dos nodos, de ida y vuelta. La siguiente grafica es igual a la de la Fig. 4-1 pero incluye los resultados para el filtro 4-conectado utilizando  $\mathbf{Q}^2$  como función de similitud.

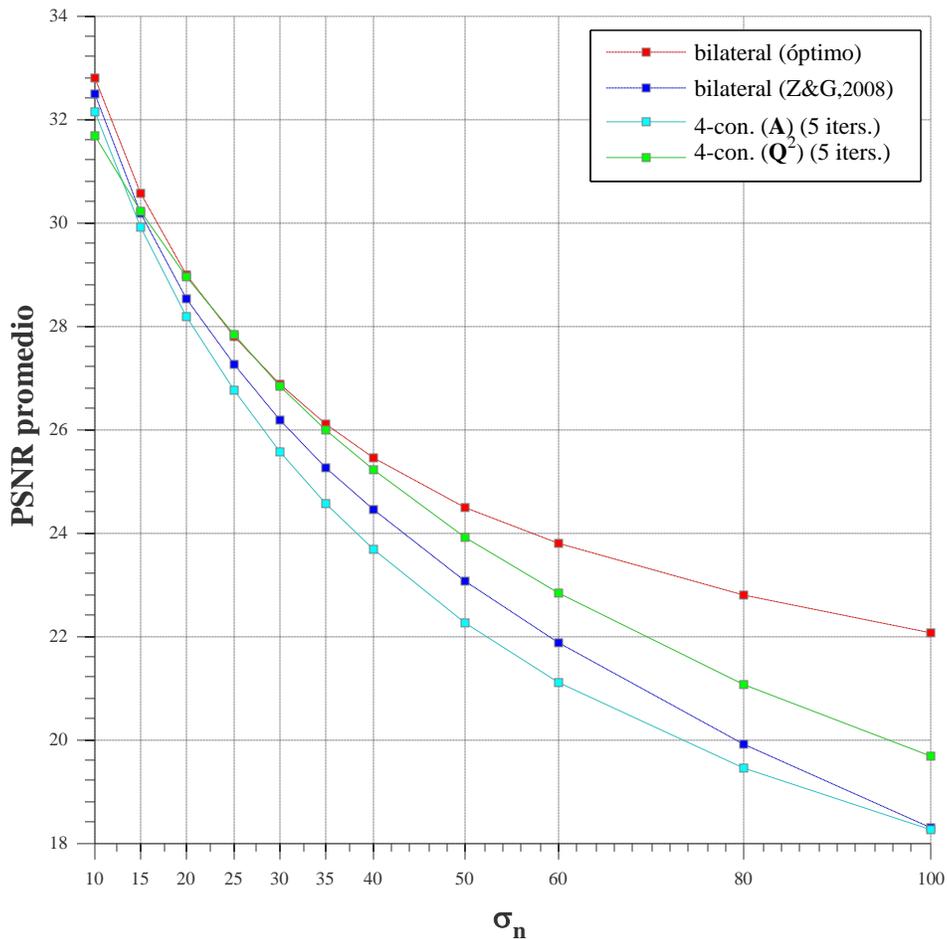


Fig. 4-2. Comparación de *funciones de similitud* ( $\mathbf{A}$  vs.  $\mathbf{Q}^2$ ) utilizando el filtro 4-conectado.

Como se puede apreciar en la gráfica anterior, con  $Q^2$  se mejoran significativamente los resultados del filtro 4-conectado.

### 4.3 Cercanía y tamaño de la vecindad

Dado un pixel en una posición de la imagen, la *función de cercanía* debe ‘ajustarse’ a la región de pixeles similares al pixel dado en la imagen original (sin ruido), es decir, a diferencia de la función de similitud, la función de cercanía no debería verse afectada por la magnitud del ruido presente en la imagen.

Sin embargo, en general se tendrá que a mayor nivel de ruido será necesaria una cantidad mayor de valores de muestra, i.e. una región local más amplia, para así poder obtener una estimación más acertada del verdadero valor del pixel actual. Por lo tanto, podemos decir que una buena definición de función de cercanía se debe considerar un equilibrio entre dos criterios opuestos: incluir un número suficiente de pixeles (muestras), pero solo los mas cercanos, de manera que solo se tomen en cuenta los más similares.

Por lo tanto, podemos conjeturar que dada una función de similitud  $g$ , existe una función de cercanía  $f$  que optimiza un *criterio de selección de región local* de manera que el filtro repare lo mejor posible una imagen. No está claro sin embargo, hasta qué punto pueden ser útiles las herramientas con las que contamos para poder encontrar dicha  $f$  óptima. De momento, solo se mostrara un indicador que parece ser útil para calibrar  $f$ , esperando que este indicador capte en esencia los criterios para definir  $f$  de buena manera.

Sea pues,  $\theta$  el vector correspondiente a la diagonal de la matriz  $\mathbf{P}$ .

$$\theta = \text{diag}(\mathbf{P}) \quad (4.5)$$

Entonces,  $\theta_a$  es la probabilidad de transitar a un pixel con etiqueta  $a$  a otro pixel con la misma etiqueta. Sea  $S_{\sim a}$  una variable aleatoria que representa el número de veces que se transitaría a una etiqueta diferente de  $a$  en  $n$  intentos, i.e.:

$$S_{\sim a} \sim \text{Binomial}(n, 1 - \theta_a) \quad (4.6)$$

Un indicador del *tamaño de la vecindad* de la etiqueta  $a$ , puede ser el número de transiciones esperadas a etiquetas distintas a la actual. Tomando esta cantidad normalizada quedaría:

$$\frac{E[S_{\sim a}]}{\sqrt{\text{Var}[S_{\sim a}]}} = \frac{n(1 - \theta_a)}{\sqrt{n\theta_a(1 - \theta_a)}} = \sqrt{\frac{n(1 - \theta_a)}{\theta_a}} \quad (4.7)$$

Para tener un solo indicador para todas las etiquetas, usaremos el promedio de los valores de  $\theta$  pesados con el histograma de la imagen de etiquetas  $\mathbf{h}$ :

$$\theta = \frac{\sum_{i \in \Lambda} \mathbf{h}_i \theta_i}{\sum_{i \in \Lambda} \mathbf{h}_i} \quad (4.8)$$

De esta forma, para  $f$  definida mediante una Gaussiana como en (4.3), una estimación para  $\sigma_d$  puede obtenerse con:

$$\hat{\sigma}_d = c \sqrt{\frac{(1 - \theta)}{\theta}} \quad (4.9)$$

Para una constante  $c$ . En la siguiente tabla se muestra el valor del indicador de tamaño de vecindad para algunas de las imágenes de prueba a distintos niveles de ruido.

Tamaño de la vecindad: $\sqrt{\frac{(1-\theta)}{\theta}}$								
$\sigma_n$	<i>Camera man</i> 256 × 256	<i>Lena</i> 512 × 512	<i>Barbara</i> 512 × 512	<i>boat</i> 512 × 512	<i>couple</i> 512 × 512	<i>fingerprint</i> 512 × 512	<i>hill</i> 512 × 512	<i>house</i> 256×256
10	5.33	5.76	6.00	5.24	5.34	6.41	5.43	5.42
15	6.07	6.58	6.62	5.98	6.14	6.82	6.13	6.38
20	6.86	7.14	7.05	6.52	6.75	7.25	6.69	7.14
25	7.32	7.55	7.44	6.96	7.20	7.60	7.07	7.77
30	7.81	7.91	7.73	7.32	7.64	7.84	7.43	8.23
35	8.10	8.15	7.96	7.63	7.99	8.18	7.73	8.41
40	8.36	8.32	8.11	7.79	8.20	8.38	7.98	8.54
50	8.70	8.73	8.48	8.15	8.63	8.60	8.34	9.04
60	8.76	9.00	8.70	8.49	8.88	8.78	8.67	9.37
80	9.33	9.32	9.02	8.92	9.35	9.09	9.08	9.72
100	9.32	9.53	9.28	9.12	9.54	9.20	9.28	9.68

Tabla 4.2

Con esto, ‘actualizamos’ el algoritmo 4-conectado proporcionado en el capítulo anterior para usar  $\mathbf{Q}^2$  como función de similitud y  $f$  calibrada mediante el indicador de tamaño de vecindad, e identificaremos a este filtro como *filtro por correlación espacial (fce)*. La diferencia primordial con el algoritmo 4-conectado es que el *fce* no es un filtro iterativo.

---

**Algoritmo 4-1. Filtrado de imágenes por correlación espacial.**

---

*Entrada:* imagen  $I$ , longitud de mitad de lado  $r$ .

*Salida:* imagen filtrada  $\hat{I}$

Calcular la imagen de etiquetas  $\tilde{I}$  (ver sección 1.1)

Calcular la matriz de adyacencia  $\mathbf{A}$  (Algoritmo 1-1)

Calcular la matriz de transición  $\mathbf{P}$  (1.7)

$$\mathbf{Q} = \left( \frac{\mathbf{P} + \mathbf{P}^T}{2} \right)$$

Para cada  $p \in \Omega$ :

$$\hat{I}_p = \frac{\sum_{q \in W_p^r} \mathbf{Q}_{a,e}^2 f(p,q) I_q}{\sum_{q \in W_p^r} \mathbf{Q}_{a,e}^2 f(p,q)} \quad (4.10)$$

Con  $f(p,q) = e^{-\frac{\|p-q\|_2^2}{2\sigma_d^2}}$ ,  $\hat{\sigma}_d = c \sqrt{\frac{(1-\theta)}{\theta}}$  para alguna constante  $c$ .

Y donde,  $a = I_p, e = I_q$  y  $W_p^r$  son las posiciones de una ventana con centro en  $p$  y con longitud de mitad de lado  $r$ .

---

La siguiente grafica muestra los resultados usando  $c = 1/5$ .

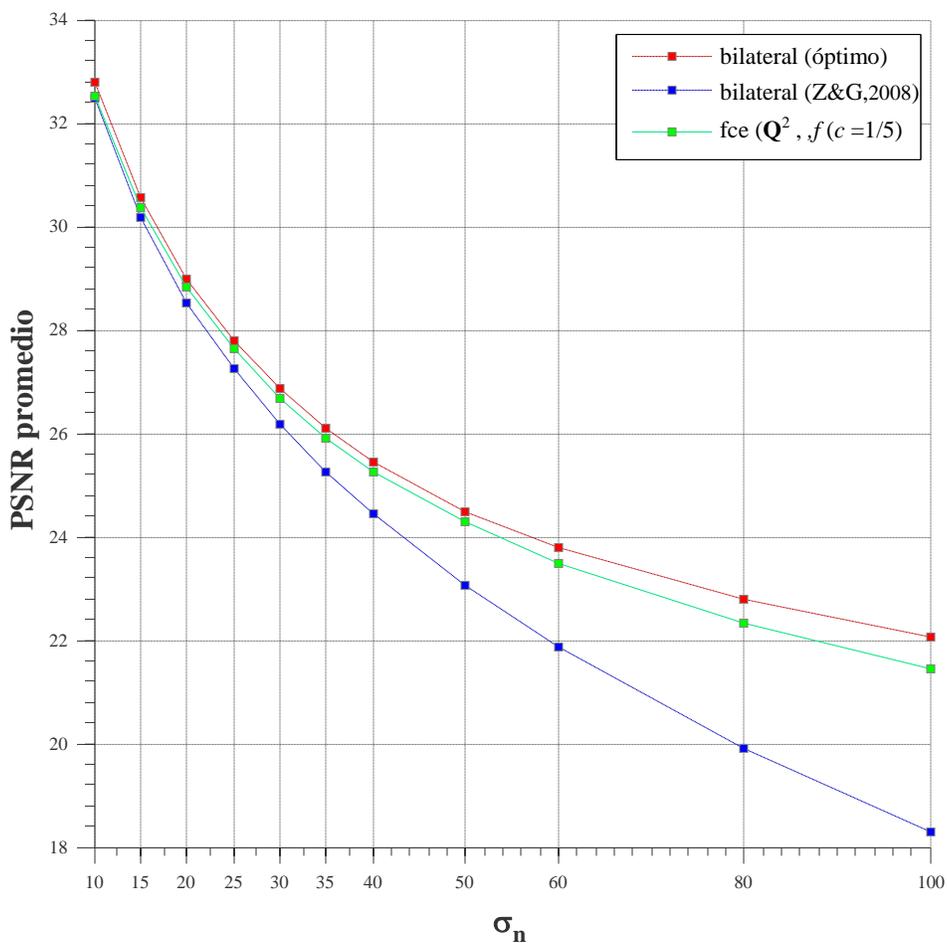
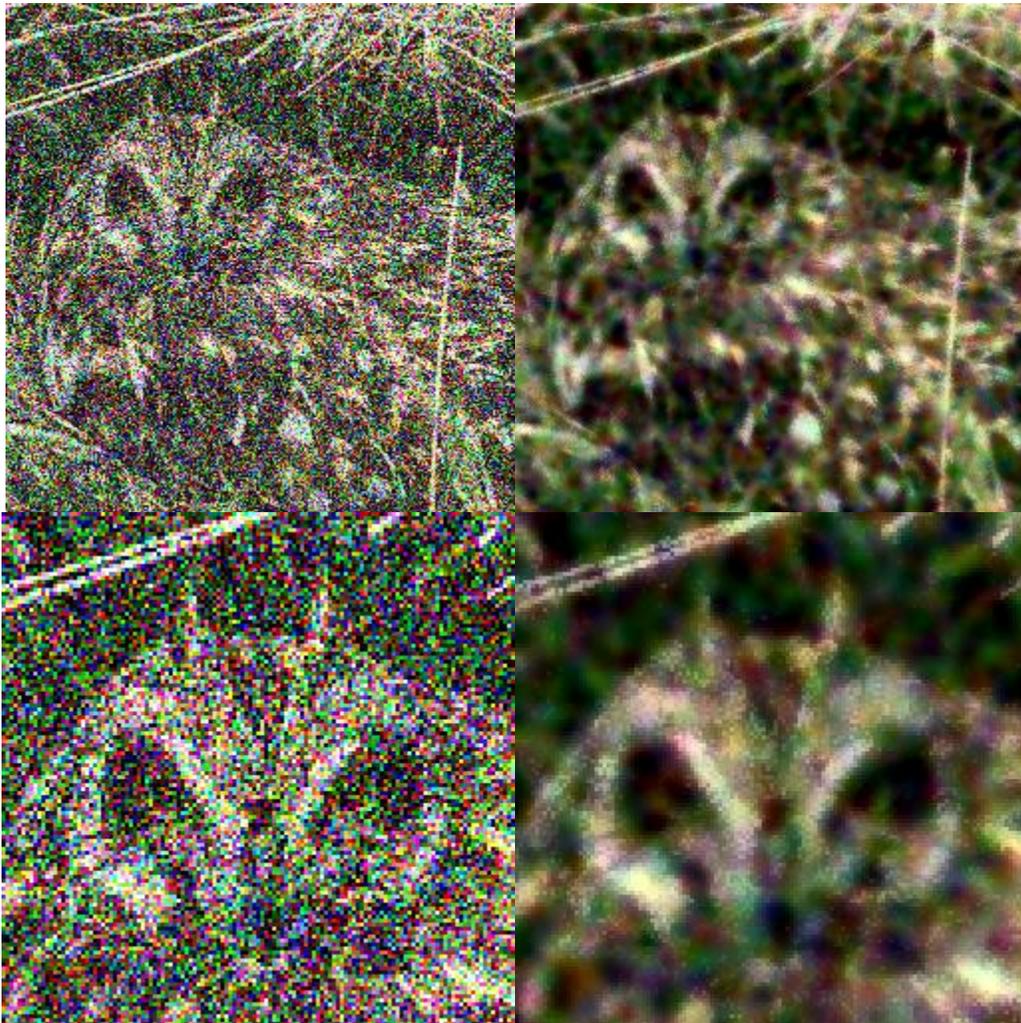


Fig. 4-3. Filtro por correlación espacial (fce) contra el filtro bilateral.

Considerando que para configurar el filtro bilateral en estos experimentos fue necesario conocer el valor de  $\sigma_n$ , la ventaja práctica del *fce* sobre el filtro bilateral es evidente. Lo siguiente es comprobar que el mismo filtro funciona bien en imágenes a color, lo cual se validó visualmente en varias imágenes.



(a)



(b)

Fig. 4-4. Arriba izquierda: imagen contaminada con AWGN, a)  $\sigma_n = 100$ , b)  $\sigma_n = 70$ ; arriba derecha: imagen filtrada con el fce ( $\mathbf{Q}^2, f(c=1/5)$ ), tamaño de ventana  $11 \times 11$ . Abajo, acercamientos de las correspondientes imágenes de arriba.

Para el caso de suavizado de imágenes, podría ser necesario ajustar los parámetros  $c$  y  $t$  de las funciones  $f$  y  $g_t := \mathbf{Q}^t$  respectivamente para obtener los resultados deseados.



---

Fig. 4-5. Arriba izquierda: imagen original; arriba derecha: imagen filtrada con el  $f_{ce}(\mathbf{Q}^2, f(\mathbf{c}=\mathbf{1}))$  y tamaño de ventana de  $11 \times 11$ . Abajo, acercamientos de las correspondientes imágenes superiores.



---

Fig. 4-6. De arriba a abajo y de izquierda a derecha: imagen original, e imágenes filtradas con el  $fce(\mathbf{Q}^t, f(c=1))$  con  $t = 4, 8, 16$  (en ese orden) y tamaño de ventana de  $11 \times 11$ .

En las figuras anteriores se muestra el efecto de modificar los parámetros  $c$  y  $t$  de las funciones de cercanía y similitud respectivamente.

Si bien el fce no es la solución definitiva para tener un filtro local auto-calibrado, si se ha mostrado durante este capítulo que la medida de correlación espacial es una herramienta valiosa en este sentido.

## Capítulo 5

# Comentarios finales y conclusiones

### 5.1 Resumen de resultados

En esta tesis se presentó:

- Una medida de correlación espacial. De esta se deriva un grafo y cadena de Markov y su equivalente caminata aleatoria sobre la imagen.
- Experimentos realizados con un filtro con vecindad ‘4-conectado’.
- Un filtro calibrado mediante la correlación espacial.
- Resultados contra el filtro bilateral, donde pudieron constatarse ventajas sobre este.

### 5.2 Conclusiones

Desde un punto de vista práctico, la aportación de esta tesis es un filtro de imágenes y un método para su autocalibración. La eficacia de la técnica, constituye por sí misma un fenómeno a estudiar y tal fue la principal motivación durante el desarrollo de esta tesis. Será necesaria, por lo tanto, una comprensión más profunda al respecto.

## Apéndice A

### Correlación espacial y el color.

El proceso de colocar pixeles con colores diferentes uno junto a otro crea la ilusión de colores adicionales, como puede apreciarse en la siguiente figura.

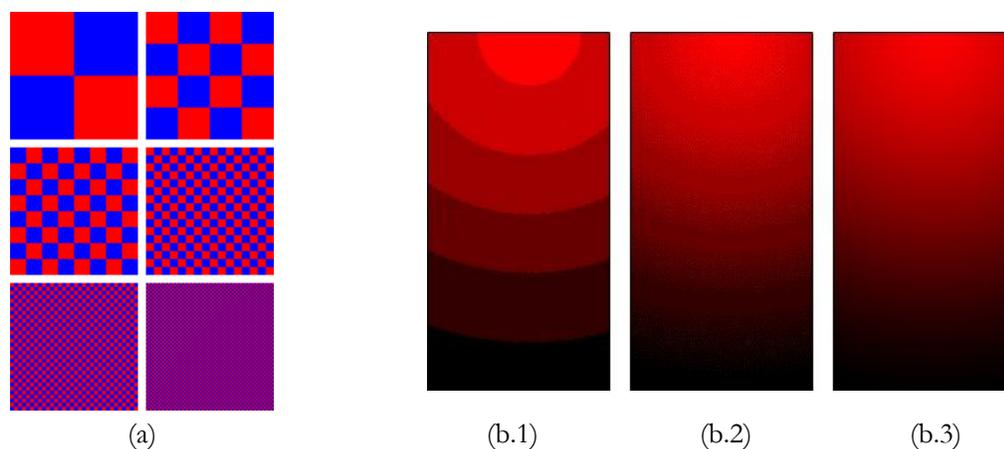


Fig. A-1. En la imagen (a) el patrón de colores rojo y azul se percibe cada vez más uniformemente como un tono morado. Este efecto es aprovechado en la técnica conocida como *dithering*, (b.1) gradiente de 8 bits, (b.2) gradiente de 8 bits después del *dithering*, (b.3) gradiente en 24 bits como comparación.

Al observar estos ejemplos y el de otras imágenes que hagan uso de la técnica de *dithering*, podemos conjeturar que este fenómeno puede medirse mediante la correlación espacial: entre más correlacionadas sean dos etiquetas, los colores correspondientes se apreciarían como la ‘combinación’ de ambos. Supongamos ahora que observamos una imagen que se aleja gradualmente de manera que cada vez es más apreciable el efecto óptico del *dithering*. Podríamos definir entonces una función de energía en función de las diferencias entre cada punto de la imagen y sus vecinos, de manera que la mayor energía corresponde a la situación en que la imagen puede apreciarse nítidamente y disminuye

conforme los detalles dejan de apreciarse al observar la imagen a una mayor distancia. Este proceso de *difusión* ocurre también al aplicar un filtro de *pasa-bajas* a la imagen, de manera que nos dedicaremos a describir el filtro propuesto en este contexto. Con esta motivación, expresamos la función de energía en el tiempo continuo  $t$  para una imagen  $I$ , tomando el caso de imágenes en tono de gris:

$$E_I(t) = \frac{1}{2\kappa} \sum_p \sum_{q \sim p} (I_p(t) - I_q(t))^2 + \frac{\tau^2}{2} \sum_p \left( \frac{\partial I_p(t)}{\partial t} \right)^2 \quad (\text{A.1})$$

Donde  $\kappa$  es el número de pixeles adyacentes a cada pixel y  $\tau$  es la unidad de desplazamiento temporal:

$$\tau \frac{\partial I_p(t)}{\partial t} \approx I_p(t) - I_p(t - \tau)$$

De manera que el segundo término de (A.1) es el error cuadrático con respecto al momento anterior. Así pues, al suavizar una imagen nos gustaría ‘aplanar’ las intensidades de la imagen, pero penalizando los ‘cambios bruscos’ de intensidad que ocurren en los pixeles en los *bordes* de las figuras en la imagen. Como se puede ver, al minimizar la función de energía se busca cumplir con estos criterios. Con el propósito de analizar la correlación espacial en el proceso de difusión, escribiremos (A.1) en términos de la MA. Sea pues  $y_a(t)$  un valor de intensidad que caracteriza a todos los pixeles con etiqueta  $a$  en el tiempo  $t$ . Para abreviar, se omitirá el parámetro de tiempo  $t$  cuando no sea necesario especificarse (i.e.  $y_a \doteq y_a(t)$ , o bien el vector  $y \doteq y(t)$ ). Podemos aproximar la función de energía en la imagen de etiquetas, rescribiéndola en términos de  $y$ :

$$E_y(t) = \frac{1}{2\kappa} \sum_p \sum_{q \sim p} (y_{\tilde{I}_p} - y_{\tilde{I}_q})^2 + \frac{\tau^2}{2} \sum_p \left( \frac{\partial y_{\tilde{I}_p}}{\partial t} \right)^2 \quad (\text{A.2})$$

Equivalentemente,

$$E_y(t) = \frac{1}{2\kappa} \sum_a \sum_e \mathbf{A}_{a,e} (y_a - y_e)^2 + \frac{\tau^2}{2} \sum_a \hat{h}_a \left( \frac{\partial y_a}{\partial t} \right)^2 \quad (\text{A.3})$$

Desarrollando el primer término por separado:

$$\begin{aligned}
\sum_a \sum_e \mathbf{A}_{a,e} (y_a - y_e)^2 &= \\
&= \sum_a y_a^2 \left( \sum_e \mathbf{A}_{a,e} \right) - 2 \sum_a y_a \left( \sum_e \mathbf{A}_{a,e} y_e \right) + \sum_e y_e^2 \left( \sum_a \mathbf{A}_{a,e} \right) \\
&= \sum_a \kappa \hat{h}_a y_a^2 - 2 \sum_a y_a [\mathbf{A}y]_a + \sum_e \kappa \hat{h}_e y_e^2 \\
&= 2 \sum_a \kappa \hat{h}_a y_a^2 - 2 \sum_a y_a [\mathbf{A}y]_a \\
&= 2y^T \mathbf{D}y - 2y^T \mathbf{A}y \\
&= 2y^T (\mathbf{D} - \mathbf{A})y
\end{aligned}$$

Donde  $\mathbf{L} = \mathbf{D} - \mathbf{A}$  es también la *matriz Laplaciana* del grafo de etiquetas asociado a la imagen. Igualmente es posible escribir el segundo término de  $E_y(t)$  en notación matricial, de tal manera que (A.3) queda:

$$E_y(t) = \frac{1}{\kappa} y^T \mathbf{L}y + \frac{\tau^2}{2\kappa} \dot{y}^T \mathbf{D}\dot{y} \quad (\text{A.4})$$

Buscamos el nivel de difusión óptimo en un instante  $t$ , así que derivando e igualando a cero:

$$\begin{aligned}
\frac{\partial E_y(t)}{\partial t} &= \frac{2}{\kappa} \dot{y}^T \mathbf{L}y + \frac{\tau^2}{\kappa} \dot{y}^T \mathbf{D}\dot{y} = 0 \\
\frac{1}{\kappa} \dot{y}^T (2\mathbf{L}y + \tau^2 \mathbf{D}\dot{y}) &= 0
\end{aligned}$$

Si  $\dot{y} \neq 0$ , la condición para obtener la mínima energía es:

$$2\mathbf{L}y + \tau^2 \mathbf{D}\dot{y} = 0 \quad (\text{A.5})$$

No hace falta resolver esta ecuación diferencial para darle una interpretación útil. Multiplicando por  $\mathbf{D}^{-1}$  y manipulando:

$$2(\mathbf{I} - \mathbf{P})y + \tau^2 \ddot{y} = 0$$

$$y + \frac{\tau^2}{2} \ddot{y} = \mathbf{P}y \quad (\text{A.6})$$

La parte izquierda de la igualdad, se asemeja a la aproximación por el teorema de Taylor  $y(t + \tau) = y(t) + \tau \dot{y}(t) + \frac{\tau^2}{2} \ddot{y}(t)$ . Completando con el término faltante, llegamos a la forma recursión:

$$y(t + \tau) = \mathbf{P}y(t) + \tau \dot{y}(t) \quad (\text{A.7})$$

Donde podemos resaltar que:

$$y(t + \tau) \approx \mathbf{P}y(t) \quad (\text{A.8})$$

es una buena aproximación siempre que el término  $\tau \dot{y}(t)$  sea lo suficientemente pequeño. Esto significa que el proceso de Markov mediante la matriz estocástica  $\mathbf{P}$  optimiza la disminución de energía para el proceso de difusión en el grafo de etiquetas para una  $\tau$  lo suficientemente chica.

Retomando la ecuación diferencial (A.5) la cual corresponde a la del *movimiento armónico simple* como el que se presenta por ejemplo, en un *sistema de resortes* y cuya solución general es de la forma:

$$y(t) = v \cos(\omega_0 t + \phi)$$

Donde  $v$  es la amplitud de la oscilación correspondiente a la frecuencia angular  $\omega_0$ .

Eligiendo  $\tau^2 = 2$  y sustituyendo la solución general en (A.5) nos queda:

$$\mathbf{L}v = \omega_0^2 \mathbf{D}v \quad (\text{A.9})$$

Una discusión amplia sobre las soluciones de la ecuación anterior se puede encontrar en (Shi & Malik, 1997), donde se muestra que la solución de frecuencia mínima  $\omega_0$  es el segundo eigenvector de (A.9), el cual satisface:

$$v_1 = \arg \cdot \min \frac{v^T \mathbf{L} v}{v^T \mathbf{D} v} \quad (\text{A.10})$$

$$\text{s. a } v_1^T \mathbf{D} \mathbf{1} = 0$$

Además, en el mínimo:

$$\frac{v_1^T \mathbf{L} v_1}{v_1^T \mathbf{D} v_1} = \lambda_1 \quad (\text{A.11})$$

Y por lo tanto  $\lambda_1 = \omega_0^2$  es el eigenvalor en (A.9). Dado que  $\omega_0$  es la mínima frecuencia  $y(t) \approx v_1$  deberá ser una aproximación razonable, así que  $v_1$  es (aproximadamente) un mínimo con respecto a la función de energía (A.4) y es también un mínimo con respecto al cociente en (A.10). Podemos decir entonces que  $v_1$  son las intensidades que mejor se adaptan a la imagen respetando los bordes, y que son obtenidas con el mínimo esfuerzo.

En las siguientes imagenes se muestra el experimento de asignar las intensidades del vector  $v_1$  a una imagen de etiquetas. El código en Matlab se anexa al final de este documento. En la practica se pudo notar que se obtienen mejores resultados ecualizando la imagen previamente a la cuantización, para posteriormente generar la imagen de etiquetas a partir de la imagen ecualizada.



Fig. 5-1. De izquierda a derecha y de arriba a abajo: imagen original, imagen de etiquetas con la intensidad establecida a los primeros tres eigenvectores del eigensistema (A.9). Puede observarse que el segundo eigenvector reproduce apropiadamente (salvo la escala) los tonos de gris de la imagen original. Nótese también que en la imagen correspondiente al tercer eigenvector se distinguen solo cambios bruscos de color (de mayor frecuencia). En contraste, los tonos del primer eigenvector aparecen como 'ruido' en la imagen. Teóricamente este último caso corresponderían a observar variaciones de muy baja frecuencia (frecuencia 0).



Fig. 5-2. Otros ejemplos mostrando la imagen original y la de intensidad dada por el segundo eigenvector en los tres primeros ejemplos, y tercer eigenvector en el último. Casos patológicos como este último pueden darse debido a histogramas difíciles de equalizar.

De acuerdo con (Shi & Malik, 1997), las intensidades asociadas a cada eigenvector pueden entenderse como coordenadas a cada nodo del grafo de etiquetas, de manera que inducen las intensidades inducen *clusters*, aunque a diferencia de la formalización de la *NCut*, en el grafo de etiquetas esto sucede en el dominio real.

Es posible realizar pruebas similares con imágenes a color, tomando tres eigenvectores (correspondientes a tres canales), sin embargo, a diferencia de las imágenes en tonos de gris las cuales tienen solo dos acomodos posibles ('blanco a negro' y 'negro a blanco') la rotación adecuada en el espacio de color para imágenes a color es desconocida; salvo esta contrariedad la idea es la misma.



---

Fig. 5-3. Imagen en falso color que se obtiene al asignar los tres eigenvectores con eigenvalor mas bajo de la matriz  $\mathbf{L}_n = \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1/2}$  a los canales 'R', 'G', 'B'.

## Apéndice B

### Ruido y correlación espacial.

Sea  $Y$  una imagen sin ruido, la cuales es alterada mediante una señal de ruido aleatorio, dando como resultado a la imagen con ruido  $I$ . Suponiendo que la distribución del ruido no depende de la posición en la imagen, denotaremos la probabilidad de que un pixel se altere del color ‘ $o$ ’ al color ‘ $a$ ’ como:

$$N_{o,a} := P(\tilde{I}_p = a \mid \tilde{Y}_p = o) \quad (\text{B.1})$$

Ahora, podríamos estimar la MA de la imagen con ruido a partir de la MA de la imagen original:

$$E[\mathbf{A}_{a,e}^{(I)}] = \sum_{o \in \Lambda} \sum_{u \in \Lambda} N_{o,a} \mathbf{A}_{o,u}^{(Y)} N_{u,e} \quad (\text{B.2})$$

Donde  $\mathbf{A}^{(I)}$  y  $\mathbf{A}^{(Y)}$  son las MA de las imágenes  $I$  e  $Y$  respectivamente. Esto puede escribirse en forma compacta usando notación matricial. Reacomodando:

$$E[\mathbf{A}_{a,e}^{(I)}] = \sum_{o \in \Lambda} \left[ N_{o,a} \sum_{u \in \Lambda} \mathbf{A}_{o,u}^{(Y)} N_{u,e} \right] = \sum_{o \in \Lambda} \left[ N_{a,o}^T \left[ \mathbf{A}^{(Y)} \mathbf{N} \right]_{o,e} \right]$$

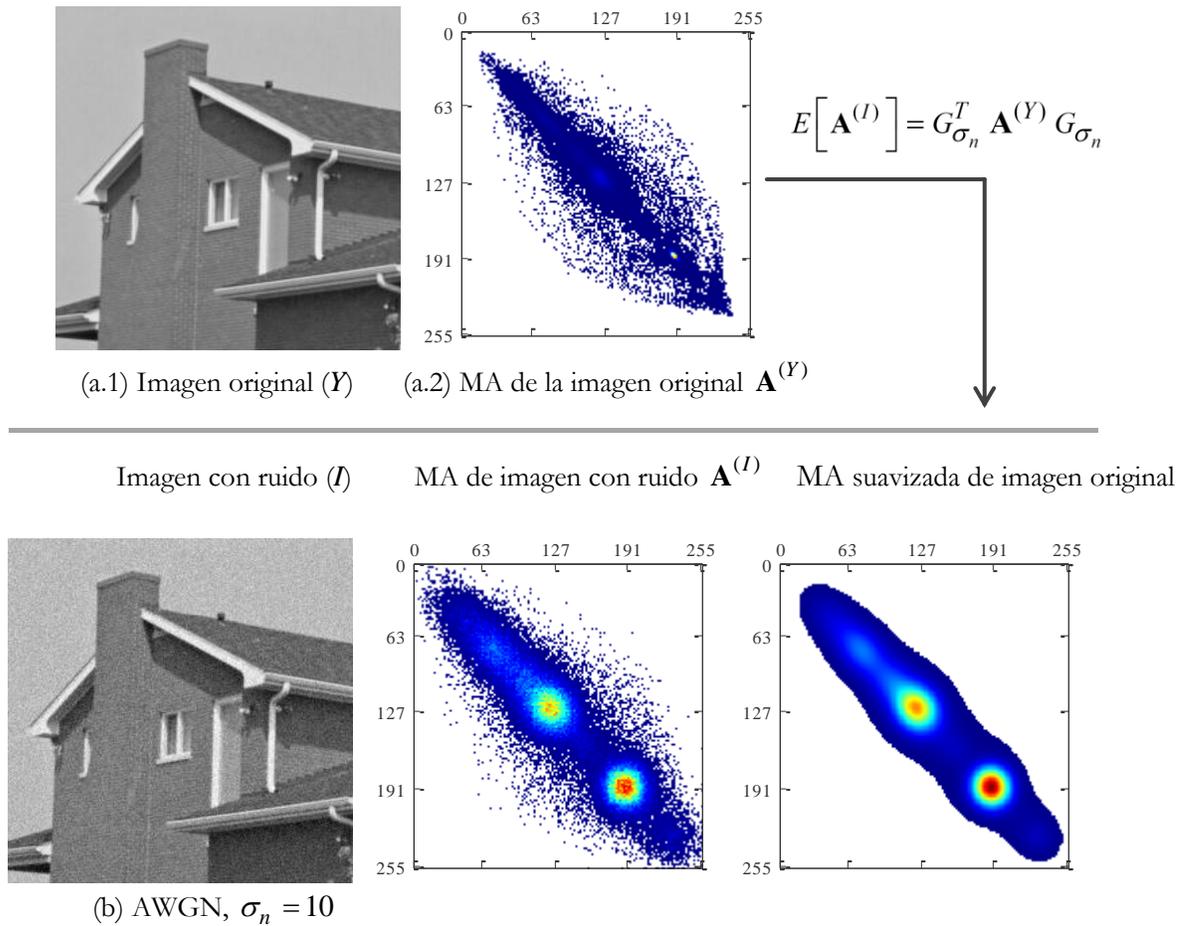
Esto es,

$$E[\mathbf{A}^{(I)}] = \mathbf{N}^T \mathbf{A}^{(Y)} \mathbf{N} \quad (\text{B.3})$$

Si el ruido es Gaussiano, lo anterior sería análogo a aplicar un filtro Gaussiano a la MA de la imagen original  $Y$  (por filas y por columnas por separado) con la misma

desviación estándar  $\sigma_n$  del ruido. Esto se puede validar en imágenes en tonos de gris, estableciendo la imagen de etiquetas a la imagen misma y comparando la MA de la imagen con ruido con la MA original suavizada con un filtro Gaussiano ajustado con  $\sigma_n$ . En la siguiente figura se muestra un ejemplo de dicho experimento, para el cual definimos la matriz  $G_{\sigma_n}$  cuyas entradas son de la forma:

$$G_{u,v|\sigma_n} = \frac{1}{\sigma_n \sqrt{2\pi}} e^{-\frac{(u-v)^2}{2\sigma_n^2}} \quad (\text{B.4})$$



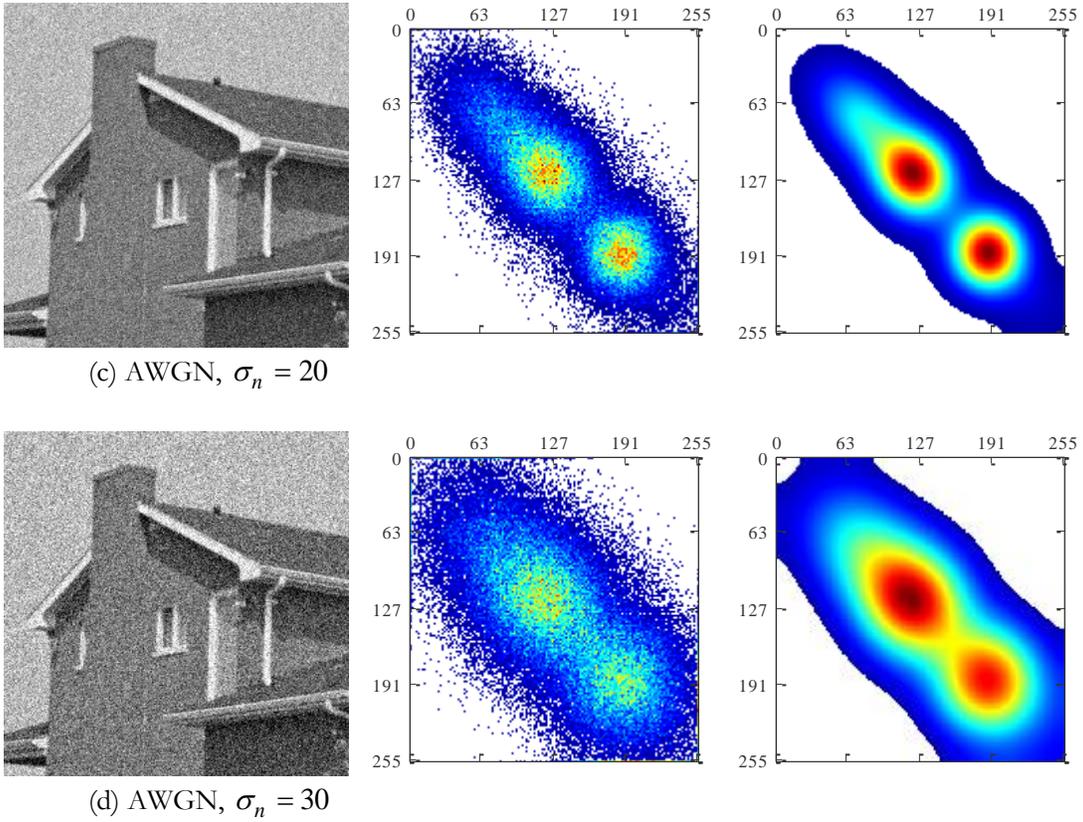


Fig. 5-4. Para cada inciso (b-d) izquierda: imagen con ruido gaussiano, centro: MA correspondiente a la imagen de la izquierda, y derecha: MA de la imagen original ‘suavizada’ con  $G_{\sigma_n}$  donde  $\sigma_n$  toma el mismo valor que para la señal de ruido de la imagen izquierda.

Ahora bien, de forma similar a (B.1) podemos expresar la probabilidad de que un pixel con color ‘ $a$ ’ en la imagen con ruido, fuera originalmente del color ‘ $o$ ’, lo cual denotaremos por:

$$T_{a,o} := P(\tilde{Y}_p = o \mid \tilde{I}_p = a) \quad (\text{B.5})$$

Las probabilidades de las matrices N y T están relacionadas por el teorema de Bayes:

$$P(\tilde{Y}_p = o \mid \tilde{I}_p = a) = \frac{P(\tilde{I}_p = a \mid \tilde{Y}_p = o)P(\tilde{Y}_p = o)}{P(\tilde{I}_p = a)}$$

Es decir,

$$T_{a,o} = \frac{N_{o,a} h_Y(o)}{h_I(a)} \quad (\text{B.6})$$

donde  $h_I$  y  $h_Y$  son los histogramas de las imágenes  $\tilde{I}$  e  $\tilde{Y}$  respectivamente.

Con esto podemos expresar la MT de la imagen con ruido en función de la MT de la imagen original. Manipulando la ec. (B.2):

$$\frac{E[\mathbf{A}_{a,e}^{(I)}]}{h_I(a)} = \sum_{o \in \Lambda} \sum_{u \in \Lambda} \left( \frac{N_{o,a} h_Y(o)}{h_I(a)} \right) \left( \frac{\mathbf{A}_{o,u}^{(Y)}}{h_Y(o)} \right) N_{u,e}$$

$$E[\mathbf{P}_{a,e}^{(I)}] = \sum_{o \in \Lambda} \sum_{u \in \Lambda} T_{a,o} \mathbf{P}_{o,u}^{(Y)} N_{u,e}$$

Luego,

$$E[\mathbf{P}^{(I)}] = \mathbf{T} \mathbf{P}^{(Y)} \mathbf{N} \quad (\text{B.7})$$

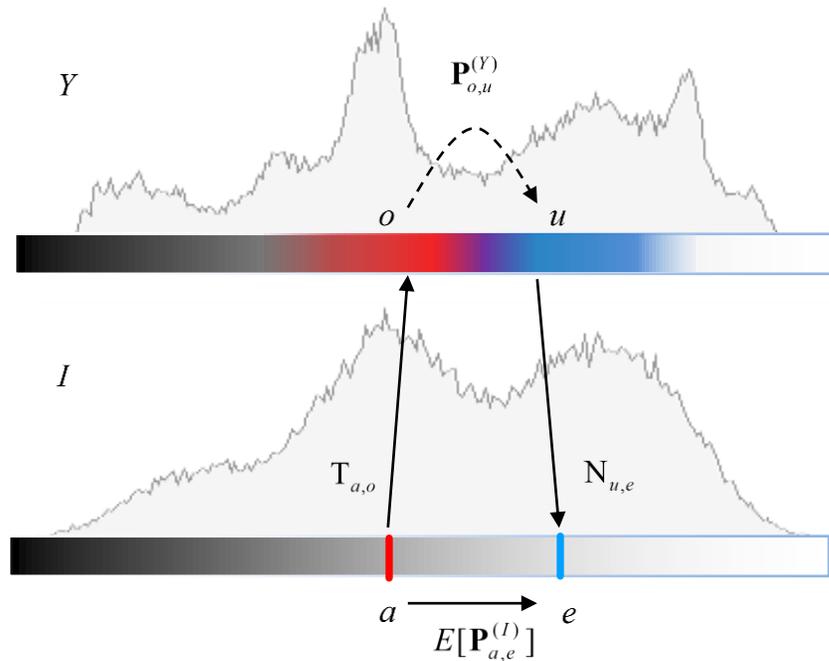


Fig. 5-5. Interpretación de la MT en una imagen con ruido. El histograma arriba es el de la imagen original (Y) y el de abajo corresponde al de la imagen con ruido. La probabilidad de transitar de una etiqueta a otra en la imagen con ruido, se relaciona con las probabilidades de transición en la imagen original.

# Anexo I

## Código fuente (Matlab)

### Filtro 4-conectado (scor4filt).

```
function Iout=scor4filt(I, iters)
% SPATIAL CORRELATION FILTER (4-connectivity)
%
% IN
% I:      Image to filter, (RGB color or grayscale)
% iters:  iterations (integer)
% %
% OUT
% Iout:   Filtered image (double)
%
% EXAMPLE:
%
% %read image
% imfile = fullfile(matlabroot,...
%     'toolbox','images','imdemos','greens.jpg');
% Io=im2double(imread(imfile));
% %add noise
% sigma_n=20; %noise std. dev.
% randn('seed', 0);
% I=Io+sigma_n/255*randn(size(Io));
% figure('Name', 'input'), imshow(I);
% %filter image
% Iout = scor4filt(I,5);
% figure('Name', 'output'), imshow(Iout);
%
% Author:   Carlos Estrada
% Date:     May 25, 2012
% Version   1.0
%

I=im2double(I);
[m n ch]=size(I);

for iter=1:iters

if ch==1           %gray scale
    maxI=max(I(:));
    minI=min(I(:));
    bins=255;     %quantization bins
```

```

Ieq=uint8((bins-1)*histeq((I-minI)./(maxI-minI), bins))+1;

labels=zeros(m,n,'uint8');
N=0;
bin_mark=zeros(bins,1,'uint8');

for i=1:m
    for j=1:n
        id=Ieq(i,j);
        if bin_mark(id)==0
            N=N+1;
            labels(i,j)=N;
            bin_mark(id)=N;
        else
            labels(i,j)=bin_mark(id);
        end
    end
end

elseif ch==3 %color
    maxI=max(I(:));
    minI=min(I(:));
    Ifit=(I-minI)/(maxI-minI);
    bins=255; %quantization bins
    [labels, centers] = rgb2ind(Ifit,bins,'nodither');
    N=size(centers,1);
    labels=labels+1;
end

%% adjacency matrix
Acum=zeros(N,'uint32');
for i=1:m
    for j=1:n
        a=labels(i,j);
        if i>1
            e=labels(i-1,j);
            Acum(a,e)=Acum(a,e)+1;
        end
        if j>1
            e=labels(i,j-1);
            Acum(a,e)=Acum(a,e)+1;
        end
    end
end
A=double(Acum+Acum');

h4=sum(A,2);
P=A./repmat(h4,1,N); %transition matrix
Q=((P+P')./2)^2;

%% compute Iout
Iout=zeros(m,n,ch);

```

```

for i=1:m
    for j=1:n
        a=labels(i,j);
        w=Q(a,a);
        v=w.*I(i,j,:);

        if i>1
            e=labels(i-1,j);
            we=Q(a,e);
            v=v+we.*I(i-1,j,:);
            w=w+we;
        end
        if j>1
            e=labels(i,j-1);
            we=Q(a,e);
            v=v+we.*I(i,j-1,:);
            w=w+we;
        end
        if i<m
            e=labels(i+1,j);
            we=Q(a,e);
            v=v+we.*I(i+1,j,:);
            w=w+we;
        end
        if j<n
            e=labels(i,j+1);
            we=Q(a,e);
            v=v+we.*I(i,j+1,:);
            w=w+we;
        end

        Iout(i,j,:)=v./w;
    end
end
I=Iout;
end

```

## Filtro por correlación espacial (scorfilt).

```

function Iout=scorfilt(I,varargin)
% SPATIAL CORRELATION FILTER
%
% IN
% I:      Image to filter, (RGB color or grayscale)
% c:      'neighborhood' constant (double)
% %
% OUT

```

```

% Iout:      Filtered image (double)
%
% EXAMPLE:
%
% %read image
% imfile = fullfile(matlabroot,...
%                 'toolbox','images','imdemos','greens.jpg');
% Io=im2double(imread(imfile));
% %add noise
% sigma_n=20; %noise std. dev.
% randn('seed', 0);
% I=Io+sigma_n/255*randn(size(Io));
% figure('Name', 'input'), imshow(I);
% %filter image
% Iout = scorfilt(I);
% figure('Name', 'output'), imshow(Iout);
%
% Author:    Carlos Estrada
% Date:      May 25, 2012
% Version    1.0
%
% Configuration
r_max=5;      %max. half-size window
c=1/5;       %c 'neighborhood' constant default value

if nargin~=1
    c=varargin{1};
end

I=im2double(I);
[m n ch]=size(I);

if ch==1      %gray scale
    maxI=max(I(:));
    minI=min(I(:));
    bins=255; %quantization bins
    Ieq=uint8((bins-1)*histeq((I-minI)./(maxI-minI), bins))+1;

    labels=zeros(m,n,'uint8');
    N=0;
    bin_mark=zeros(bins,1,'uint8');

    for i=1:m
        for j=1:n
            id=Ieq(i,j);
            if bin_mark(id)==0
                N=N+1;
                labels(i,j)=N;
                bin_mark(id)=N;
            else
                labels(i,j)=bin_mark(id);
            end
        end
    end
end

```

```

end

elseif ch==3 %color
%   maxI=max(I(:));
%   minI=min(I(:));
%   Ifit=(I-minI)/(maxI-minI);
bins=255; %quantization bins
[labels, centers] = rgb2ind(I,bins,'nodither');
N=size(centers,1);
labels=labels+1;
end

%% adjacency matrix
Acum=zeros(N,'uint32');
for i=1:m
    for j=1:n
        a=labels(i,j);
        if i>1
            e=labels(i-1,j);
            Acum(a,e)=Acum(a,e)+1;
        end
        if j>1
            e=labels(i,j-1);
            Acum(a,e)=Acum(a,e)+1;
        end
    end
end
A=double(Acum+Acum');

h4=sum(A,2);
P=A./repmat(h4,1,N); %transition matrix
Q=((P+P') ./2)^2;
%% neighborhood size
teta=diag(P)'*h4./sum(h4);
w=sqrt((1-teta)/teta);
sigma_d=c*w;
r=max(1,min(r_max,round(sigma_d*2.5)));
%fprintf('\tN: %d\tw: %.4f\tsigma_d: %.4f', N,w,sigma_d);
%% compute Iout
Iout=zeros(m,n,ch);
[x,y] = meshgrid(-r:r,-r:r);
F=exp(-(x.^2 + y.^2)./(2*sigma_d^2));

for i=1:m
    iMin = max(i-r,1);
    iMax = min(i+r,m);
    di=iMin:iMax;
    for j=1:n
        jMin = max(j-r,1);
        jMax = min(j+r,n);
        dj=jMin:jMax;

        a=labels(i,j);

```

```

l = labels(di,dj);
f=F(di-i+r+1,dj-j+r+1);

w=Q(l(:),a).*f(:);
w=w./sum(w);

y=I(di,dj,:);

for cc=1:ch
    yc=y(:,:,cc);
    Iout(i,j,cc)=w'*yc(:);
end
end
end

```

## Inferencia de las intensidades mediante el espectro.

La siguiente rutina ilustra la inferencia de las intensidades a partir de la imagen de etiquetas en imágenes en escala de gris.

```

path='./man.png';
info=imfinfo(path);
fprintf('\nfile:%s', info.FileName);
Io=im2double(imread(path));
sigma_n=0;
randn('seed', 0);
I=Io+sigma_n/255*randn(size(Io));

bins=160;           %quantization bins
%% labels
[m n ch]=size(I);
if ch==1 %gray scale
    minI=min(I(:));
    maxI=max(I(:));
    Ihe=histeq((I-minI)./(maxI-minI), bins);
    Ieq=uint8((bins-1)*Ihe)+1;
    labels=zeros(m,n,'uint8');
    N=0;
    bin_mark=zeros(bins,1,'uint8');

    for i=1:m
        for j=1:n
            id=Ieq(i,j);

```

```

        if bin_mark(id)==0
            N=N+1;
            labels(i,j)=N;
            bin_mark(id)=N;
        else
            labels(i,j)=bin_mark(id);
        end
    end
end
elseif ch==3 %color
    fprintf('\nroutine not implemented for color images...');
end

%% adjacency matrix
Acum=zeros(N,'uint32');
for i=1:m
    for j=1:n
        a=labels(i,j);
        if i>1
            e=labels(i-1,j);
            Acum(a,e)=Acum(a,e)+1;
        end
        if j>1
            e=labels(i,j-1);
            Acum(a,e)=Acum(a,e)+1;
        end
    end
end
A=double(Acum+Acum');

h4=sum(A,2);
P=A./repmat(h4,1,N);
D=diag(h4);
L=(D-A);
[V,l] = eigs(L,D,3,'sa');
lamda = diag(l)
v2=V(:,2);
minv=min(v2);
maxv=max(v2);
u2a=(v2-minv)/(maxv-minv);
u2b=(maxv-v2)/(maxv-minv);
imge2a=u2a(labels);
imge2b=u2b(labels);
figure('Name','eigen 2 (a)'),imshow(imge2a);
figure('Name','eigen 2 (b)'),imshow(imge2b);

```

## Anexo II

### Resultados

(A-II.1). Tabla del PSNR (dB) obtenido por el filtro fce en imágenes en tonos de gris.

$\sigma_n$	Camera man 256x256	Lena 512x512	Barbara 512x512	boats 512x512	couple 512x512	fingerprint 512x512	hill 512x512	house 256x256	man 512x512	montage 256x256	peppers 256x256	Promedio
10	32.19	33.79	30.42	32.44	32.23	29.92	32.54	33.68	32.91	34.32	33.30	32.52
15	30.20	31.56	28.34	30.33	29.91	28.05	30.50	31.40	30.72	31.97	30.93	30.36
20	28.61	30.00	26.90	28.79	28.36	26.60	29.05	29.91	29.21	30.22	29.24	28.81
25	27.31	28.85	25.84	27.63	27.26	25.46	27.95	28.77	28.09	28.76	27.90	27.62
30	26.18	27.98	25.04	26.77	26.45	24.56	27.09	27.83	27.22	27.60	26.83	26.68
35	25.28	27.29	24.41	26.06	25.80	23.74	26.42	27.08	26.53	26.51	25.91	25.91
40	24.55	26.73	23.89	25.48	25.29	23.10	25.90	26.44	25.94	25.59	25.16	25.28
50	23.39	25.85	23.10	24.59	24.47	22.08	25.13	25.48	25.00	24.04	23.94	24.28
60	22.49	25.12	22.48	23.91	23.82	21.30	24.56	24.71	24.25	22.84	23.04	23.50
80	21.26	23.91	21.52	22.81	22.80	20.20	23.53	23.52	23.08	21.24	21.81	22.33
100	20.36	22.94	20.82	21.95	22.00	19.52	22.62	22.59	22.19	20.23	20.97	21.47

(A-II.2). Tabla del PSNR (dB) obtenido por el filtro bilateral en imágenes en tonos de gris, para la configuración *óptima* encontrada.

$\sigma_n$	Camera man 256x256	Lena 512x512	Barbara 512x512	boats 512x512	couple 512x512	fingerprint 512x512	hill 512x512	house 256x256	man 512x512	montage 256x256	peppers 256x256	Promedio
10	32.54	33.99	31.22	32.56	32.30	30.46	32.62	33.88	32.92	34.68	33.43	32.78
15	30.20	31.92	28.43	30.45	30.03	28.37	30.66	31.72	30.78	32.30	31.11	30.54
20	28.47	30.46	26.70	28.97	28.54	26.89	29.35	30.17	29.33	30.56	29.37	28.98
25	27.02	29.35	25.57	27.85	27.47	25.77	28.39	29.00	28.26	29.12	27.98	27.80
30	25.86	28.48	24.80	26.98	26.65	24.88	27.63	28.09	27.43	27.75	26.84	26.85
35	24.93	27.77	24.23	26.28	26.00	24.14	27.03	27.36	26.77	26.43	25.92	26.08
40	24.18	27.21	23.78	25.71	25.48	23.51	26.54	26.78	26.24	25.26	25.16	25.44
50	23.08	26.36	23.13	24.85	24.67	22.44	25.79	25.91	25.43	23.50	24.03	24.47
60	22.33	25.73	22.68	24.20	24.06	21.51	25.23	25.25	24.82	22.42	23.24	23.77
80	21.33	24.78	22.05	23.24	23.15	20.02	24.38	24.23	23.91	21.24	22.19	22.77
100	20.66	24.01	21.59	22.53	22.47	18.95	23.69	23.42	23.20	20.55	21.45	22.05

(A-II.3). Tabla del PSNR (dB) obtenido por el filtro bilateral en imágenes en tonos de gris, para la configuración *heurística*.

$\sigma_n$	Camera man 256x256	Lena 512x512	Barbara 512x512	boats 512x512	couple 512x512	fingerprint 512x512	hill 512x512	house 256x256	man 512x512	montage 256x256	peppers 256x256	Promedio
10	32.73	33.60	31.41	32.18	31.94	29.46	32.26	33.74	32.51	34.57	32.99	32.49
15	30.27	31.40	28.81	30.04	29.65	27.12	30.24	31.36	30.35	32.00	30.62	30.17
20	28.52	29.78	27.09	28.49	28.10	25.62	28.83	29.61	28.83	30.10	28.87	28.53
25	27.08	28.47	25.86	27.28	26.93	24.51	27.71	28.22	27.63	28.57	27.45	27.25
30	25.85	27.36	24.92	26.27	25.98	23.66	26.76	27.08	26.62	27.25	26.26	26.18
35	24.82	26.38	24.15	25.40	25.16	22.97	25.90	26.09	25.73	26.06	25.25	25.27
40	23.94	25.50	23.49	24.62	24.43	22.38	25.12	25.22	24.94	25.00	24.38	24.46
50	22.50	23.97	22.35	23.27	23.13	21.38	23.71	23.71	23.54	23.22	22.92	23.06
60	21.34	22.66	21.35	22.09	22.00	20.51	22.47	22.43	22.32	21.80	21.72	21.88
80	19.47	20.47	19.59	20.09	20.04	18.97	20.36	20.29	20.26	19.65	19.76	19.91
100	17.94	18.69	18.08	18.43	18.40	17.62	18.63	18.55	18.55	18.01	18.15	18.28

# Bibliografía

- Egiazarian, K. D. (2009). BM3D Image Denoising with Shape-Adaptive Principal Component Analysis. *Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations*.
- Foi, A. (s.f.). *Results of the proposed grayscale BM3D method*. Obtenido de <http://www.cs.tut.fi/~foi/GCF-BM3D/>
- Howe, C., Lotto, B., & Dale, P. (2006). Comparison of Bayesian and empirical ranking approaches to visual perception. *Journal of Theoretical Biology* 241, 866-875.
- Jing Huang, S. R.-J. (1997). Image indexing using color correlograms. *IEEE Conference on Computer Vision and Pattern Recognition*, (págs. 762-768).
- Morel, A. B. (2005). A review of image denoising algorithms, with a new one. *Simul*, 490--530.
- Paris, S., & Durand, F. (1 de January de 2009). A Fast Approximation of the Bilateral Filter Using a Signal Processing Approach. *Int J Comput Vis*, 81(1), 24-52.
- Shi, J., & Malik, J. (1997). Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 888--905,(22).
- Tomasi, C., & Manduchi, R. (1998). Bilateral filtering for gray and color images. *IEEE International Conference on Computer Vision*, (págs. 839–846).
- Wu, X. (1991). Efficient Statistical Computations for Optimal Color Quantization. En *Graphic Gems II* (págs. 126-133). Elsevier Science and Technology.
- Zhang , M., & Gunturk, B. (2008). Multiresolution Bilateral Filtering for Image Denoising. *IEEE Transactions on Image Processing*, 17(12), 2324 - 2333.