



Recuperación de Información Multimedia

Descriptores Globales Gris (Histogramas, Bordes, Texturas)

CC5213 – Recuperación de Información Multimedia

Departamento de Ciencias de la Computación

Universidad de Chile

Juan Manuel Barrios – <https://juan.cl/mir/> – 2020



Vector de Intensidades

- Escalar la imagen a un tamaño fijo $W \times H$
- Convertir la imagen a escala de grises
- El descriptor será un vector de largo $W \times H$ con la intensidad de cada pixel
- Comparar vectores usando:
 - Distancia L_1 : Manhattan, Taxicab o City block
 - Distancia L_2 : Euclidiana

$$L_1(\vec{x}, \vec{y}) = \sum_{i=1}^n |x_i - y_i|$$

$$L_2(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$





Ordinal Measurement Descriptor (OMD)

- Dividir la imagen en zonas (ej., 8 x 8)
- Calcular la intensidad promedio en cada zona
- Ordenar las intensidades de menor a mayor
- Representar cada zona por la posición en que queda en el arreglo ordenado
- Distancia de Hamming:

$$H(\vec{x}, \vec{y}) = \sum_{i=1}^n d_i \quad d_i = \begin{cases} 0 & \text{si } x_i = y_i \\ 1 & \text{si } x_i \neq y_i \end{cases}$$

Ordinal Measurement (OMD)



64	61	46
135	146	116
185	174	145

3	2	1
5	7	4
9	8	6

10	40	20
50	70	90
30	80	60

1	4	2
5	7	9
3	8	6

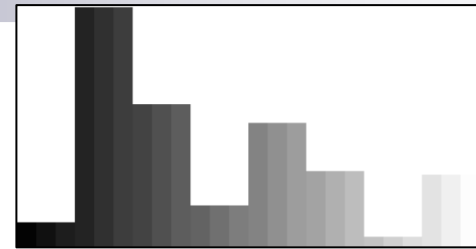


Histograma de intensidades

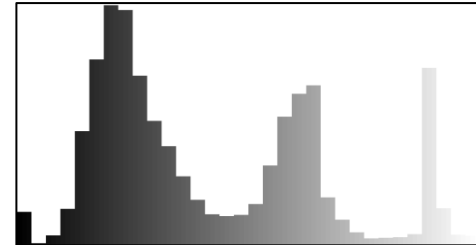
- Calcular un histograma de grises de la imagen
- Parámetros:
 - Cantidad de bins
 - Tipo de asignación (hard vs soft)
 - Tipo de normalización (norma L1, L2)
 - Tipo de dato a guardar (float, unsigned char)
 - Función de distancia: Minkowski (L_p), Test estadístico como χ^2 o Kullback-Leibler

$$\chi^2(\vec{x}, \vec{y}) = \sum_{i=1}^n \frac{(x_i - \bar{p}_i)^2}{\bar{p}_i} \quad \bar{p}_i = \frac{x_i + y_i}{2}$$

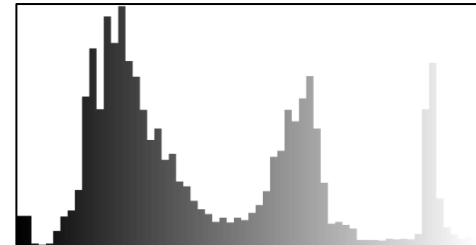
Histograma



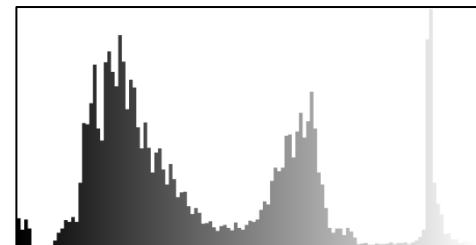
8 bins



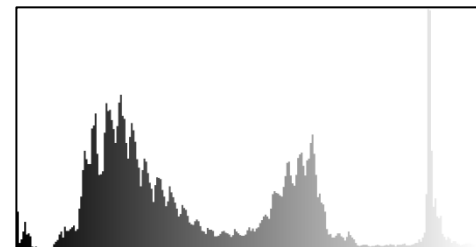
32 bins



64 bins



128 bins



256 bins

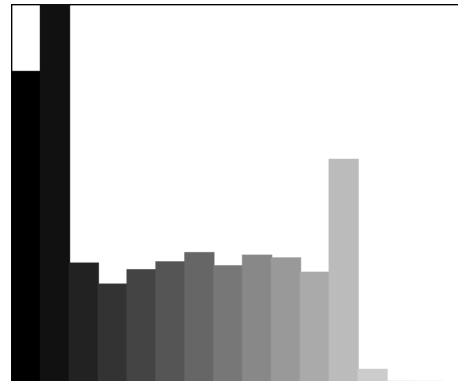


Histograma por Zonas

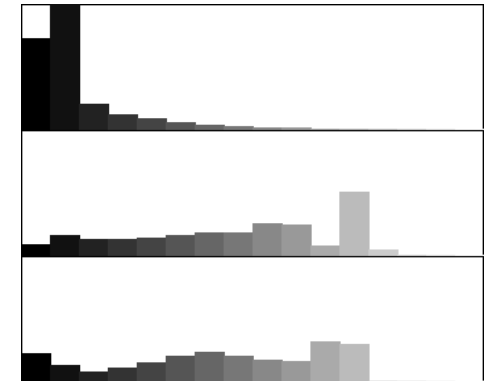
- Un histograma descarta la ubicación espacial de los pixeles
- La información espacial se puede mantener (en cierta medida) si se divide la imagen en $P \times Q$ zonas y se calcula un histograma de N bins independiente por zona
 - Asignación “soft” entre zonas → Interpolación “tri-lineal”
- Se concatenan todos los histogramas para formar un descriptor de largo $P \times Q \times N$
 - Se puede también concatenar el histograma global de la imagen más otras divisiones
- Concatenar divisiones cada vez más finas
 - Por ejemplo $1 \times 1 + 2 \times 2 + 4 \times 4$, total de 21 zonas
 - Estrategia conocida como “**Spatial Pyramid**”



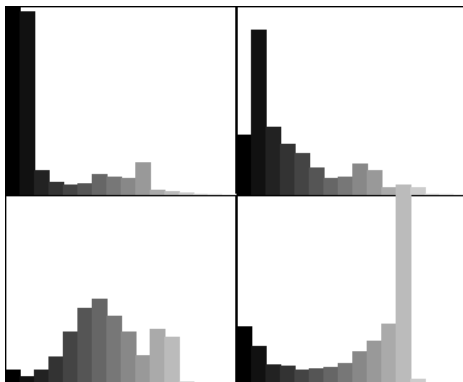
Histograma por Zonas



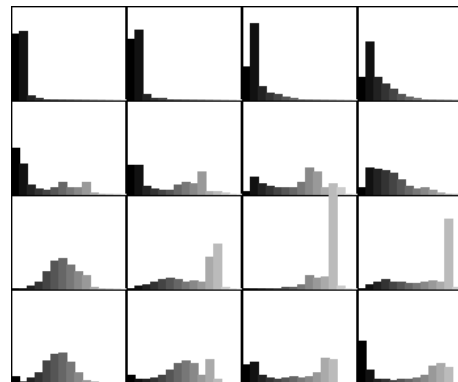
1 x 1 x 16 bins



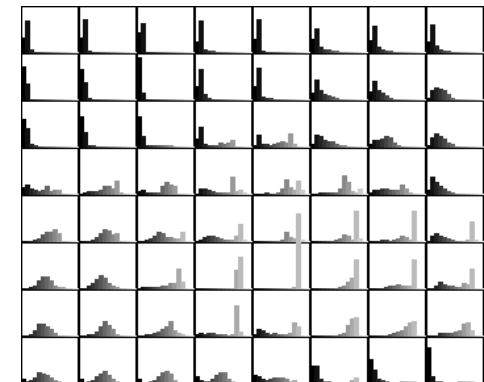
1 x 3 x 16 bins



2 x 2 x 16 bins



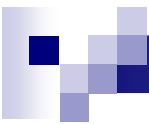
4 x 4 x 16 bins



8 x 8 x 16 bins



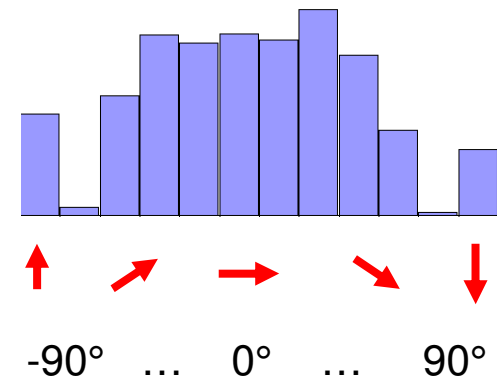
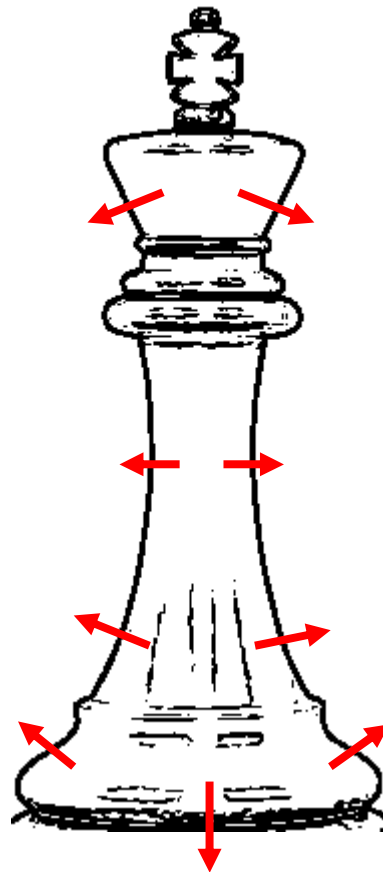
Descriptores de Bordas



Histogram of Oriented Gradients (HOG)

- Calcular pixeles de borde
 - Ej: filtros Sobel y luego un threshold sobre magnitud del gradiente
- Para cada pixel de borde calcular su dirección:
$$\theta(i, j) = \arctan \left(\frac{I_y(i, j)}{I_x(i, j)} \right)$$
- Histograma de las direcciones del gradiente
 - Usualmente 18 bins en el rango $[-180, 180]$ o en el rango $[-90, 90]$ (juntando los sentidos opuestos)
- Mejoras:
 - Mejorar detección de bordes (segunda derivada, etc.)
 - Mejorar histograma (usar zonas, asignación suave, normalización, etc.)

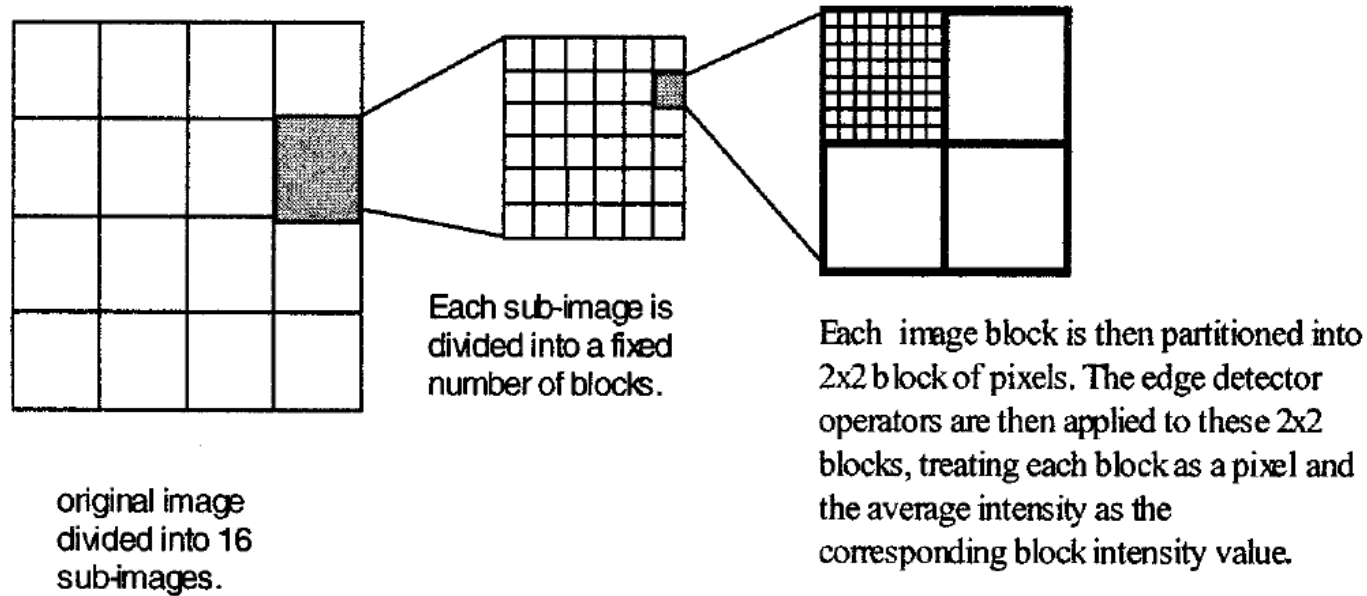
Histogram of Oriented Gradients (HOG)





Edge Histogram Descriptor (EHD)

- Dividir la imagen en 4x4 zonas
 - Cada zona dividir en muchos bloques de 2x2
 - Mediante filtros detectar la orientación dominante en cada bloque (probar 5 filtros)
 - La orientación dominante debe superar cierto umbral
 - Cada zona se representa por un histograma de los filtros dominantes dentro de ésta (cantidad de bloques de 2x2 para cada filtro)
- Comparación con distancia Manhattan



1	-1
1	-1

1	1
-1	-1

$\sqrt{2}$	0
0	$-\sqrt{2}$

0	$\sqrt{2}$
$-\sqrt{2}$	0

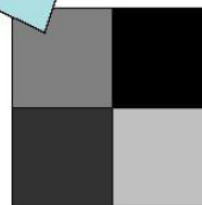
2	-2
-2	2



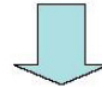
Variación EHD

- Una variante es no usar un histograma si no que representar cada bloque por su tipo de borde
- Dividir la imagen en N bloques de 2x2
 - Mediante filtros detectar la orientación dominante en una zona
 - Representar el bloque por su orientación dominante (entre 10 posibles) o sin orientación (si no supera un valor umbral)
 - Vector de largo N, cada dimensión varía entre 0 y 10 (0=sin orientación o “No-Edge”)
- Comparación mediante distancia de Hamming

(1) Partition the image into N blocks



(2) Generate a 2x2 pixel size icon for each block



(3) Classify the icon into edge type using a filter

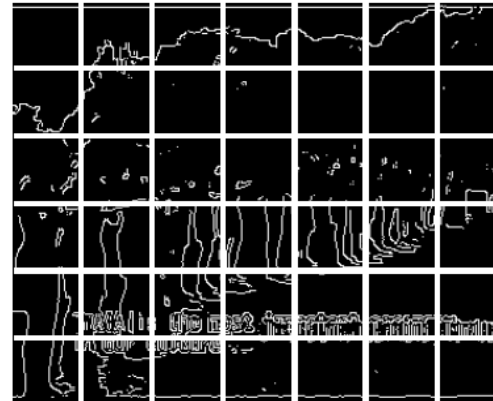
Edge Type	Filter
0 deg.	-1 1
	-1 1
45 deg.	0 $\sqrt{2}$
	$-\sqrt{2}$ 0
90 deg.	1 1
	-1 -1
135 deg.	$\sqrt{2}$ 0
	0 $-\sqrt{2}$

Edge Type	Filter
180 deg.	1 -1
	1 -1
-135 deg.	0 $-\sqrt{2}$
	$\sqrt{2}$ 0
-90 deg.	-1 -1
	1 1
-45 deg.	$-\sqrt{2}$ 0
	0 $\sqrt{2}$

Edge Type	Filter
Non-Direction1	2 -2
	-2 2
Non-Direction2	-2 2
	2 -2
No Edge	

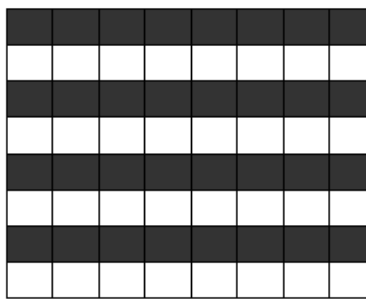
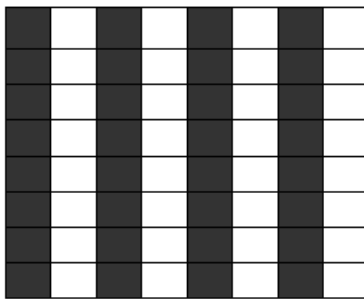
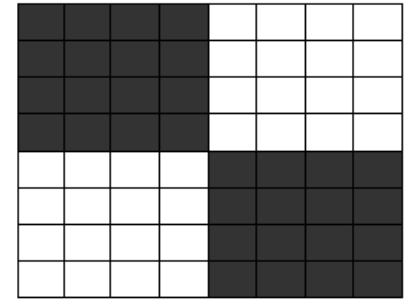
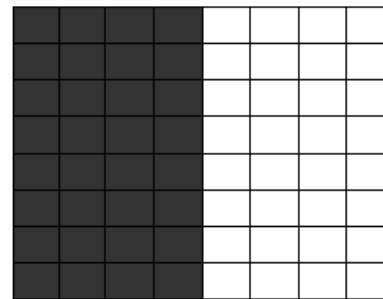
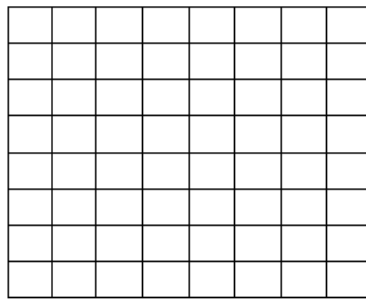
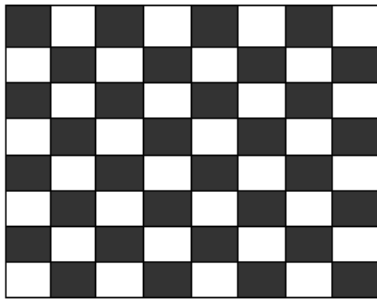
Un descriptor basado en Canny

- Calcular bordes según Canny, dividir imagen $N \times M$ zonas
- Calcular el centroide de bordes y cuantizarlo en $P \times Q$ posiciones
- Vector de $N \times M$ dimensiones, cada dim entre 1 y $P \times Q$
 - Parámetros $N=M=15$ y $P=Q=4$
- Comparar vectores con distancia de Hamming



Ejercicio

- Calcular los descriptores globales anteriores para estas imágenes de 8x8:



$$\frac{\partial I}{\partial y}(x, y) = I(x, y + 1) - I(x, y)$$

$$\frac{\partial I}{\partial x}(x, y) = I(x + 1, y) - I(x, y)$$



Descriptores de Texturas (Frecuencias)



Coeficientes DCT

- Tomar los primeros n^2 coeficientes (esquina superior izquierda) de aplicar DCT sobre la imagen y cuantizar:

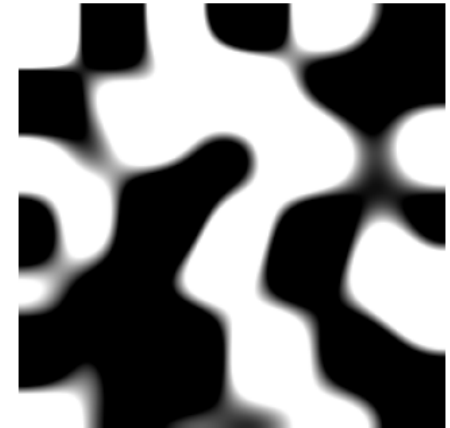
$$\sigma(i) = \begin{cases} 1 & \text{if } DCT\left(\lfloor \frac{i}{n} \rfloor, i - \lfloor \frac{i}{n} \rfloor n\right) \geq m \\ 0 & \text{otherwise} \end{cases} \quad \text{for } i \in [1, n^2]$$

- Vector de n^2 bits

m =media de los n^2 coeficientes



n=8



n=7



n=6



n=5

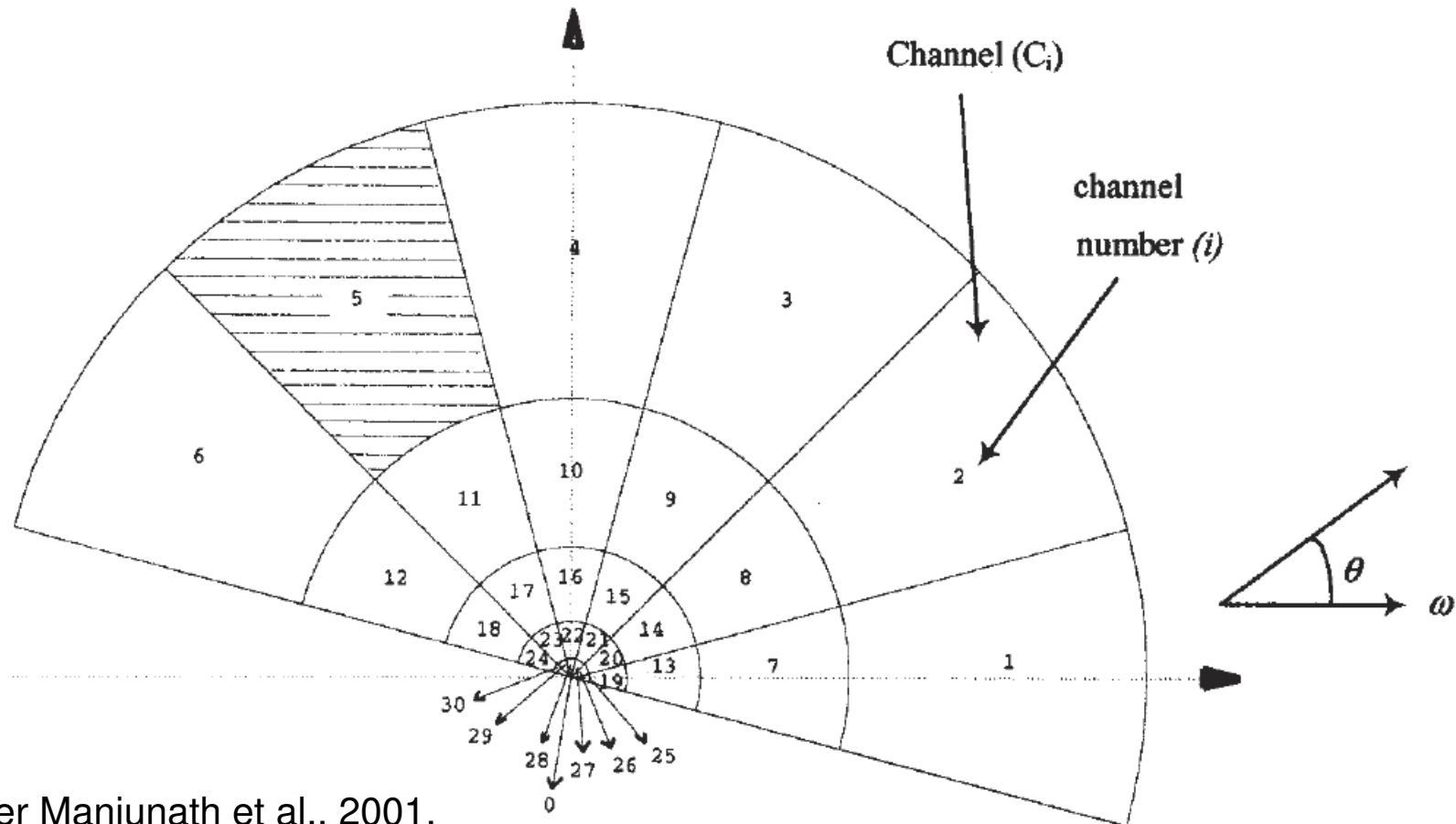
Ordinal Measurement con DCT

- Usando DCT, el descriptor OMD puede ser robusto a flip
- Calcular la matriz de intensidades, aplicar DCT y crear una nueva matriz con los coeficientes
- A la matriz de coeficientes aplicar Ordinal Measurement

$$\begin{bmatrix} 1 & 4 & 2 \\ 5 & 7 & 9 \\ 3 & 8 & 6 \end{bmatrix} \quad \begin{bmatrix} 2 & 4 & 1 \\ 9 & 7 & 5 \\ 6 & 8 & 3 \end{bmatrix} \quad \begin{bmatrix} 15.0 & -3.3 & -2.8 \\ -4.1 & 1.0 & 0.6 \\ -4.2 & 1.2 & -2.0 \end{bmatrix} \quad \begin{bmatrix} 15.0 & 3.3 & -2.8 \\ -4.1 & -1.0 & 0.6 \\ -4.2 & -1.2 & -2.0 \end{bmatrix}$$

Homogeneous Texture Descriptor

- Dividir el espacio de frecuencias en 30 zonas:

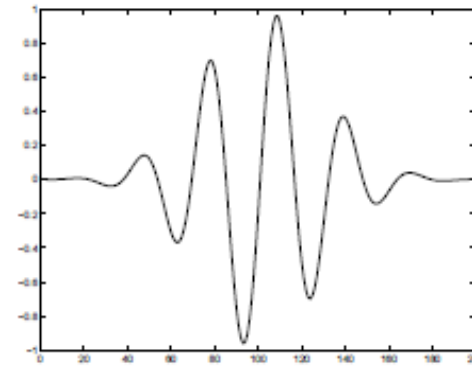
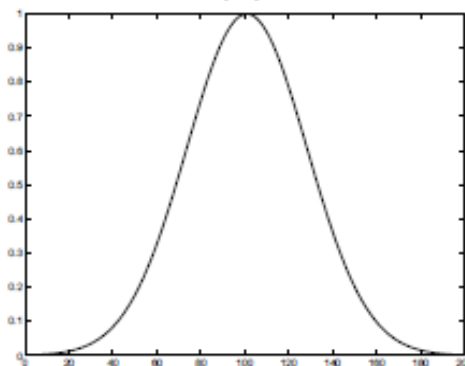
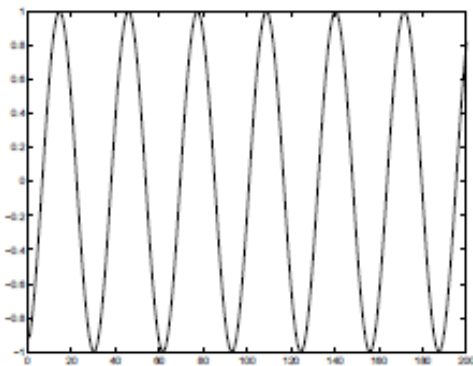


Ver paper Manjunath et al., 2001.

Homogeneous Texture Descriptor

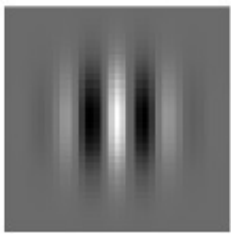
- Utilizar filtros de Gabor:

$$h_p(k, \ell) = \frac{1}{2\pi\sigma_p^2} \exp\left(\frac{-(k^2 + \ell^2)}{2\sigma_p^2}\right) \exp(j2\pi(u_p k + v_p \ell))$$

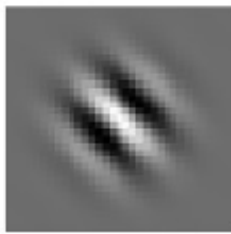


Filtros Gabor

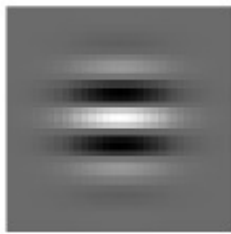
cosine, 0 deg



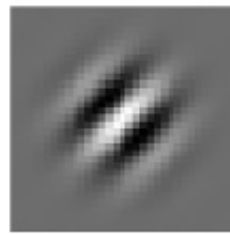
cosine, 45 deg



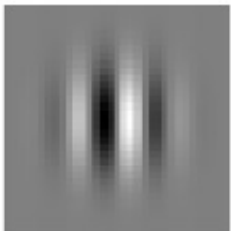
cosine, 90 deg



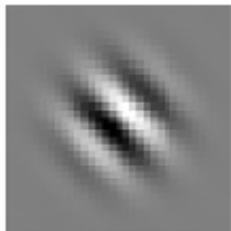
cosine, 135 deg



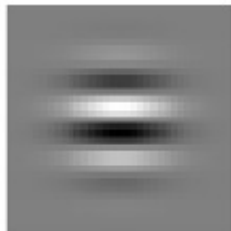
sine, 0 deg



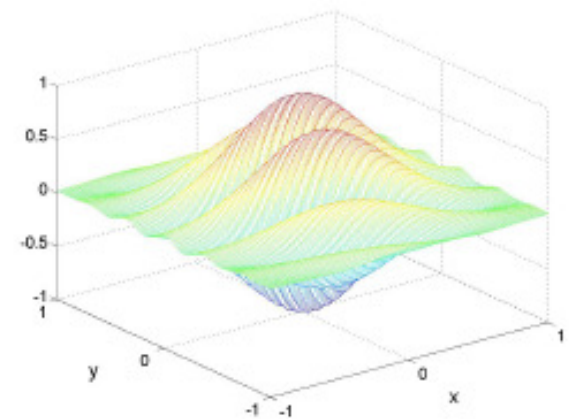
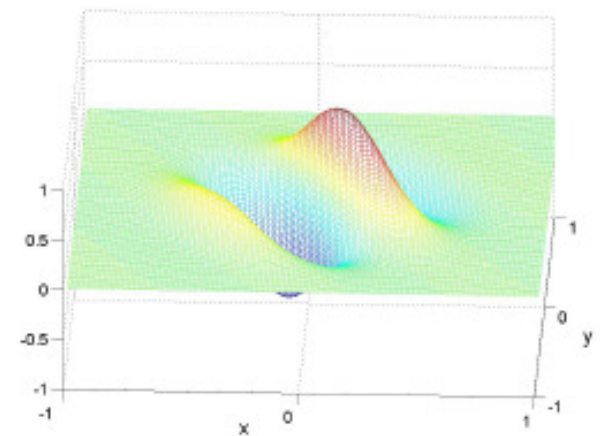
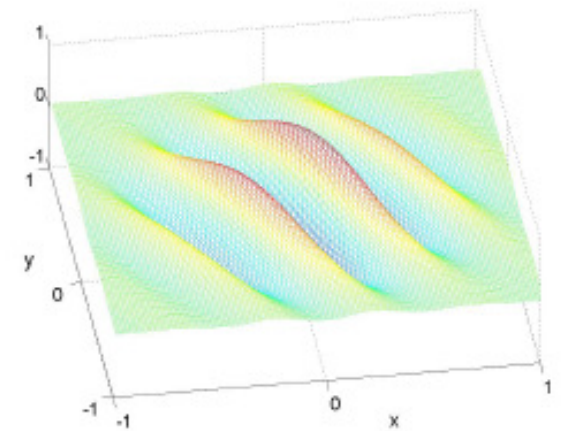
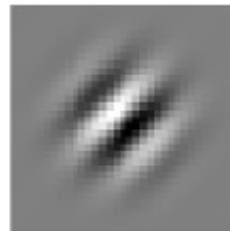
sine, 45 deg



sine, 90 deg



sine, 135 deg





Homogeneous Texture Descriptor

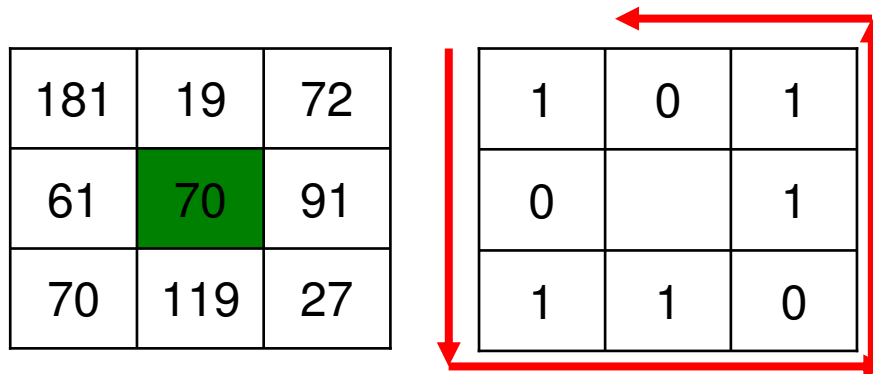
- Para cada zona se aplica un filtro Gabor, según la orientación y el tamaño
- Se representa cada zona según el logaritmo de la media y de la desviación estándar
- Además se guarda la media y desviación estándar de la imagen
- Vector de 62 dimensiones:

$$(\mu, \sigma, \mu_1, \dots, \mu_{30}, \sigma_1, \dots, \sigma_{30})$$

- Dividir cada valor por la desviación estándar de esa dimensión en la bd
- Distancia Manhattan

Local Binary Patterns

- En una imagen se convierten los pixeles en gris en pixeles del tipo de patrón
- El valor de gris de un pixel se compara con sus 8 pixeles vecinos
 - Es un 1 si es mayor o igual al pixel central
- Convertirlos a una cadena de 8 bits recorriendo según algún orden.
- Interpretar los 8 bits como entero que es el nuevo valor del pixel
- El descriptor de la imagen es un histograma de estos valores



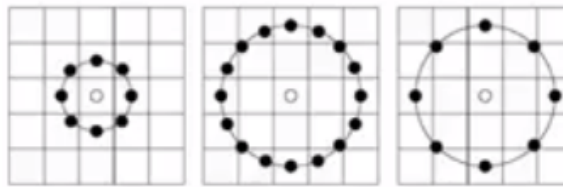
10110110 = **182**

Nuevo valor del pixel

Local Binary Patterns

■ Variantes:

- Considerar vecindades a un radio r (interpolar pixeles)

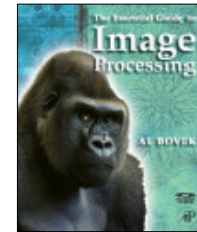


- El comparar 2 pixeles, es 1 si la diferencia es superior a cierto umbral
- Restringir los posibles patrones:
 - Las cadenas de bits con tres o más transiciones 0->1 1->0 se agrupan en un único código
 - Usar machine learning para seleccionar solo las mejores patrones

Bibliografía

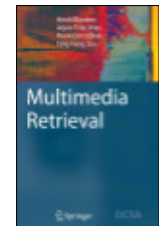
- **The Essential Guide To Image Processing.** Bovik. 2009.

- Cap 3 (Histogramas)



- **Multimedia Retrieval.** Blanken et al.

- pág 143 (Gabor)





Papers

- **Manjunath et al.** “Color and Texture Descriptors”. 2001.
- **Kim et al.** “Content-based image copy detection” y “Spatiotemporal Sequence Matching for Efficient Video Copy Detection”. 2002 y 2005.
- **Iwamoto et al.** “Image signature robust to caption superimposition for video sequence identification”. 2006.
- **Naturel et al.** “A Fast Shot Matching Strategy for Detecting Duplicate Sequences in a Television Stream”. 2005.
- **Hampapur et al.** “Comparison of distance measures for video copy detection” y “Videogrep: video copy detection using inverted file indices”. 2001 y 2002.
- **Ojala et al.** “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns”. 2002.