

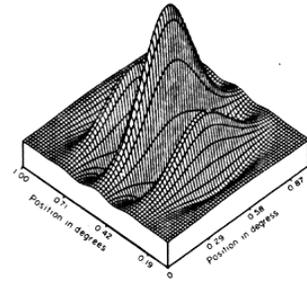
## 4. Análisis Espacial Frecuencial de Imágenes



The image shows a musical score with three staves. The top staff contains a sequence of 16 eighth notes, each with a small 'phi' symbol above it. The middle staff shows these notes grouped into four pairs, each pair connected by a horizontal brace. The bottom staff shows the four pairs grouped into two larger groups, each connected by a horizontal brace. To the right of the staves, a list of mathematical expressions is aligned with the hierarchical levels of the music:

- $\phi(16t) \cdots \phi(16t - 15)$
- $\phi(8t) \cdots \phi(8t - 7)$
- $\phi(4t) \cdots \phi(4t - 3)$
- $\phi(2t)\phi(2t - 1)$
- $\phi(t)$

## 4.1. Funciones Gabor

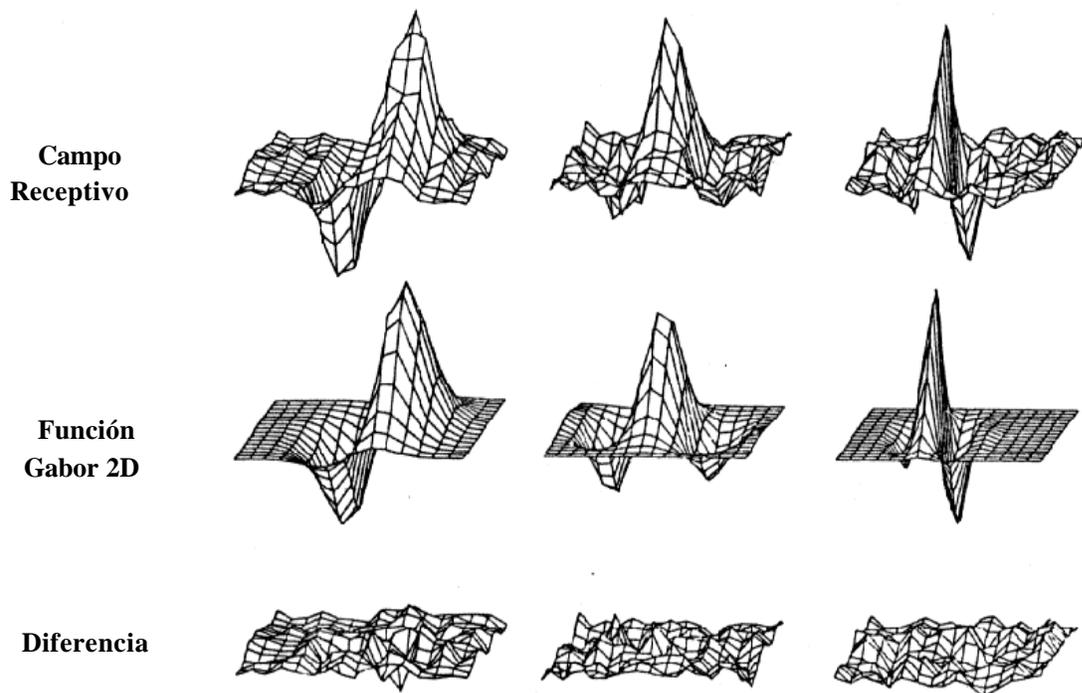


- Propuestas originalmente (forma 1D) por Dennis Gabor en 1946.
- Principio de incerteza informacional: producto del ancho de banda y de la duración temporal de una señal finita se encuentra limitado inferiormente:  
$$\Delta t \Delta f \geq \frac{1}{4} \pi$$
- Funciones Gabor alcanzan el límite inferior de este producto (únicas funciones complejas), corresponden al producto entre una función Gaussiana y una función sinusoidal.
- A principios de la década de los ochenta son formuladas bidimensionalmente (Daugman, 1980) y usadas para modelar el comportamiento de las células simples y complejas de la corteza visual (Áreas V1 y V2).
- Hoy en día corresponden a una poderosa herramienta de procesamiento en el ámbito del procesamiento de imágenes. En este contexto se denominan Filtros Gabor.

- Su principal características es que permiten obtener información acerca de la respuesta en frecuencia de las señales a procesar, sin sacrificar información acerca de su localización espacial, como lo hace la Transformada de Fourier.

### 4.1.1. Aspectos Básicos

#### Funciones Gabor y Células Corticales



## Definición

$$h(x, y) = g(x', y') \cdot \exp(2pj(Ux + Vy))$$

$$g(x, y) = \left( \frac{1}{2pIs^2} \right) \cdot \exp \left[ - \frac{\left( \frac{x}{I} \right)^2 + y^2}{2s^2} \right]$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \mathbf{M}_R \cdot \begin{pmatrix} x \\ y \end{pmatrix} \quad ; \quad \mathbf{M}_R = \begin{pmatrix} \cos f & \sin f \\ -\sin f & \cos f \end{pmatrix} ,$$

donde  $M_R$  corresponde a una matriz de rotación de coordenadas;  $\phi$  es el ángulo (con respecto al eje x),  $\lambda$  es la relación entre los ejes principales de la función Gaussiana,  $\sigma$  es la desviación estándar de la función Gaussiana, U y V son las frecuencias de la función sinusoidal.

## Respuesta en frecuencia

$$H(u, v) = \exp \left( -2p^2 s^2 \cdot \left[ (u' - U')^2 \cdot I^2 + (v' - V')^2 \right] \right) ,$$

donde  $(u', v')$  y  $(U', V')$  corresponden a una rotación (a través de  $M_R$ ) de las variables  $(u, v)$  y  $(U, V)$ . En el plano de las frecuencia  $H(u, v)$  corresponde a una función Gaussiana anisotrópica, pasa-banda, cuya frecuencia central tiene una magnitud  $F=(U^2+V^2)^{1/2}$  y una orientación  $\theta=\arctan(V/U)$  respecto al eje u.

- Las funciones Gabor pueden ser simplificadas si  $\phi=\theta$  . En este caso  $h(x, y)$  y  $H(u, v)$  quedan definidas por:

$$h(x,y) = g(x', y') \cdot \exp(2\mathbf{p}jFx')$$

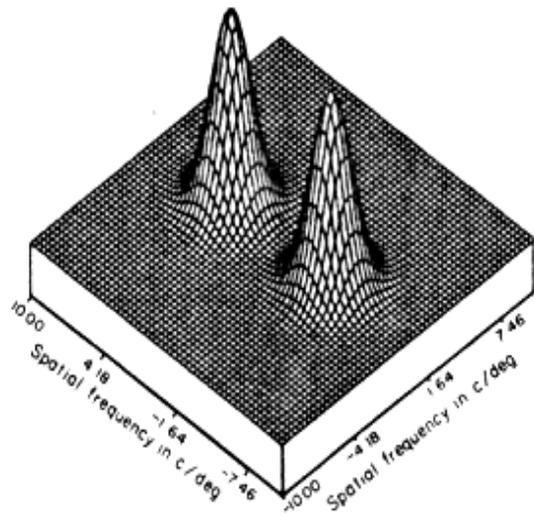
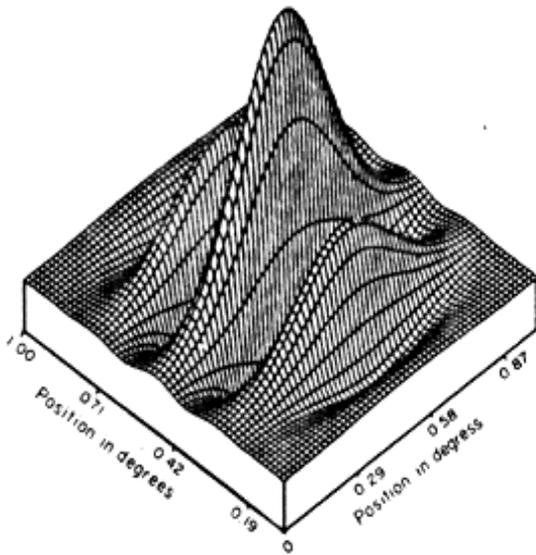
$$H(u,v) = \exp\left(-2\mathbf{p}^2\mathbf{s}^2 \cdot \left[(u-F)^2 \cdot I^2 + (v')^2\right]\right).$$

- Frecuentemente se utilizan los componentes par ( $h_c$ ) y/o impar ( $h_s$ ) de los filtros:

$$h_c(x,y) = g(x', y') \cdot \cos(2\mathbf{p}Fx')$$

$$h_s(x,y) = g(x', y') \cdot \sin(2\mathbf{p}Fx')$$

### Función Gabor (componente par) y su respuesta en frecuencia



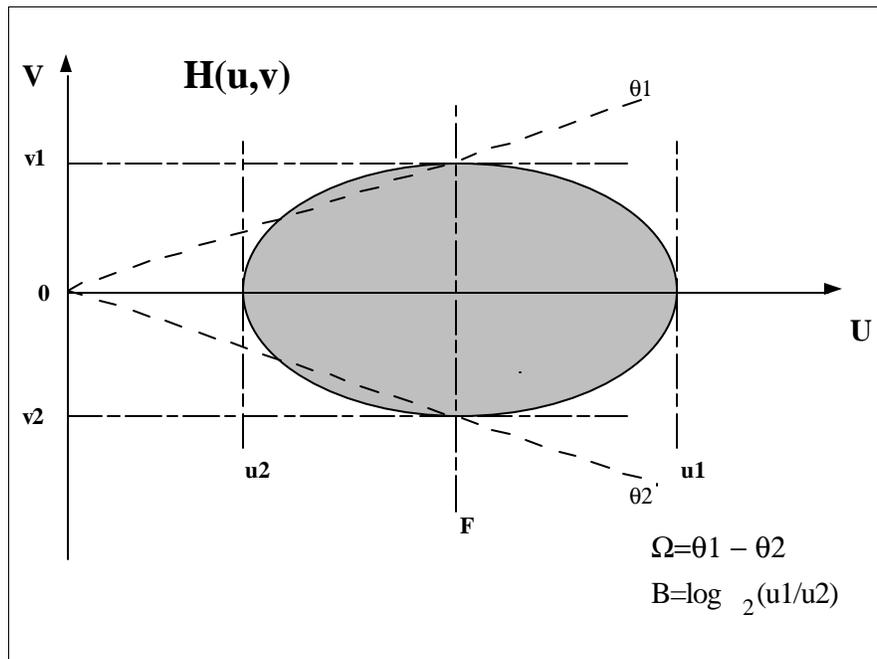
## Ancho de banda de los filtros

- Dado que las funciones Gabor son utilizadas frecuentemente como filtros, es muy importante definir los anchos de banda frecuencial (B) y orientacional ( $\Omega$ ):

$$B = \log_2 \left[ \frac{pF1s + a}{pF1s - a} \right] \text{ [Octavas] ,}$$

$$\Omega = 2 \cdot \arctan \left[ \frac{a}{pFs} \right] \text{ [rad]}$$

con:  $a = \sqrt{\ln 2} / 2$



### 4.1.2. Expansión-Gabor de una función

- Cualquier función de dimensión finita puede ser expresada como la suma ponderada de diferentes *logons* o funciones Gabor. En el caso bi-dimensional, una función arbitraria  $t(x,y)$  (una imagen), puede ser expandida de la siguiente forma:

$$t(x, y) = \sum_{r=-\infty}^{\infty} \sum_{s=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \mathbf{b}_{rsmn} \cdot h_{rsmn}(x, y)$$

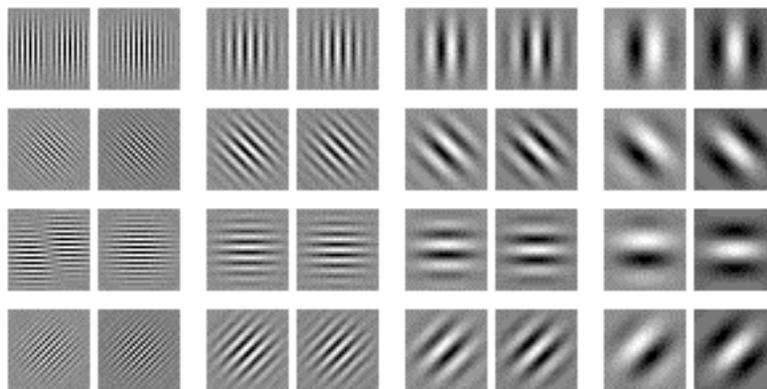
con:

$$h_{rsmn}(x, y) = g(x' - x_r, y' - y_s) \cdot \exp(2\mathbf{p}j \cdot [U_m(x - x_r) + V_n(y - y_s)]),$$

donde los  $\beta_{smn}$  corresponden a los coeficientes Gabor de la expansión. Los vectores base de la expansión son las funciones Gabor (logons) parametrizadas  $h_{rsmn}$ .

### 4.1.3. Filtrado Mediante Funciones Gabor

- Dado que los filtros Gabor son lineales, el filtrado se realiza mediante convolución.
- El resultado del filtrado Gabor es en rigor complejo, por este motivo se usa normalmente la magnitud de la respuesta, i.e.  $(\text{RE}^2 + \text{IM}^2)^{1/2}$ , o  $(|\text{RE}| + |\text{IM}|)$ . En algunas aplicaciones especializadas se ha utilizado también la información de fase.
- Por simplicidad, algunas veces se utiliza sólo la componente par o la impar de los filtros.
- Frecuentemente se debe utilizar un banco de filtros, sintonizados a diferentes frecuencias y orientaciones, para lograr un adecuado análisis de las imágenes. A manera de ejemplo:



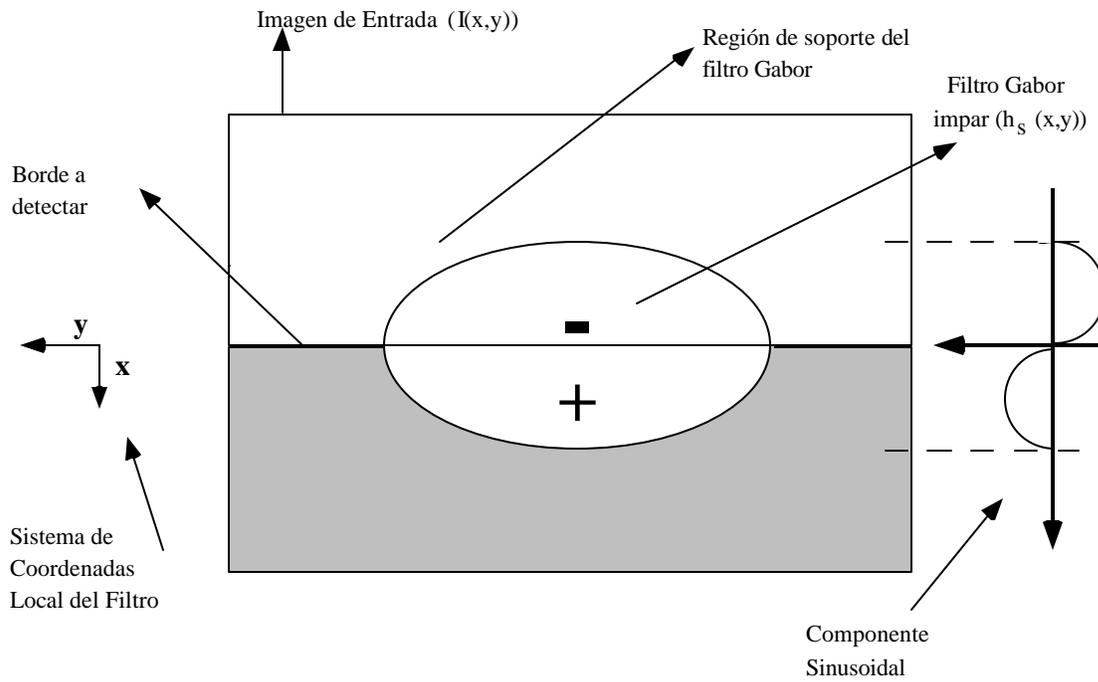
- El usar un banco de filtros hace que este tipo de filtrado sea normalmente lento. Por este motivo se han buscado alternativas rápidas de implementación (aplicación 1D de los filtros, pirámides de filtros Gabor, Gabor-Wavelets, implementaciones en hardware y ópticas, ...).

## 4.1.4. Aplicaciones de Filtros Gabor

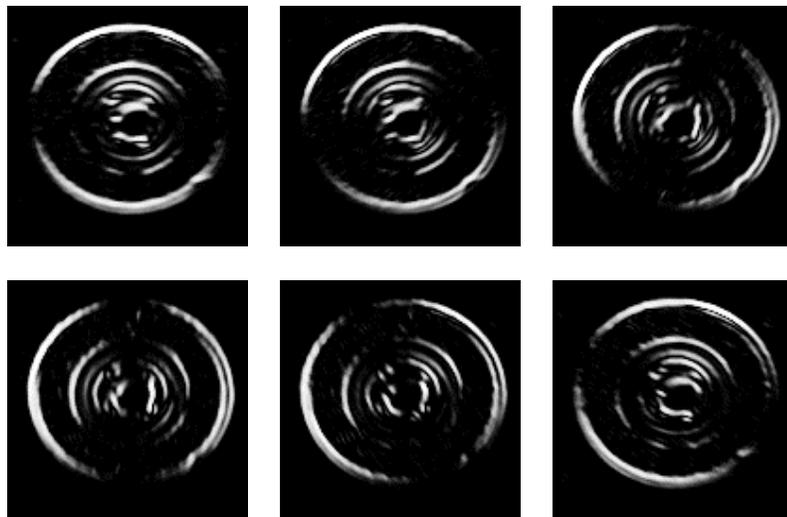
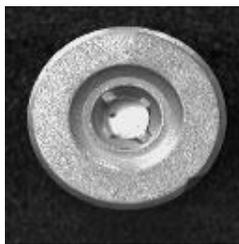
- Principales Aplicaciones:
  - Detección de contornos
  - Análisis y clasificación de texturas
  - Síntesis de texturas
  - Detección de fallas en imágenes texturadas
  - Análisis de movimiento
  - Análisis de imágenes stereo
  - Búsqueda de información en bancos de datos de imágenes
  - Identificación biométrica (iris, caras, etc.)
  - Compresión de imágenes y video
  - Reconocimiento de escritura
  - Determinación de la dimensión fractal en imágenes texturadas
  - Separación de texto e imágenes para procesamiento automático de documentos
  - Fusión de información proveniente de diferentes sensores

## Ejemplo1:

### Detección de Contornos



### Ejemplo de detección de contornos



**Ejemplo 2:**  
**Compresión de imágenes usando número variable de**  
**coeficientes Gabor**



## 4.2. Wavelets y Bancos de Filtros

### 4.2.1. Motivación



$\phi(16t) \cdots \phi(16t - 15)$   
 $\phi(8t) \cdots \phi(8t - 7)$   
 $\phi(4t) \cdots \phi(4t - 3)$   
 $\phi(2t)\phi(2t - 1)$   
 $\phi(t)$

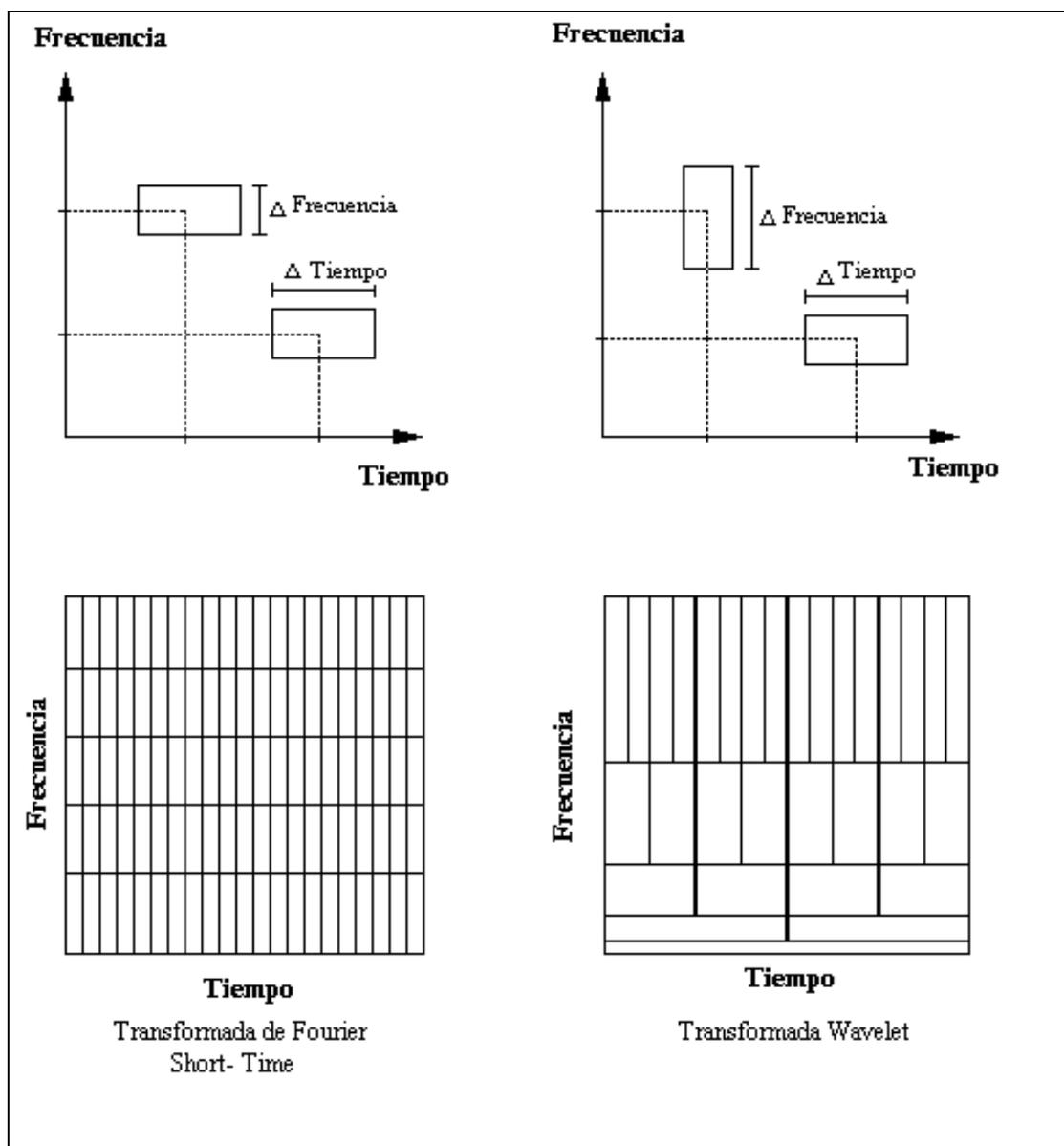
- En la notación musical clásica, cada nota representa una frecuencia y una posición temporal. De esta forma las melodías son representadas simultáneamente en el plano frecuencial y temporal, permitiendo a los músicos interpretarlas adecuadamente
- A diferencia de esto, las herramientas utilizadas comúnmente en Ingeniería para analizar o procesar señales, permiten representar las señales solamente en el plano espacial (temporal) o frecuencial
- A modo de ejemplo, la Transformada de Fourier al representar un señal en el plano de las frecuencias, pierde totalmente la información espacial (temporal)
- Las wavelets corresponden a una nueva herramienta de análisis que permiten representa una señal simultáneamente en el plano frecuencial y espacial (temporal)

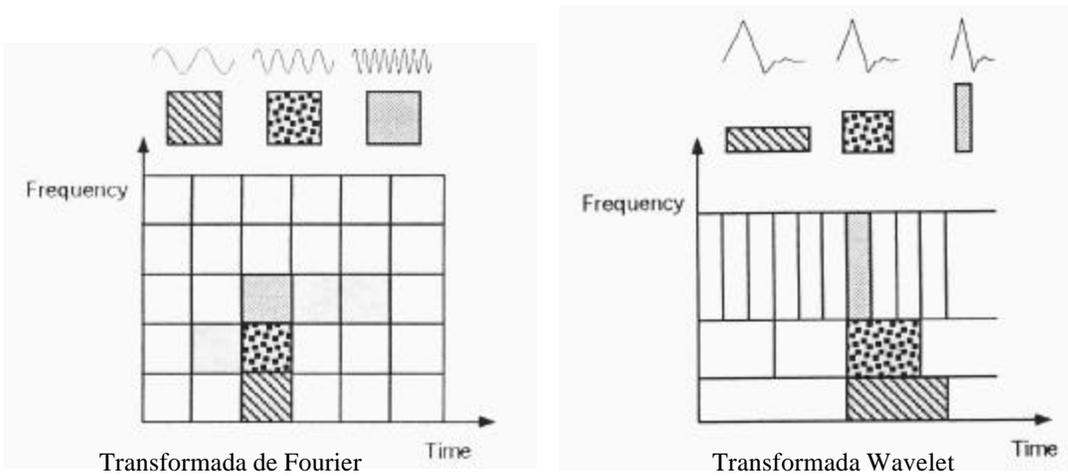
- En forma totalmente equivalente permiten procesar un señal bajo diferentes escalas (resoluciones espaciales y frecuenciales), en forma similar a como el sistema visual humano procesa las imágenes del mundo real

## 4.2.2. Introducción histórica

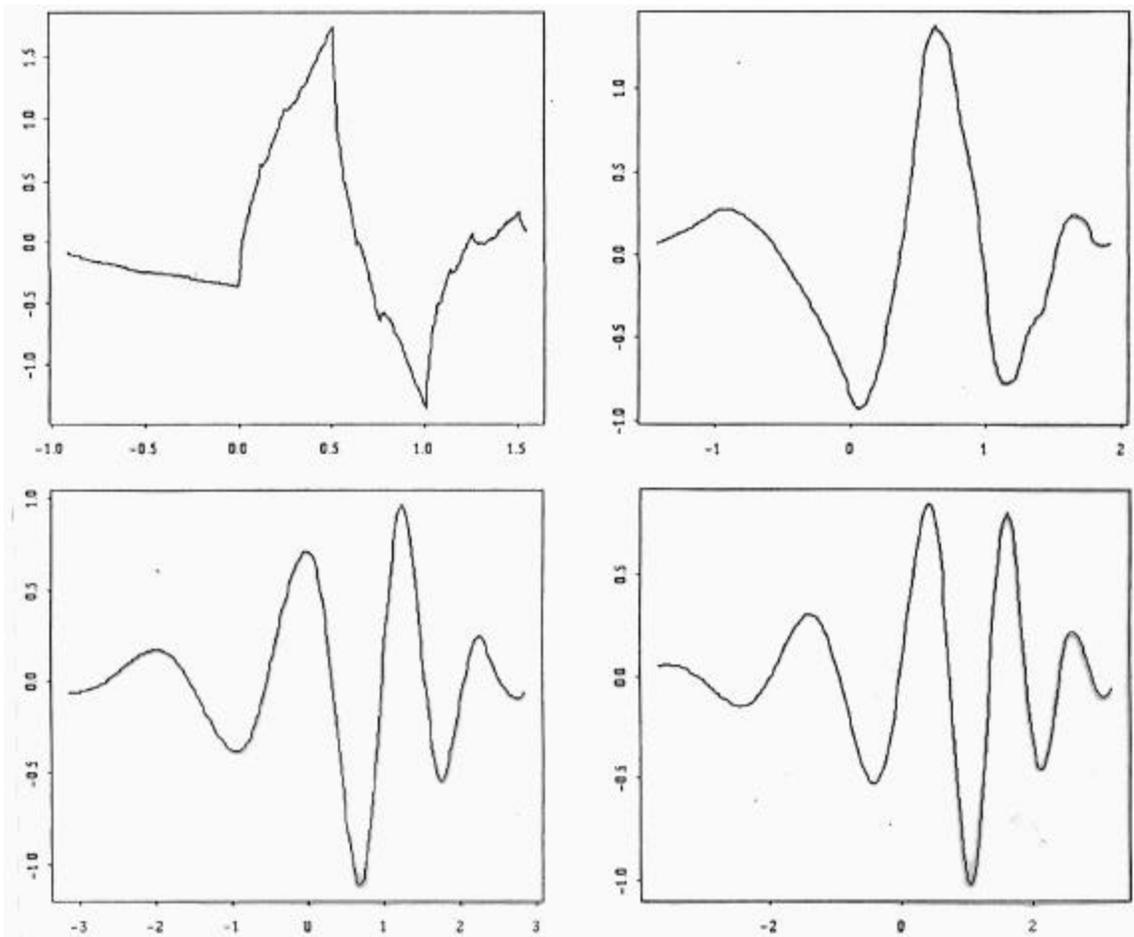
- Fourier (1807): Fourier Análisis
- Haar (1910): Funciones Haar
- Gabor (1943): Principio de Incerteza Comunicacional
  - $Dx \times Df \geq 1/4p$
- Hubel & Wiesel (años 60): Estudios Sistema Visual
- Daugman (1980): Modela células sistema visual primario mediante filtros Gabor bi-dimensionales
- Morlet (finales años 70): Ondelettes como una alternativa a la *Short time Fourier Transform*
- Grossmann (1985), Meyer (1985): Desarrollo sistemático y marco teórico de la teoría de wavelets
- Daubechies (finales años 80): Estudio matemático de las wavelets e introducción de una familia importante de wavelets

- Mallat (finales años 80): Wavelets para descomposición de señales a través de multiresolución





- A manera de ejemplo tenemos la familia de wavelets de Daubechies



### 4.2.3. Definición

- Las wavelets son ondas localizadas, que en vez de oscilar indefinidamente, caen a cero
- Las wavelets cumplen el mismo papel que las funciones seno y coseno en el análisis de Fourier (series o transformadas), es decir son los vectores base que permiten representar señales en un espacio diferente
- Para encontrar estos vectores base se define una wavelet inicial conocida como *mother wavelet*:  $\mathbf{y}(x)$
- Las otras wavelets (vectores base) se encuentran mediante traslaciones y dilataciones (cambios de escala) de la mother wavelet:

$$\mathbf{y}\left(\frac{(x-b)}{a}\right), (a,b) \in \mathbb{R}^+ \times \mathbb{R}$$

Comúnmente se elige  $a$  y  $b$  como múltiplos de 2 (dyadic Wavelets)

- A través de las traslaciones se logra obtener localización espacial y a través de las dilataciones se logra localización frecuencial (cambios de escala)
- Complejidad computacional  $O(n \cdot \log n)$ , pero utilizando algoritmos rápidos (Fast Wavelet Transform)  $O(n)$
- Una señal se puede representar a través de Wavelets de la siguiente forma:

$$y(x) = \sum_{j,k}^m b_{jk} \cdot \mathbf{y}_{jk}(x)$$

$b_{jk}$ : coeficientes de la expansión

$\psi_{jk}$ : Funciones base de la expansión (vectores l.i.)

- Mallat (1989) unifica el concepto Wavelets con el de multiresolución, e introduce la función de escala  $\phi(x)$  tal que:

$$y(x) = b_{00}f(x) + \sum_{j=0}^{n-1} \sum_{k=0}^n b_{jk} \cdot y_{jk}(x)$$

- Asimismo asocia el concepto de Wavelets con el de filtrado sub-banda (filtros en cuadratura) donde se define:

$g(x)$  : filtro pasa-altos

$h(x)$  : filtro pasa-bajos tal que

$g(x) : (-1)^k h(1-k)$

$g(x)$  y  $h(x)$  definen la expansión de Wavelets,  $\phi(x)$  y  $\Psi(x)$  se definen en forma recursiva, utilizando ambos filtros:

$$f(x) = \sqrt{2} \sum_k h(k) \cdot f(2x - k)$$

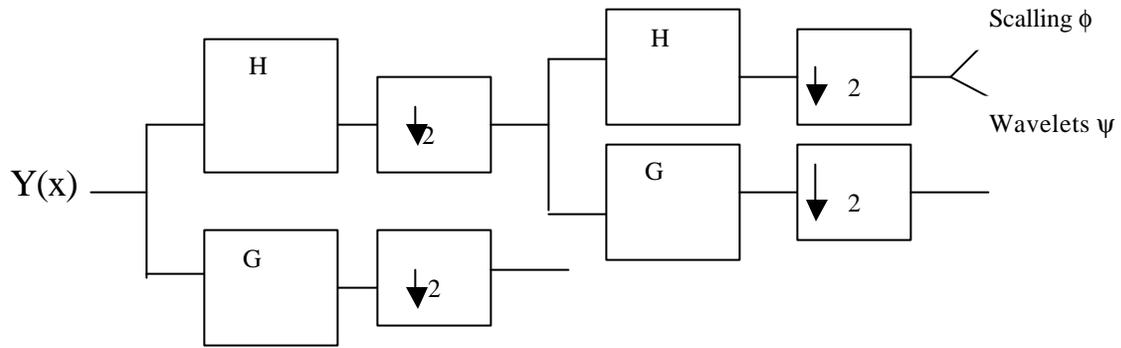
$$y(x) = \sqrt{2} \sum_k g(k) \cdot f(2x - k)$$

- Finalmente se definen los operadores H y G tal que

$$(H \cdot a)_k = \sum_n h(n - 2k) \cdot a_n$$

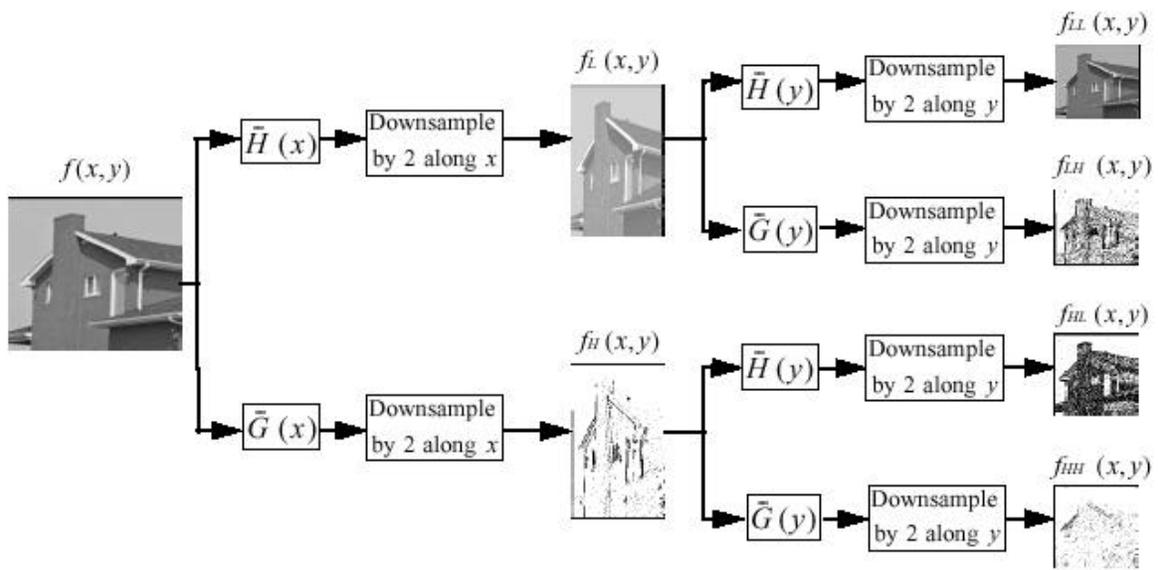
$$(G \cdot a)_k = \sum_n g(n - 2k) \cdot a_n$$

- Podemos descomponer una señal de entrada recursivamente utilizando ambos tipos de filtros:

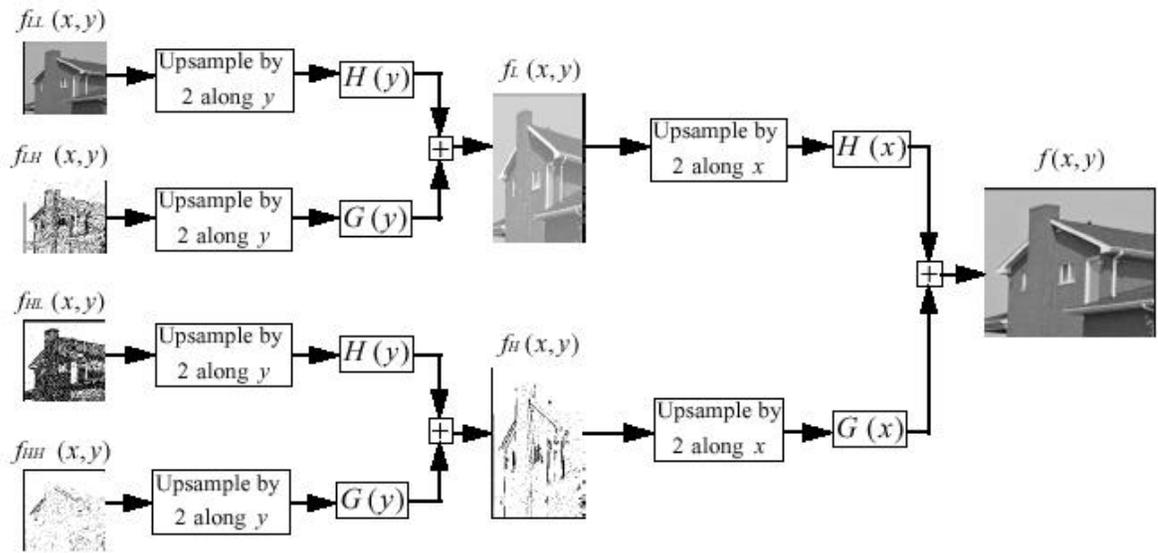


Ejemplos:

- Transformada Wavelet de una Imagen:



- Recuperación de una imagen a partir de su transformada Wavelet:

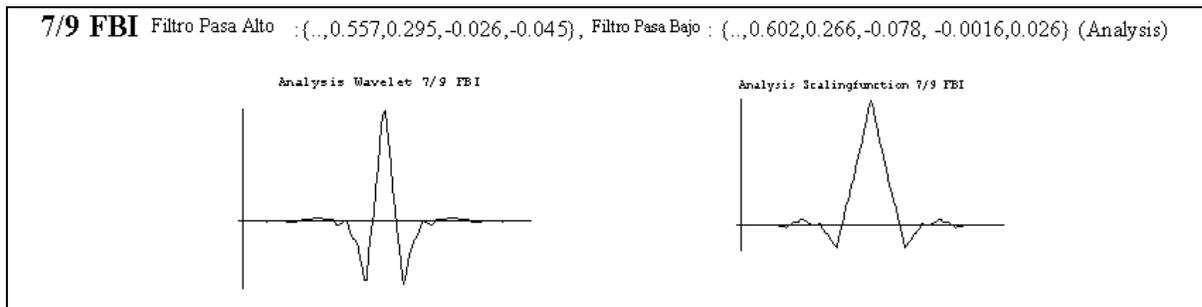


## 4.2.4. Aplicaciones

- Codificación de Huellas Dactilares (FBI)
- Compresión de Imágenes y Video
- Compresión de voz, audio y señales ECG
- Aplicaciones a sistemas de comunicación
- Análisis de Texturas
- Análisis Transiente de Señales (detección de fallas de máquinas eléctricas)
- Resolución Ecuaciones Diferenciales
- Representación, Interpolación de Curvas
- Otras

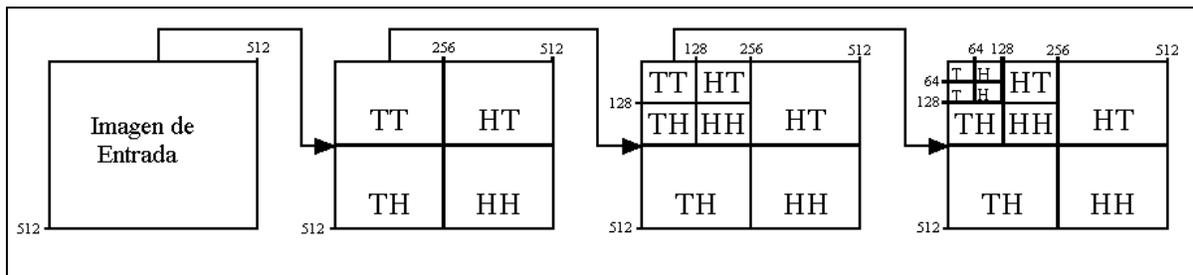
## Codificación de Huellas Dactilares

- A la fecha corresponde a la aplicación más importante de las wavelets
- El standard WSQ (Wavelet/scalar quantization) adoptado por el FBI (1993) para almacenar los 300 millones de huellas dactilares (30 millones de personas \* 10 dedos)
- La utilización de este standard se prefirió a la utilización de JPEG, permite además de almacenar la información en forma compactada, el realizar reconocimiento de huellas en diferentes resoluciones (mayor rapidez)



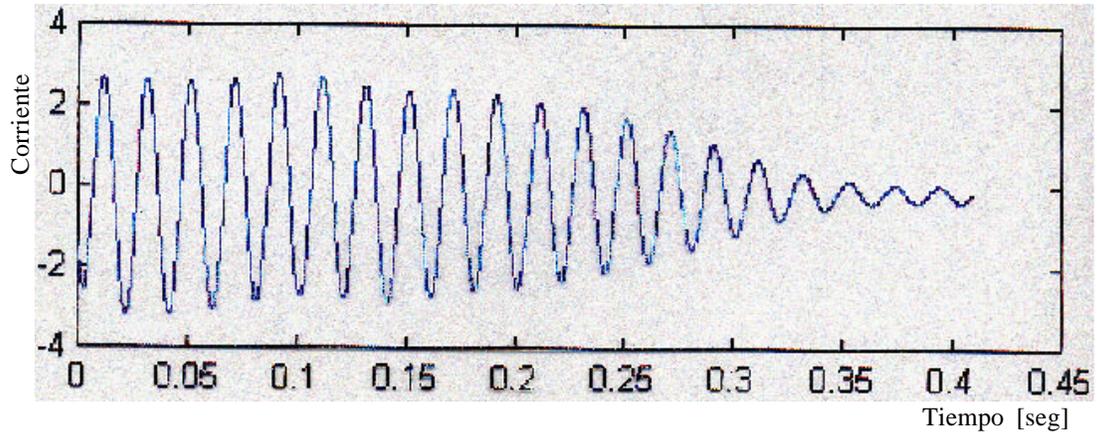
		$\pi/8$	$\pi/4$	$\pi/2$		
0	0 1					
	2 3	4	7	8	19	20
					23	24
$\pi/8$		5	6	9	10	21
					22	25
		11	12	15	16	27
					28	31
		13	14	17	18	29
					30	33
$\pi/4$		35	36	39	40	
		37	38	41	42	
		43	44	47	48	
$\pi/2$		45	46	49	50	51
						54
						55
		56				
			57			
				60		
					61	
		58				
			59			
				62		
					63	

Figura: La Subbanda WSQ

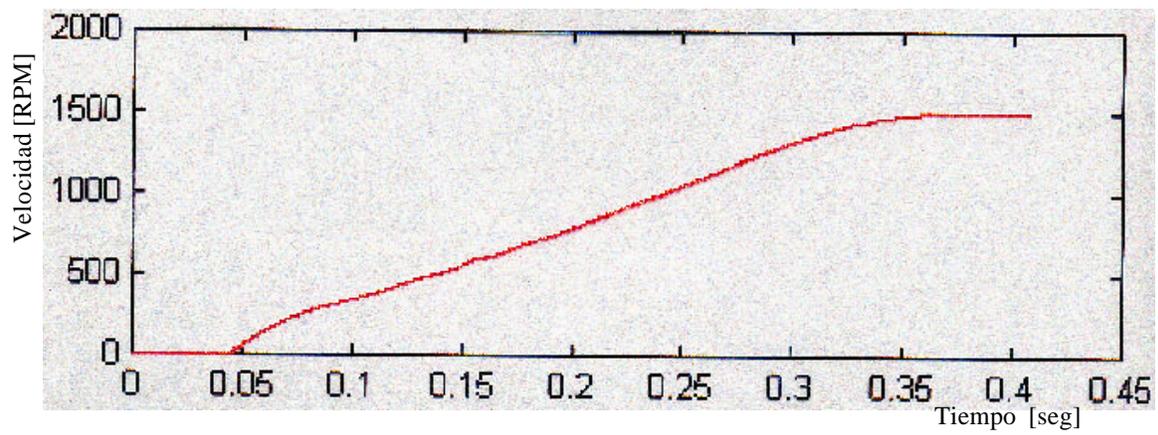


## Análisis de transientes en partida de Motores:

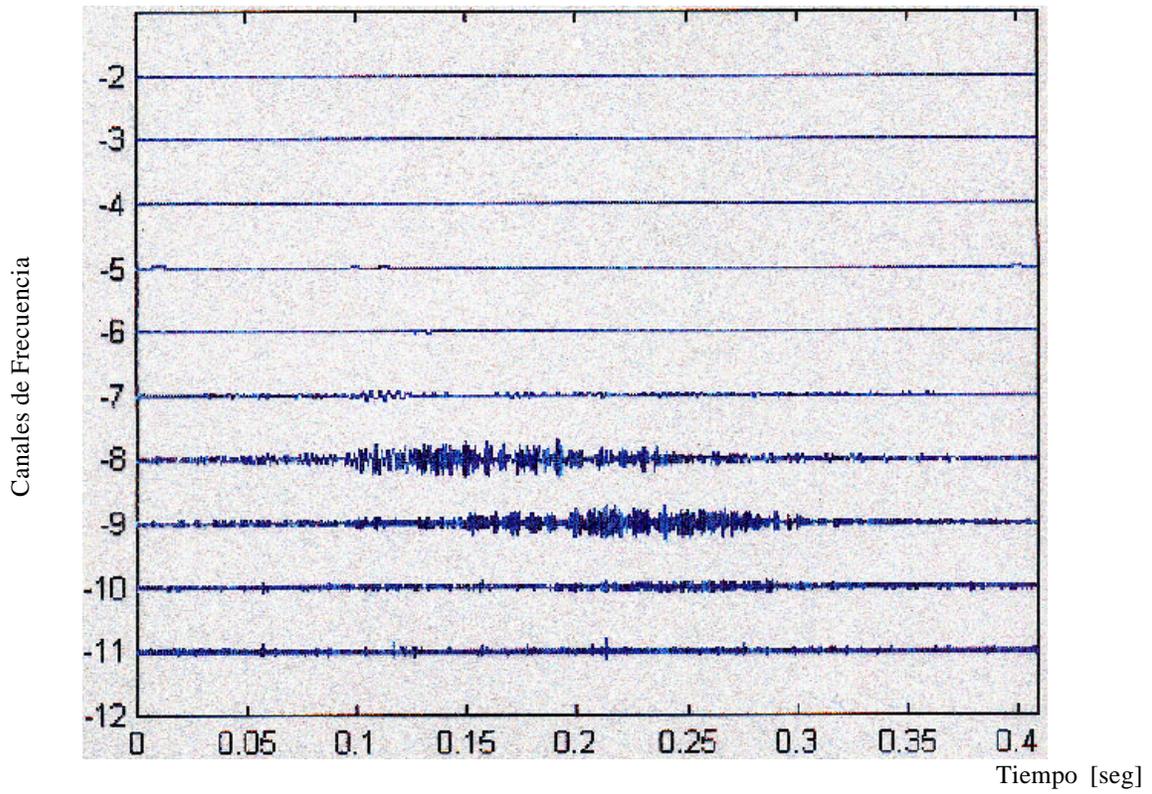
Corriente en la Partida de un Motor



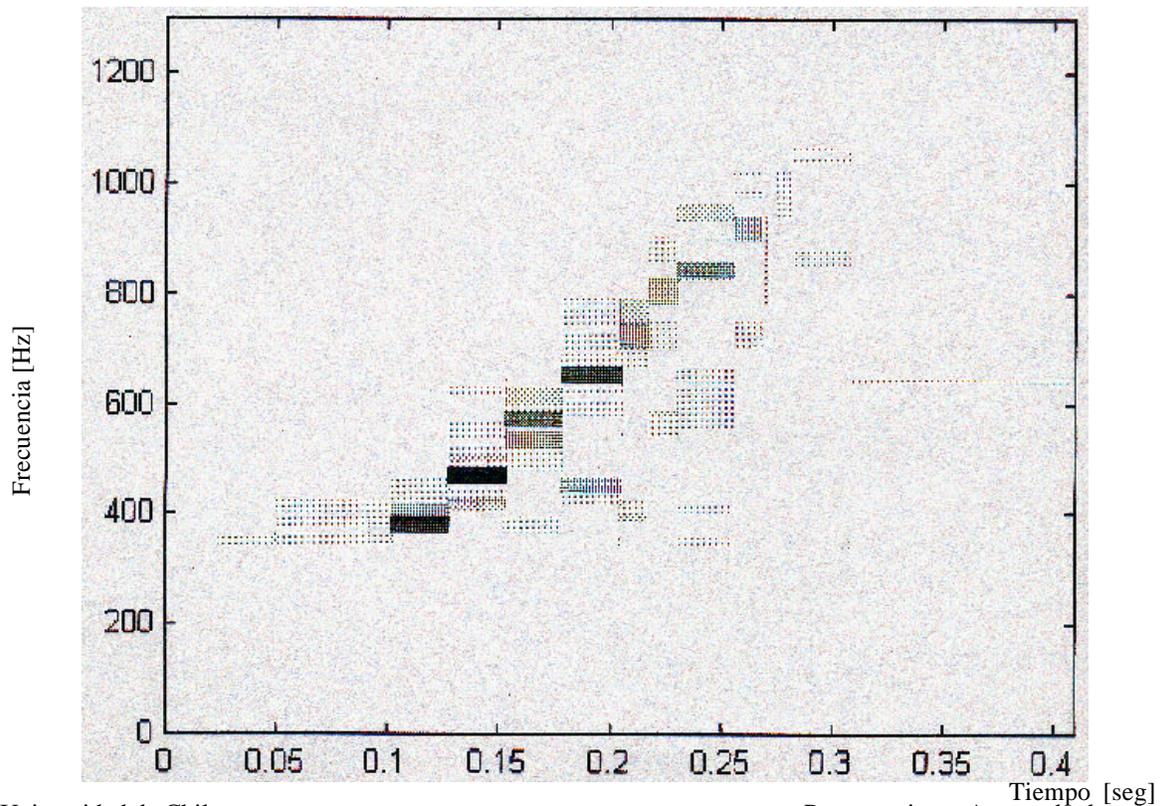
Velocidad en la Partida de un Motor



### Análisis Multiresolución de Corriente de Partida



### Análisis de la Corriente de Partida de un Motor

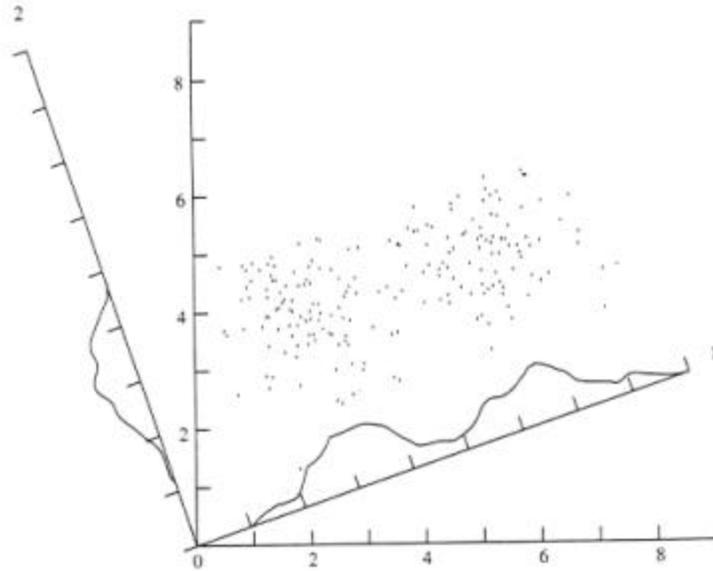


## 4.3. Análisis de Componentes Principales e Independientes

### 4.3.1. Análisis de Componentes Principales (PCA)

- El método de **Componentes Principales** corresponde a una poderosa herramienta de análisis de datos. Este tipo de análisis es conocido también como **Karhuven-Loève Transform (KLT)** y como **Hotelling Transform**.
- PCA fue desarrollado independientemente por diversos investigadores. Fue introducido originalmente por Pearson (1901) para realizar regresión lineal y en forma posterior redescubierto por Hotelling (1933). En el contexto de teoría de probabilidades fue desarrollado por Karhuven (1947) y luego generalizado por Loève (1963).
- En tareas de análisis de imágenes y en general de reconocimiento de patrones es muy importante la selección de un adecuado conjunto de características (features). Las características debieran no estar correlacionadas entre sí, de tal forma que cada una aportara nueva información para el proceso de clasificación. PCA se utiliza normalmente para reducir el conjunto inicial de características. Este conjunto es elegido normalmente en forma arbitraria o intuitiva, y sin conocer con exactitud la composición de éste, más adecuada a los datos de entrada.

- Dado un espacio  $n$ -dimensional de entrada (espacio de características), se desea encontrar  $m$  (nuevas) características (vectores de características de  $m$  componentes) que permiten representar el espacio de entrada, donde  $m < n$ . Dada la condición de que los vectores de características a encontrar sean ortogonales entre sí, el problema equivale a proyectar un espacio de dimensión  $n$  en uno de dimensión  $m$ , lo cual implica una reducción de dimensión. PCA es un método **lineal** para realizar esta reducción de dimensión, que mantiene la información intrínseca de los datos de entrada y reduce al máximo el error generado (en el sentido de error cuadrático medio).
- La idea central del método PCA consiste en realizar un cambio de coordenadas (la mencionada proyección) de tal forma que los nuevos ejes (ortogonales entre sí) correspondan a aquellas direcciones donde los datos de entrada presentan mayor varianza. Las proyecciones de los datos de entrada en los ejes del nuevo sistema de coordenadas corresponderán a los **componentes principales**. El primer componente principal se elige a lo largo de la dirección con máxima varianza. Este proceso se repite con todos los componentes.



Ej. Nube de puntos en 2D. Las densidades dibujadas se forman proyectando la nube en los ejes 1 y 2 indicados, correspondientes al nuevo sistema de coordenadas elegido (tomado de R. Linsker, 1988, con permiso del IEEE).

- En general se puede demostrar (ver Haykin, 1994) que los vectores propios de la **matriz de covarianza cruzada** corresponden a las direcciones de los componentes principales. De esta forma la dirección del k-ésimo componente principal corresponde al **vector propio** correspondiente al k-ésimo más grande **valor propio** de la matriz de covarianza. Si los vectores de entrada tienen media cero debe considerarse los vectores propios de la **matriz de correlación cruzada**. Si los vectores de entrada no tienen media cero, entonces el valor medio puede ser restado a cada uno de ellos.
- La correlación cruzada entre 2 características  $X_i$  y  $X_j$  corresponderá a  $C_{ij}$ . La varianza de una característica corresponderá a  $C_{ii}$ . La matriz de correlación cruzada (matriz de tipo simétrico) corresponderá a la matriz de

los  $C_{ij}$ , calculada como:  $\mathbf{C} = E\{\langle \mathbf{X} \cdot \mathbf{X}^T \rangle\}$  ( $\mathbf{X}$  corresponde a la secuencia de vectores de entrada), o explícitamente:

$$\mathbf{C} = \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{12} & C_{22} & & \vdots \\ \vdots & & \ddots & \\ C_{1n} & \dots & \dots & C_{nn} \end{bmatrix}$$

- Se desea que la correlación cruzada entre características sea cero, luego debe buscarse  $C_{ij} = 0$  ( $i \neq j$ ) mediante una diagonalización de  $\mathbf{C}$ , lo cual equivale al mencionado cambio de coordenadas. La diagonalización de  $\mathbf{C}$  se calcula como:

$$\mathbf{C}' = \mathbf{U}^T \cdot \mathbf{C} \cdot \mathbf{U}$$

$\mathbf{C}'$  es una matriz diagonal definida por los valores propios ( $\lambda$ ) de  $\mathbf{C}$ :

$$\mathbf{C}' = \begin{bmatrix} I_1 & 0 & \dots & 0 \\ 0 & I_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & I_n \end{bmatrix}$$

$\mathbf{U}$  es una matriz formada por los vectores propios (nuevas coordenadas)  $u_i$ , tal que:

$$\mathbf{u}_i^T \cdot \mathbf{u}_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}$$

- La proyección de los vectores de entrada en las nuevas coordenadas, i.e. los componentes principales, está determinada por:

$$a_j = \mathbf{u}_j^T \cdot \mathbf{x} = \mathbf{x}^T \cdot \mathbf{u}_j \quad j = 1, \dots, n$$

- La determinación de los  $a_j$  corresponde a una operación de **análisis**. La operación dual de **síntesis**, es decir, la expansión o representación de los vectores de entrada mediante los vectores propios está determinada por:

$$\mathbf{x} = \mathbf{U} \cdot \mathbf{a} = \sum_{j=1}^n a_j \mathbf{u}_j$$

(ésta es la Karhunen-Loève Transform – KLT ).

- Como ya fue mencionado uno de los atractivos mayores de PCA es la reducción de la dimensionalidad ( $m < n$ ). En este caso los datos originales serán reconstruidos o representados en forma aproximada mediante:

$$\mathbf{x}' = \sum_{j=1}^m a_j \mathbf{u}_j$$

y el error de la aproximación (perpendicular a  $\mathbf{x}'$ ) estará determinado por:

$$\mathbf{e} = \sum_{j=m+1}^n a_j \mathbf{u}_j$$

esta representación truncada es muy utilizada en aplicaciones de compresión de datos.

- Entre las aplicaciones más interesantes derivada de PCA se encuentra el método de **Eigenfaces** utilizado en reconocimiento de caras.
- Existen implementaciones de PCA utilizando redes neuronales, que se basan en aprendizaje de tipo Hebbiano. De hecho, el procesamiento de los datos realizado por la red SOM (Red de Kohonen) puede verse como una suerte de PCA.

### 4.3.2. Análisis de Componentes Independientes (ICA)

- El principal objetivo de PCA es reducir la dimensión de los datos de entrada, de tal forma que la representación resultante sea lo más parecida posible a la original (en el sentido de error cuadrático medio) y esté formada por componentes no correlacionados. El principal objetivo de ICA no es reducir la dimensión de los datos de entrada, sino que encontrar una transformación cuyos componentes sean tan estadísticamente independientes como sea posible. Para datos sin distribución Gaussiana, éste requerimiento es mucho más fuertes que el de no correlación.
- ICA se basa en el uso de PCA no lineales.

#### Non-linear PCA

- Derivado de PCA existen desarrollados posteriores conocidos como **non-linear PCA**, implementados generalmente en base a redes neuronales (mayor eficiencia, no existen “closed-forms”). Las principales ventajas de los PCA no lineales son:
  1. El mapeo entre los espacios de entrada y salida es no-lineal.
  2. A diferencia de PCA que toma en cuenta estadísticas de segundo orden en su análisis, non-linear PCA puede usar estadísticas de orden superior. Esto es muy importante para aplicaciones del tipo “blind signal separation”. Las estadísticas de segundo orden son suficientes solo para caracterizar

completamente datos con distribución Gaussiana, y para aplicaciones de procesamiento lineales.

- Non-linear PCA tiene aplicaciones en al menos las siguientes áreas:
  1. PCA robusto. Usando no-linealidades que crezcan en forma más lenta que lineal, se produce mayor robustez en el análisis y se puede lidiar con ruido no-gaussiano.
  2. Blind signal processing e ICA. Estos métodos tienen aplicaciones en comunicaciones, procesamiento de voz, ingeniería biomédica, etc.
  3. Clustering de datos.

### **ICA: definición y conceptos básicos**

- Como ya fue señalado ICA es un método de análisis estadístico cuyo objetivo es expresar un conjunto de variables o señales aleatorias como combinaciones de variables estadísticamente independientes.
- Sean  $n$  variables escalares aleatorias  $x(1), \dots, x(n)$ , que se asume son combinaciones lineales de  $m$  componentes desconocidos  $s(1), \dots, s(m)$  con media cero y estadísticamente independientes:

$$p(s(1), s(2), \dots, s(m)) = p(s(1)) \cdot p(s(2)) \cdots p(s(m))$$

se asume que  $m=n$ , es decir que existen a los más tantos componentes  $s(i)$  como mezclas  $x(j)$ . Además de esto, lo único que se asume es que los componentes  $s(i)$  son procesos estocásticos estacionarios con media cero,

y que solo uno de los puede tener una distribución de probabilidad Gaussiana.

1. La mezcla lineal está dada por  $\mathbf{x}=\mathbf{A}\cdot\mathbf{s}$ , donde  $\mathbf{A}$  es una matriz de dimensión  $m \times n$  desconocida, llamada la **matriz de mezcla** (mixing matrix).
2. El problema de encontrar  $\mathbf{A}$  puede ser simplificado realizando un *sphering* o *prewhitening* de los datos de entrada (la matriz de covarianza se transforma en la matriz unitaria). De esta forma el vector observado  $\mathbf{x}$  es transformado en el vector  $\mathbf{v}=\mathbf{V}\cdot\mathbf{x}$ , tal que los elementos  $v(i)$  son mutuamente no correlacionados y tienen varianza unitaria. La dimensión de los datos debe también ser reducida de tal forma que el vector transformado  $\mathbf{v}$  tenga dimensión  $m$ , i.e. el número de componentes independientes. Ambas transformaciones puede ser realizadas usando PCA.
3. Luego de la transformación tenemos

$$\mathbf{v}=\mathbf{V}\cdot\mathbf{x}=\mathbf{V}\cdot\mathbf{A}\cdot\mathbf{s}=\mathbf{B}\cdot\mathbf{s}$$

donde  $\mathbf{B}$  es una matriz ortogonal ( $\mathbf{B}\cdot\mathbf{B}^T=\mathbf{I}$ ), conocida como la **matriz de separación** (separating matrix). Es decir, se ha transformado el problema de encontrar una matriz arbitraria  $\mathbf{A}$  de rango completo (full-rank), en uno más simple de encontrar una matriz ortogonal  $\mathbf{B}$ . Luego los componentes buscados  $s(i)$  pueden ser encontrados como:

$$\mathbf{s}=\mathbf{B}^T\cdot\mathbf{v}$$

- El problema es que al no conocerse la matriz  $\mathbf{B}$ , los  $s(i)$  pueden encontrarse solo en forma aproximada. Llamaremos a los componentes aproximados

$y(i)$ . Estos componentes pueden encontrarse usando non-linear PCA. Una posible implementación neural es:

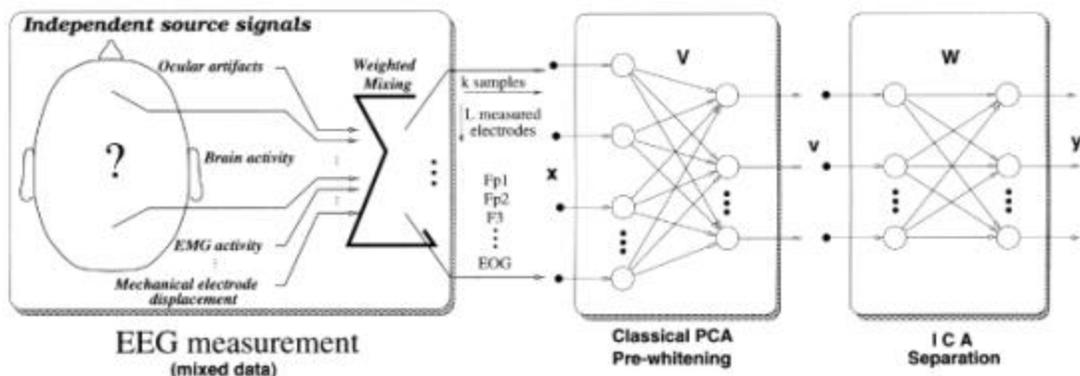
$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mathbf{m}_k [\mathbf{v}_k - \mathbf{W}_k \cdot \mathbf{g}(\mathbf{y}_k)] \mathbf{g}(\mathbf{y}_k^T)$$

donde  $\mathbf{m}_k$  es la tasa de aprendizaje;  $g(\cdot)$  es una función no-lineal adecuada (ej. tangente hiperbólica); e  $\mathbf{y}_k$  es la salida de una red neuronal cuyos pesos sinápticos están dados por la matriz  $\mathbf{W}_k$ .

### 4.3.3. Aplicaciones de ICA

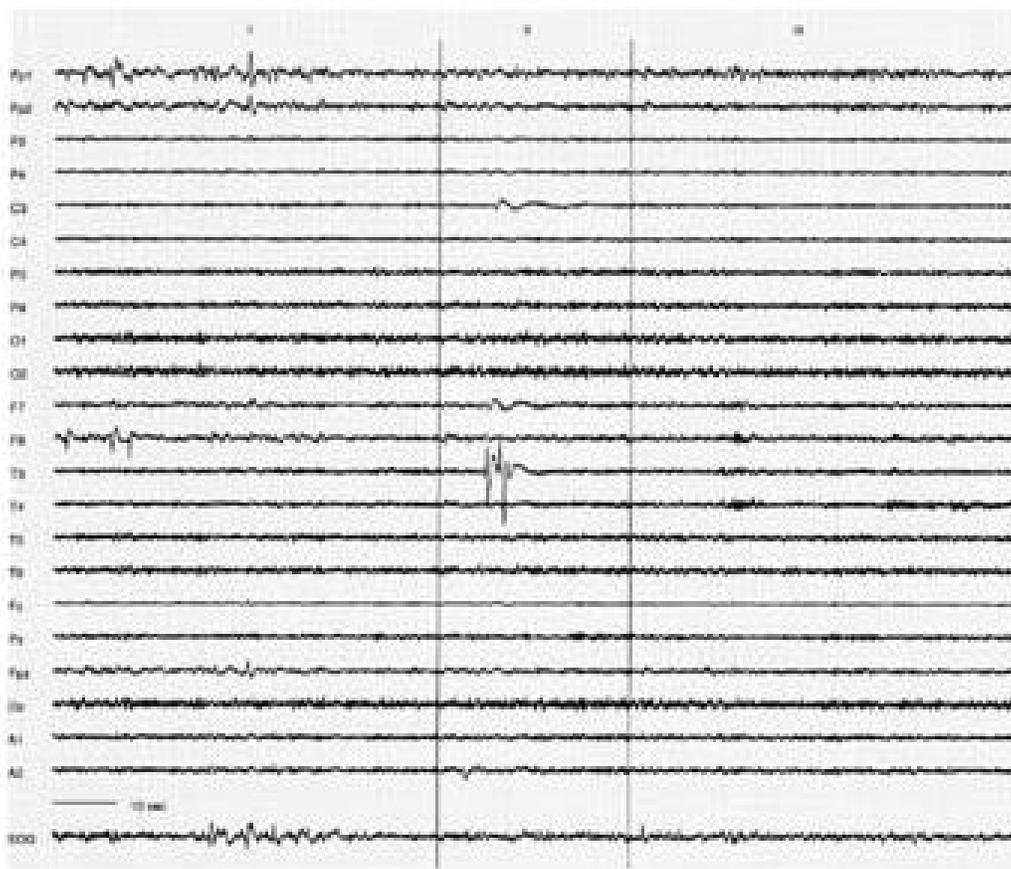
#### Análisis de EEG mediante ICA

- Frecuentemente al tomar un electroencefalograma (EEG) se producen perturbaciones producidas por factores externos tales como movimiento ocular, actividad muscular, desplazamientos mecánicos de los electrodos, etc.
- En la Universidad de Finlandia (Prof. Oja) se propuso el siguiente sistema en base a ICA para análisis de EEG:

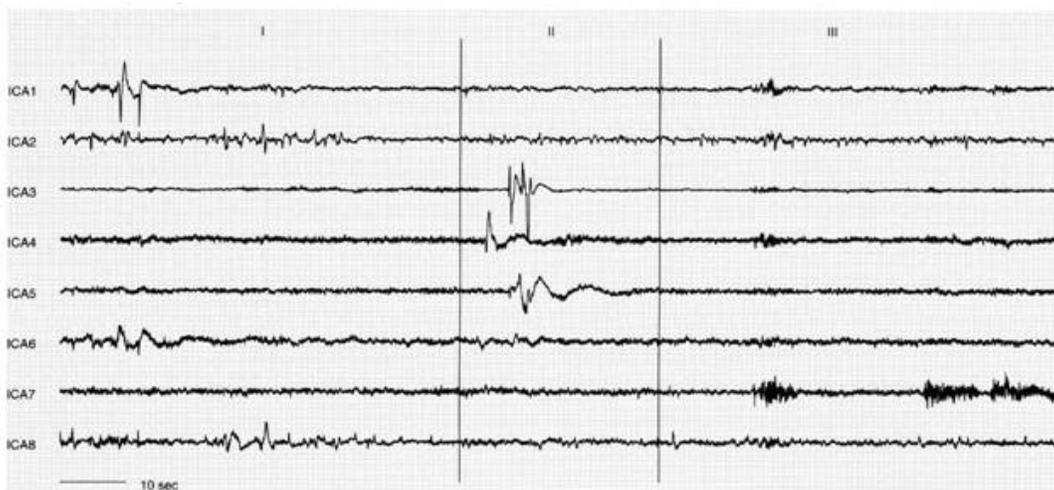


- El sistema funciona bajo el supuesto de que la actividad cerebral, de los ojos, y otras (muscular) son procesos separados, y cuya independencia se ve reflejada en sus estadísticas.
- Como ejemplo del funcionamiento del sistema se muestra el análisis de un EEG de 22 canales perteneciente a un niño, y la salida con 8 componentes independientes. Cabe destacar la salida del cuarto componentes (ICA 4), donde una señal con probable significado fisiológico es detectada (en el EEG original no aparece).

- 22 canales de EEG de un niño

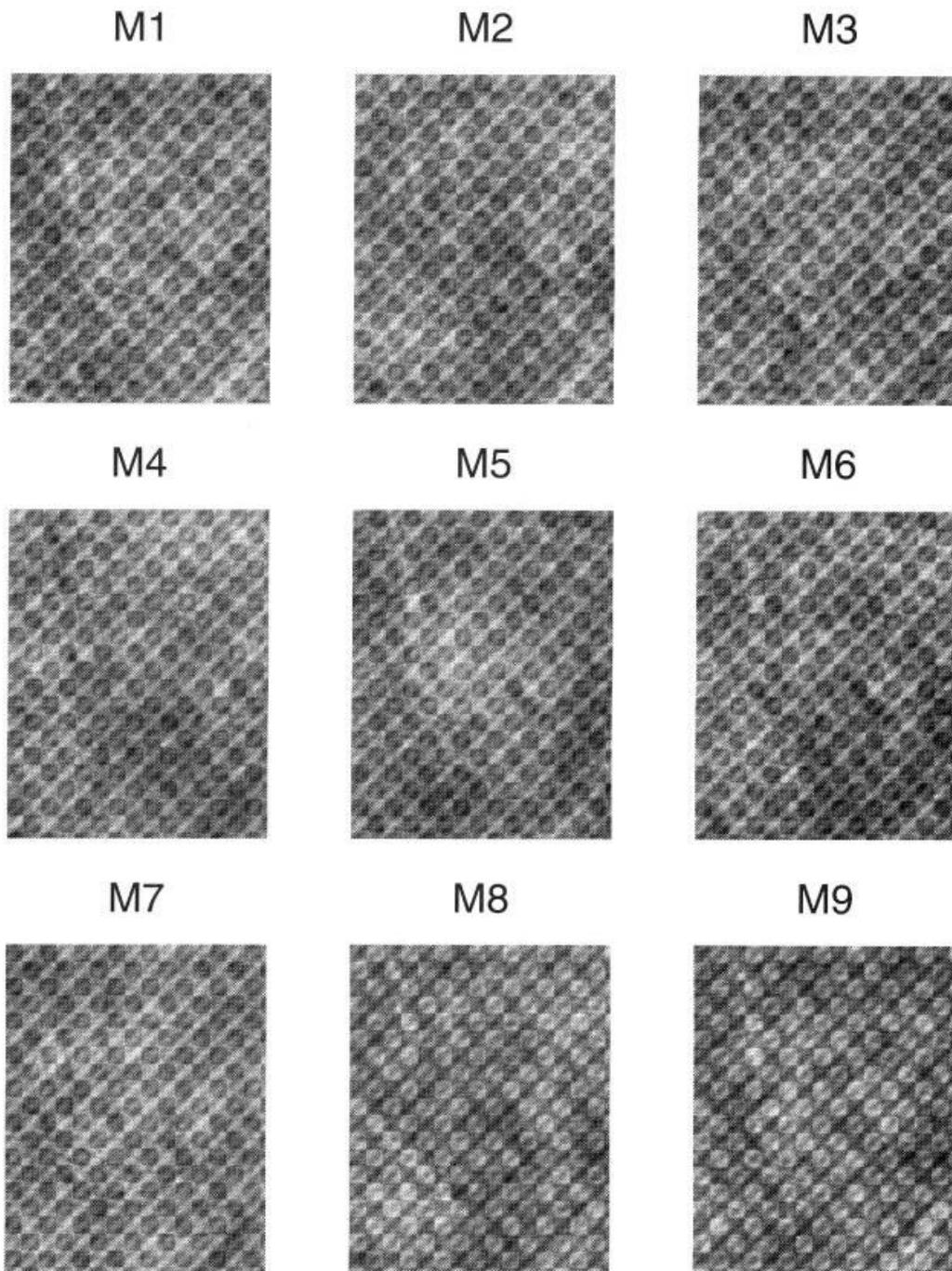


- Los primeros 8 componentes independientes (ICA1-ICA8)



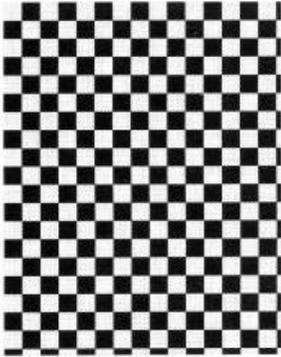
## Blind Signal Separation con ICA

- 9 mezclas de 9 imágenes originales

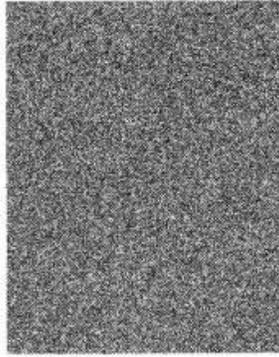


- Imágenes recuperadas a partir de las 9 imágenes mezcladas

$-(W+NPCA1)$



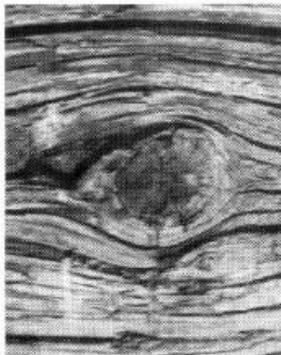
$W+NPCA2$



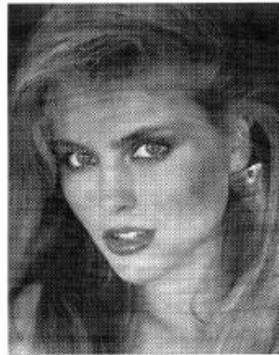
$-(W+NPCA3)$



$-(W+NPCA4)$



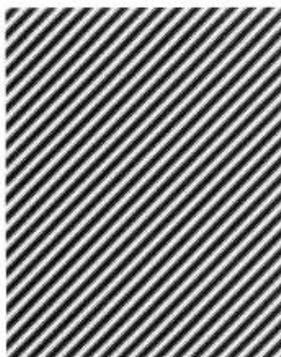
$W+NPCA5$



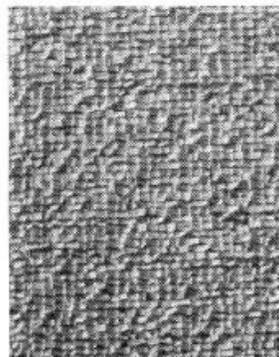
$W+NPCA6$



$W+NPCA7$



$W+NPCA8$



$W+NPCA9$

