UNIVERSIDAD ADOLFO IBÁÑEZ
FACULTAD DE INGENIERÍA Y CIENCIAS

iUAI
UNIVERSIDAD ADOLFO IBÁÑEZ
FACULTAD DE INGENIERÍA Y CIENCIAS

MDS^{2019} PROCESAMIENTO DE IMÁGENES

MULTI RESOLUCIÓN

Miguel Carrasco
miguel.carrasco@uai.cl
1er Semestre 2020

Temario

- Multiresolución y Compresión
  - Transformada de Haar

Multiresolución y codificación

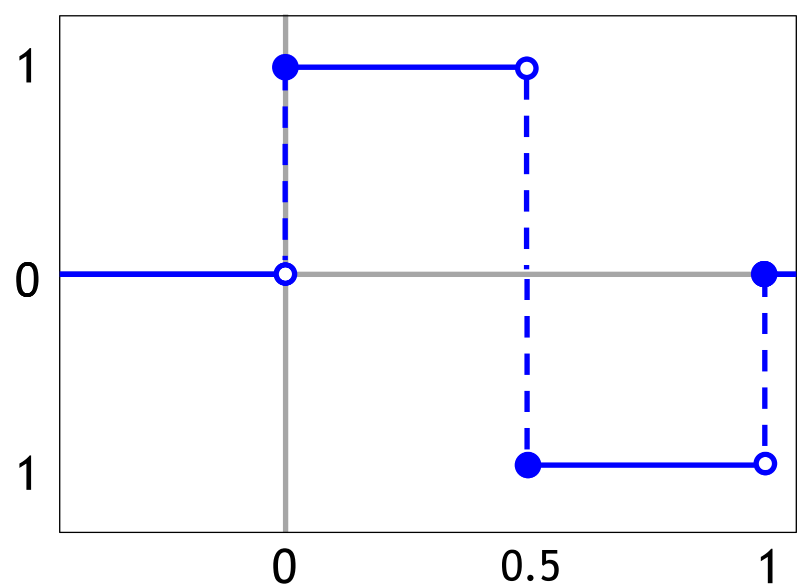
- Transformada de Haar

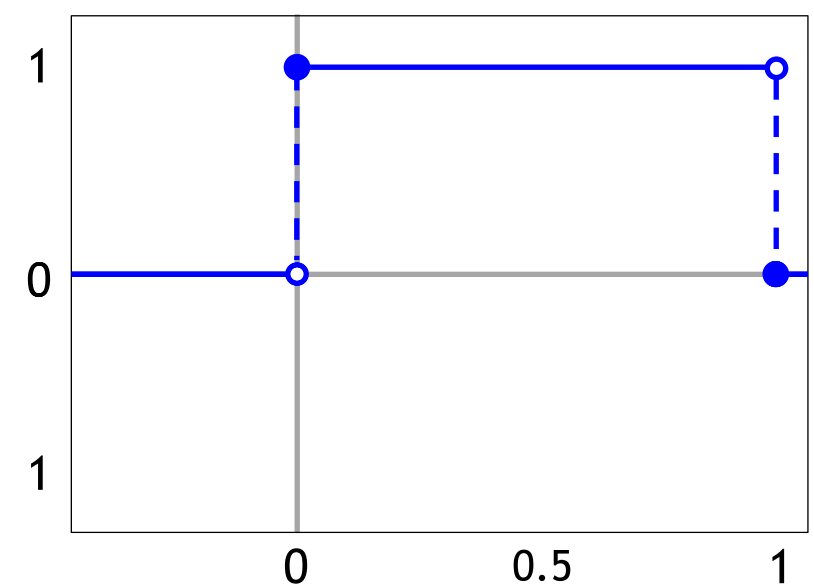
  - La transformada de Haar fue propuesta por Alfred Haar (Húngaro) en 1909. La función de Haar empleada tiempo después dentro del ámbito de las señales de Wavelets, extremadamente útiles en la compresión de imágenes.

  - La función madre (o generadora) corresponde a:

$$\psi(t)=\begin{cases} 1 & 0 \leq t < 0.5, \\ -1 & 0.5 \leq t < 1, \\ 0 & \text{otro caso.} \end{cases}$$

$$\phi(t)=\begin{cases} 1 & 0 \leq t < 1, \\ 0 & \text{otro caso.} \end{cases}$$



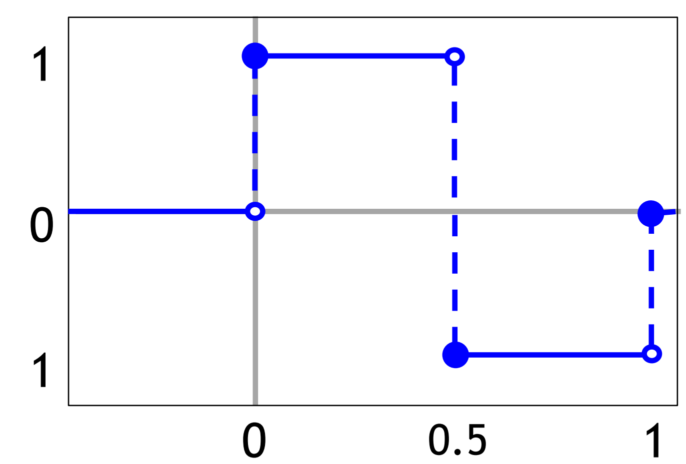


Multiresolución y codificación

- Transformada de Haar
  - La propieda más importante de la función de Haar indica que cualquier función real puede ser expresada por combinaciones lineales de la función de Haar. En el presente, empleamos las funciones de Wavelets de Haar siendo esta la versión más simple.
  - En términos funcionales,

$$\left\{ t \mapsto \psi_{n,k}(t) = \underbrace{\psi\left(2^n t - k\right)}; n \in \mathrm{N}, 0 \leq k \leq 2^n \right\}$$

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 0.5, \\ -1 & 0.5 \leq t < 1, \\ 0 & \text{otro caso.} \end{cases}$$



Multiresolución y codificación

- Transformada de Haar
  - Analicemos en detalle la ecuación para diferentes k y n

$$\boxed{\psi_{n,k}(t) = \psi\left(2^n t - k\right)}$$

$$\psi_{0,0}(t) = \psi\left(2^0 t - 0\right) \longrightarrow = \psi(t) \longrightarrow \boxed{\psi(t) = \begin{cases} 1 & 0 \le t < 0.5, \\ -1 & 0.5 \le t < 1, \\ 0 & \text{otro caso.} \end{cases}}$$

Este es el caso más simple

$$\psi_{1,0}(t) = \psi\left(2^1 t - 0\right) \longrightarrow = \psi(2t)$$

$$\psi_{1,1}(t) = \psi\left(2^1 t - 1\right) \longrightarrow = \psi(2t - 1)$$

$$\psi_{2,0}(t) = \psi\left(2^2 t\right) \longrightarrow = \psi(4t)$$

$$\psi_{2,1}(t) = \psi\left(2^2 t - 1\right) \longrightarrow = \psi(4t - 1)$$

$$\psi_{2,2}(t) = \psi\left(2^2 t - 2\right) \longrightarrow = \psi(4t - 2)$$

$$\psi_{2,3}(t) = \psi\left(2^2 t - 3\right) \longrightarrow = \psi(4t - 3)$$

¿Cómo son las funciones en los otros casos?

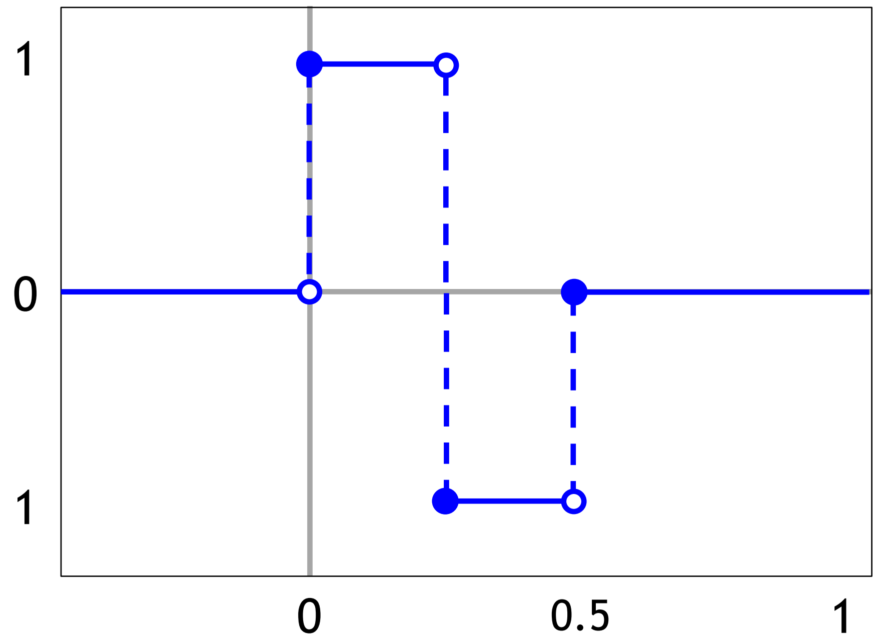
Multiresolución y codificación

- Transformada de Haar
  - Analicemos el segundo caso: $\psi_{1,0}(t) = \psi(2t)$

$$\psi(t) = \begin{cases} 1 & 0 \le t < 0.5, \\ -1 & 0.5 \le t < 1, \\ 0 & \text{otro caso.} \end{cases}$$

$$\psi(2t) = \begin{cases} 1 & 0 \le 2t < 0.5, \\ -1 & 0.5 \le 2t < 1, \\ 0 & \text{otro caso.} \end{cases}$$

$$\psi(2t) = \begin{cases} 1 & 0 \le t < 0.25, \\ -1 & 0.25 \le t < 0.5, \\ 0 & \text{otro caso.} \end{cases}$$



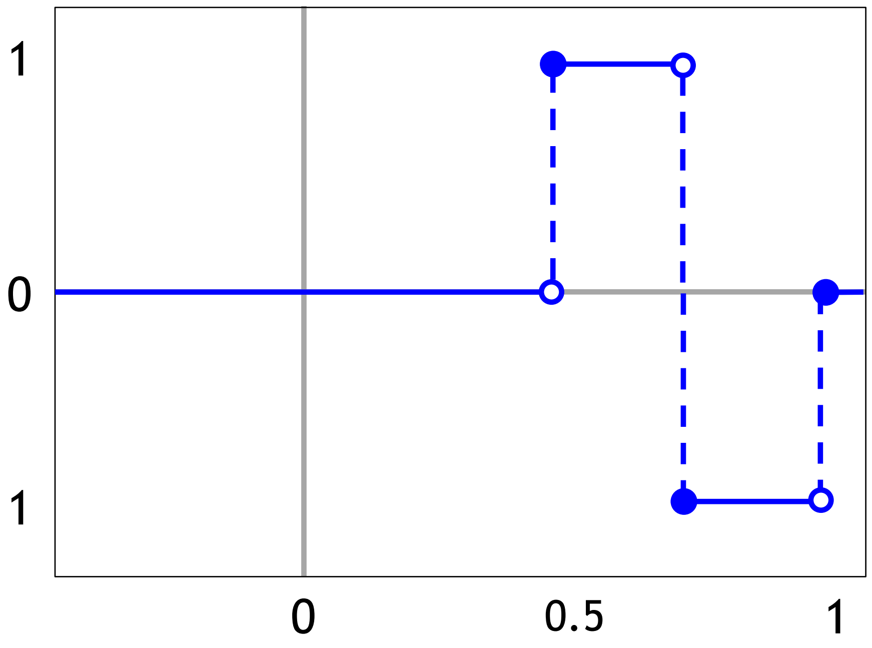
Multiresolución y codificación

- Transformada de Haar
  - Analicemos el tercer caso: $\psi_{1,1}(t) = \psi(2^1 t - 1)$

$$\psi(t) = \begin{cases} 1 & 0 \le t < 0.5, \\ -1 & 0.5 \le t < 1, \\ 0 & \text{otro caso.} \end{cases}$$

$$\psi(2t-1) = \begin{cases} 1 & 0 \le 2t-1 < 0.5, \\ -1 & 0.5 \le 2t-1 < 1, \\ 0 & \text{otro caso.} \end{cases}$$

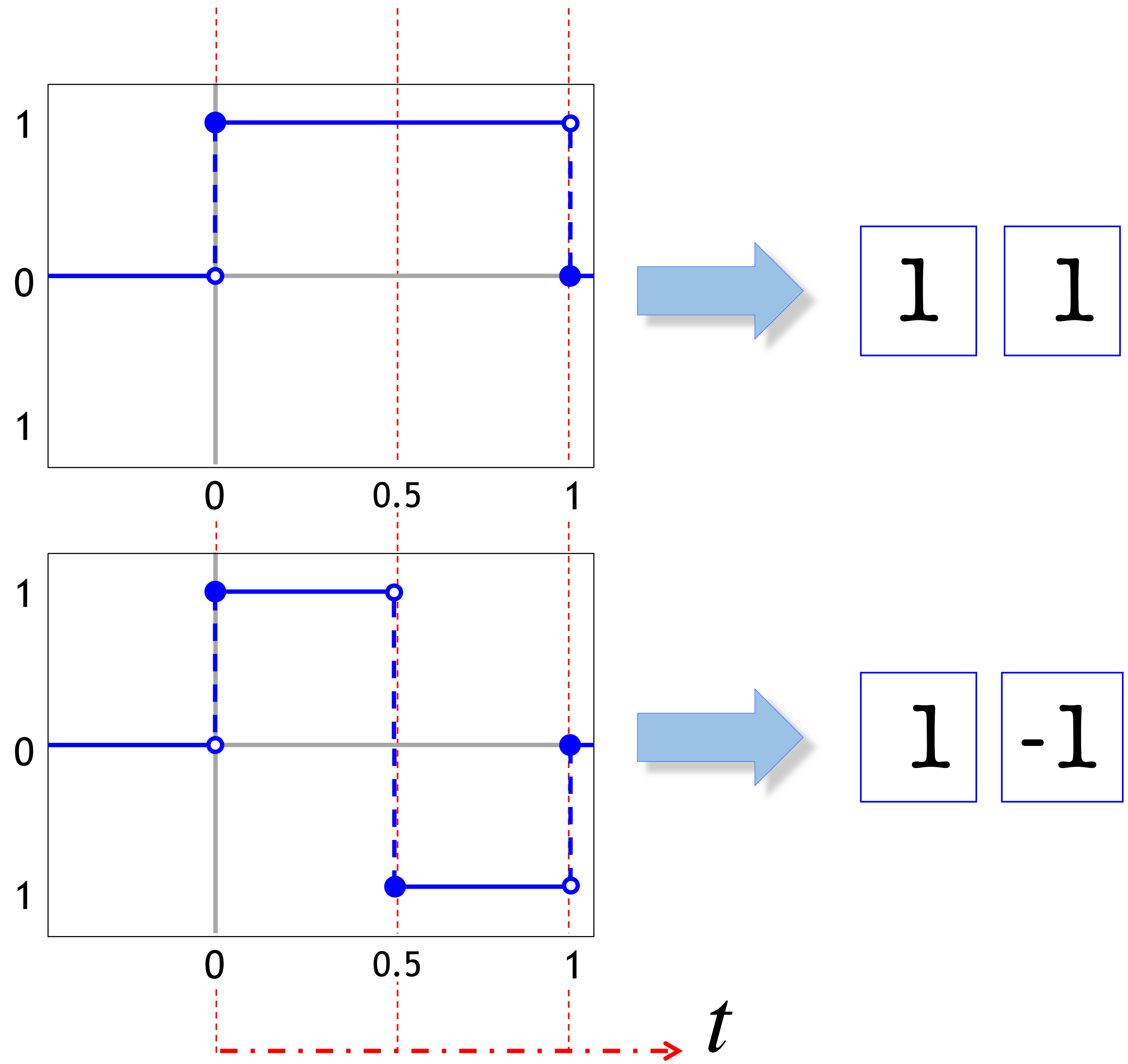
$$\psi(2t-1) = \begin{cases} 1 & 0.5 \le t < 0.75, \\ -1 & 0.75 \le t < 1, \\ 0 & \text{otro caso.} \end{cases}$$



- Transformada de Haar
  - Estos resultados pueden ser expresados matricialmente

$$\phi(t) =$$

$$\psi_{0,0}(t) =$$



$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

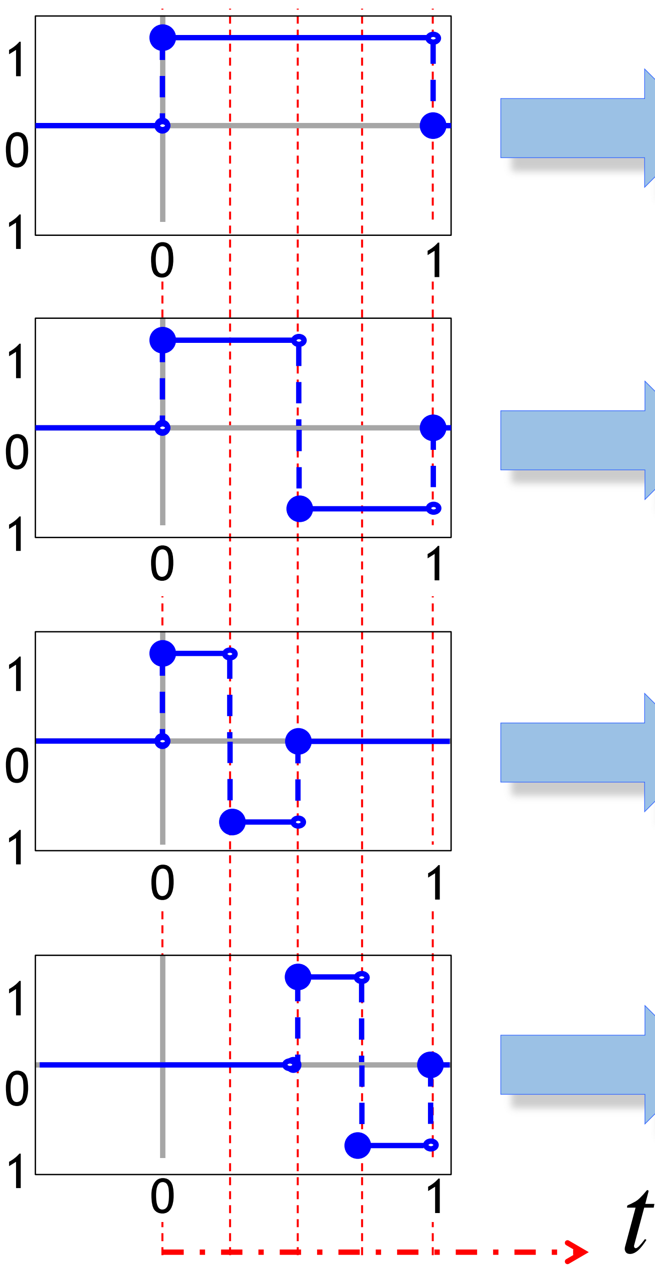
- Transformada de Haar
  - Veamos otras matrices calculadas con dintintas bases

$\phi(t) =$

$\psi_{0,0}(t) =$

$\psi_{1,0}(t) =$

$\psi_{1,1}(t) =$



1	1	1	1
1	1	-1	-1
1	-1	0	0
0	0	1	-1

$$H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Multiresolución y codificación

- Transformada de Haar
  - Veamos otras matrices calculadas con dintintas bases

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

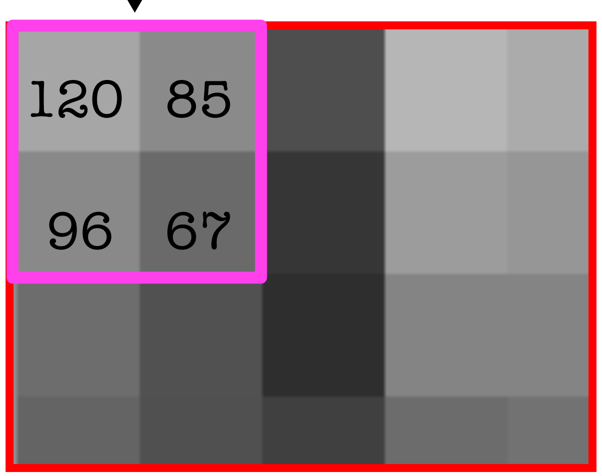
$$H_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

Multiresolución y codificación

- Transformada de Haar
  - Cómo aplicamos la transformada de Haar a una imagen.
  - Simplemente tomamos una ventana según el tamaño de la matriz y realizamos la siguiente operación

Ventana

$$G = H \cdot V \cdot H^{T}$$



$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Ya que ocupamos una matriz de segundo orden, empleamos una máscara de 2x2

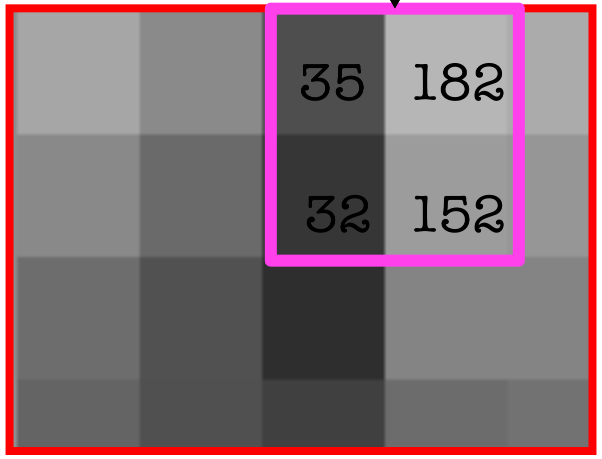
$$G = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} 120 & 85 \\ 96 & 67 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$G = \begin{bmatrix} 368 & 64 \\ 42 & 6 \end{bmatrix}$$

- Transformada de Haar
  - Cómo aplicamos la transformada de Haar a una imagen.
  - Simplemente tomamos una ventana según el tamaño de la matriz y realizamos la siguiente operación

Ventana

$$G = H \cdot V \cdot H^{\mathrm{T}}$$



$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Ya que ocupamos una matriz de segundo orden, empleamos una máscara de 2x2

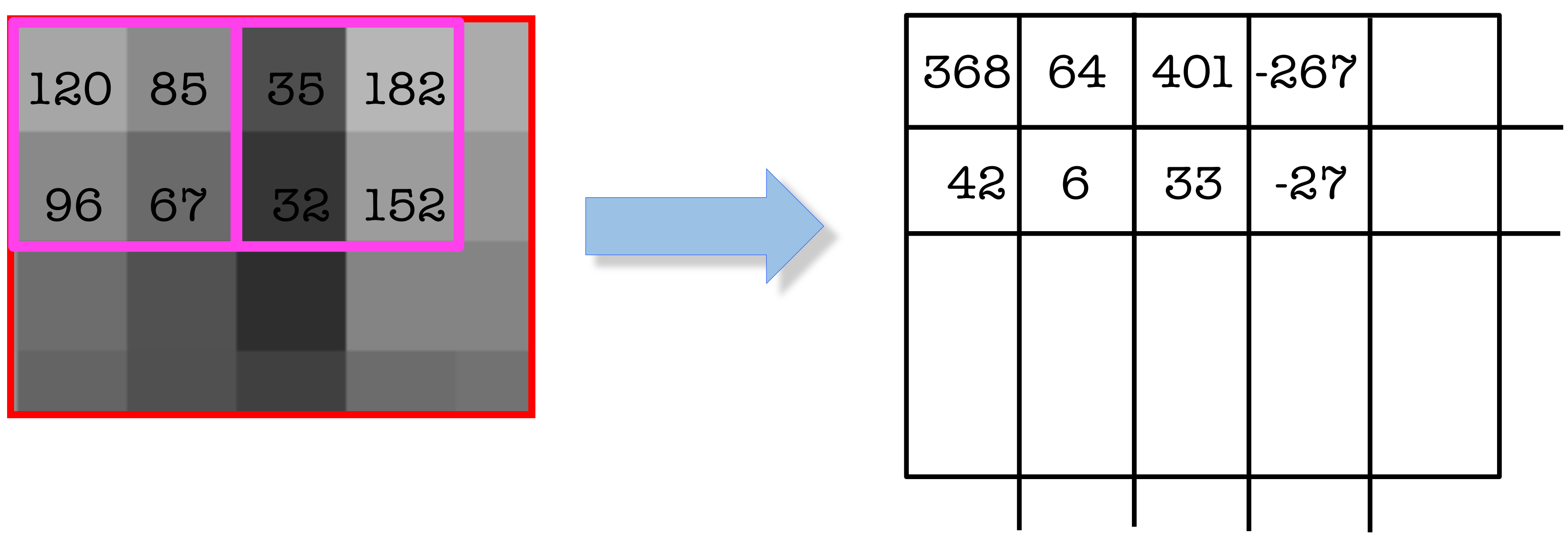
$$G = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} 35 & 182 \\ 32 & 152 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$G = \begin{bmatrix} 401 & -267 \\ 33 & -27 \end{bmatrix}$$

- Transformada de Haar
  - A medida que el bloque de la máscara se mueve en la imagen, vamos creando una nueva imagen que corresponde a la transformación de Haar

Imagen original

Imagen transformada

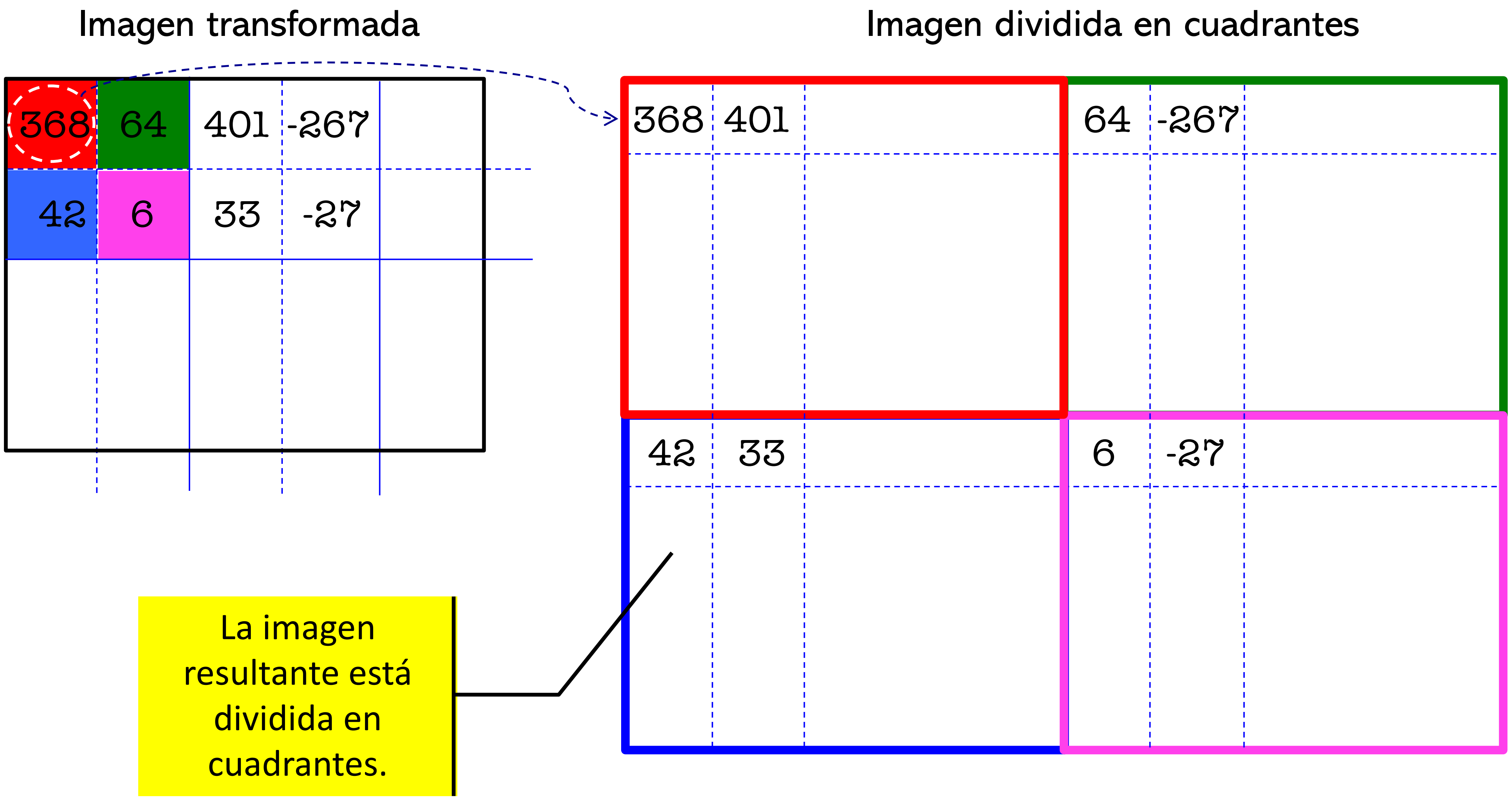


- Transformada de Haar
  - Podemos visualizar en parte el imagen original en el fondo. Esto es porque realmente se encuentra repartida en la imagen. Para ello debemos recomponerla en cuadrantes.

Imagen original

Imagen generada con transformada Haar

- Transformada de Haar
  - A medida que el bloque de la máscara se mueve en la imagen, vamos creando una nueva imagen que corresponde a la ***transformación de Haar***



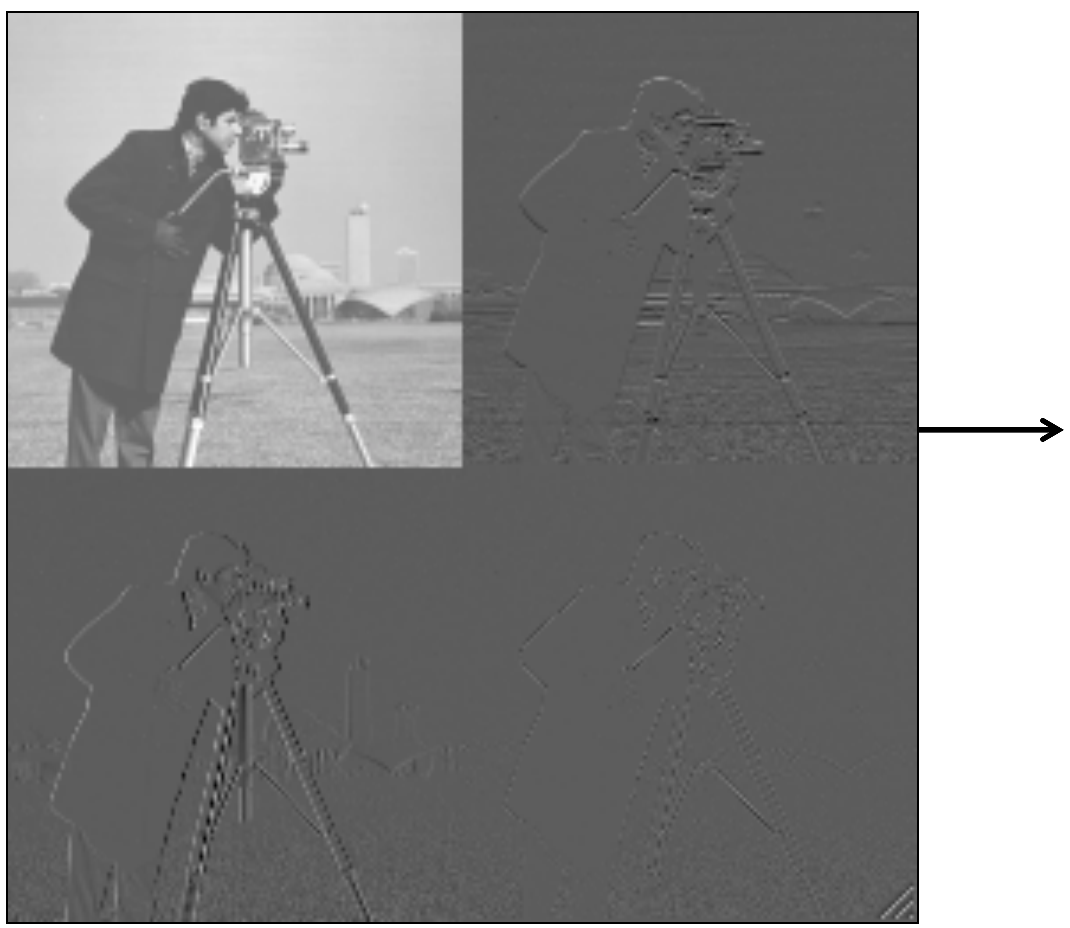
- Transformada de Haar
  - Podemos visualizar en parte el imagen original en el fondo. Esto es porque realmente se encuentra repartida en la imagen. Para ello debemos **recomponerla en cuadrantes.**

Imagen generada con transformada Haar

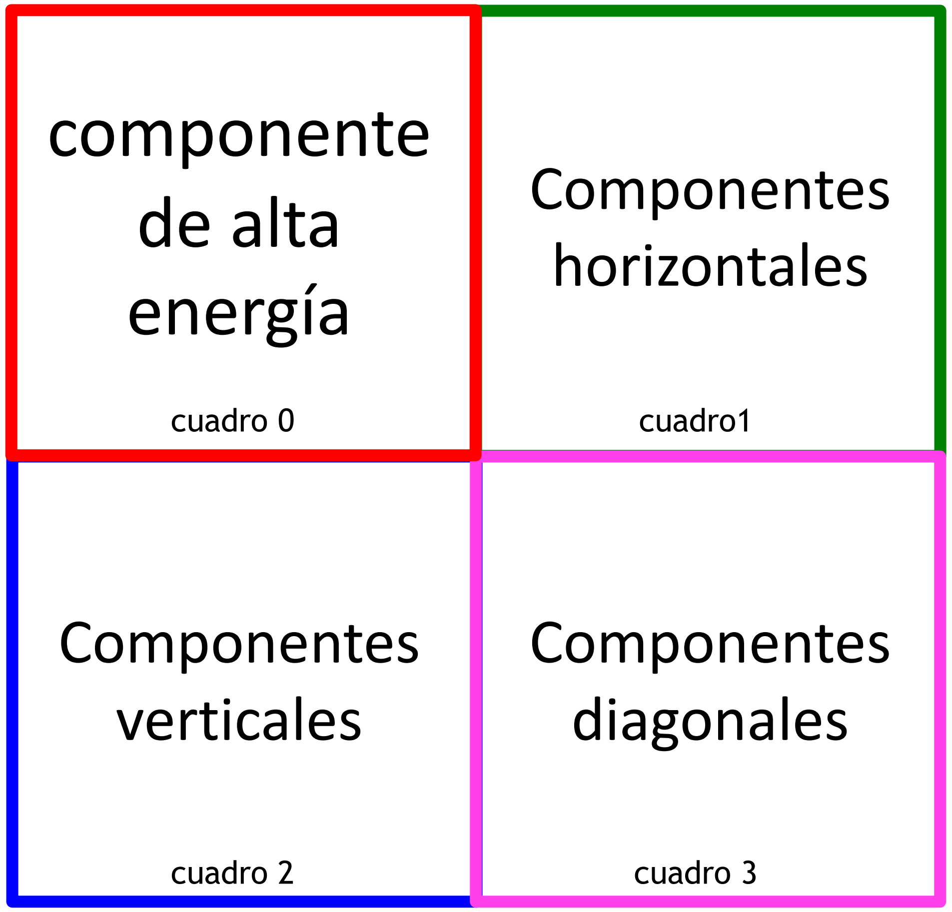
Imagen reordenada con transformada Haar

- Transformada de Haar
  - ¿Qué información hay en cada cuadrante? La información de cada cuadrante permite inferir información sobre la imagen y además filtrar cierto tipo de información.

Imagen reordenada con transformada Haar



Cuadrantes de Haar



- Transformada de Haar
  - Y cómo podemos volver hacia atrás y restaurar la imagen? Simplemente debemos reconstruir la transformada de haar original empleando las posiciones de los píxeles que antes agrupamos

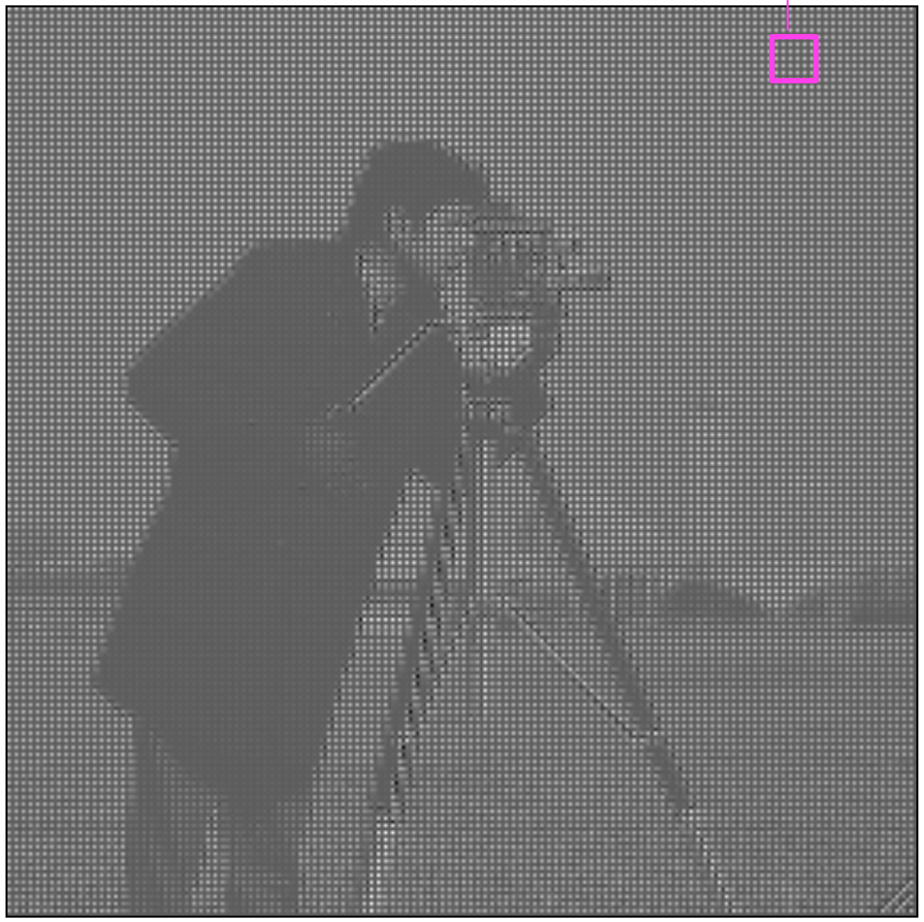
Imagen reordenada con transformada Haar

Imagen generada con transformada Haar

- Transformada de Haar (proceso inverso)
  - Para reconstruir, debemos emplear el proceso inverso, es decir, empleando la ecuación de Haar 2D en sentido inverso.

Ventana

Imagen generada con transformada Haar

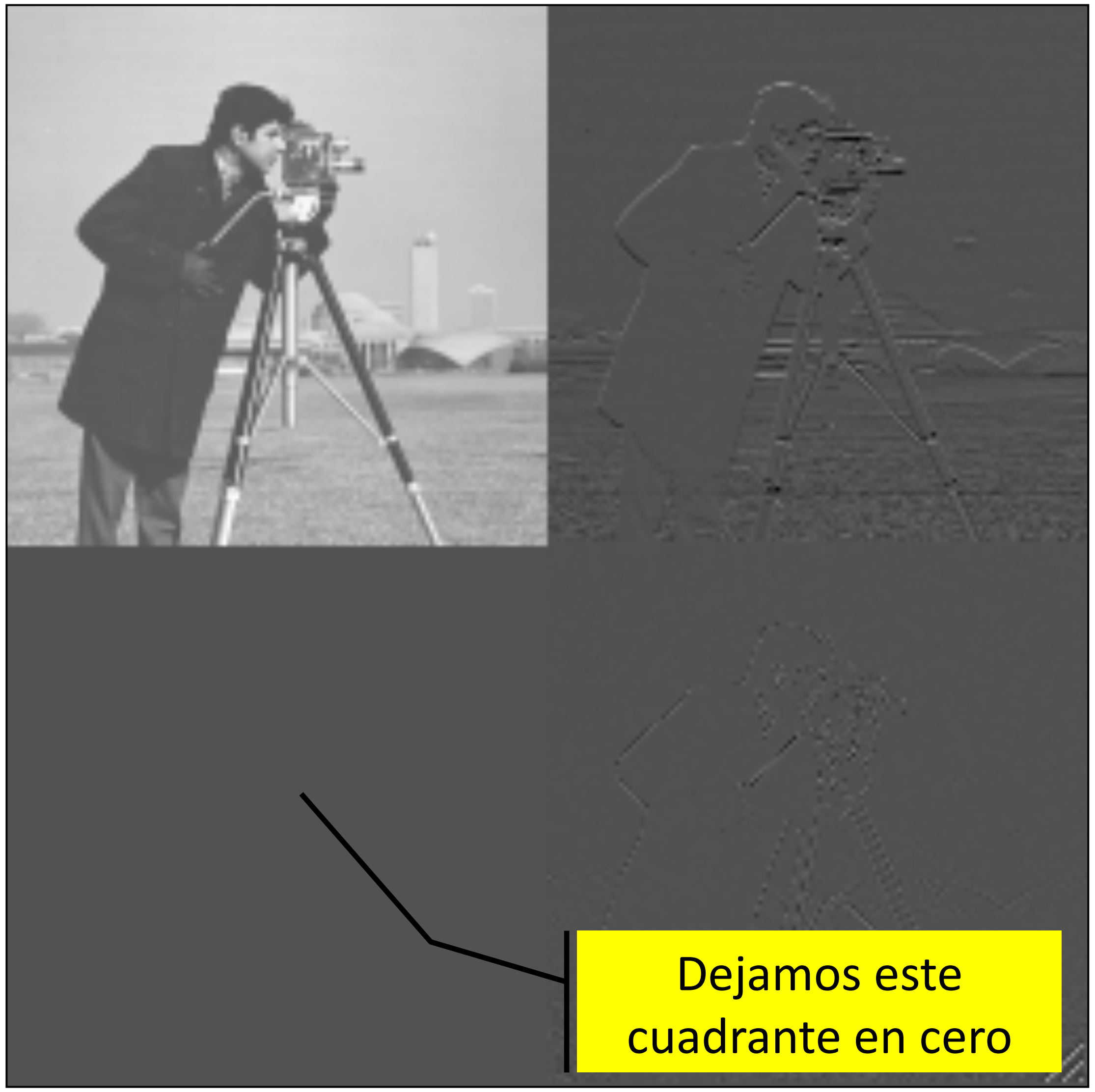


$$G = H^{\mathrm{T}} \cdot V \cdot H$$

- Transformada de Haar
  - Al modificar ciertas componentes podemos reducir la información en la imagen.

Imagen reordenada con transformada Haar

Imagen reordenada modificada



- Transformada de Haar
  - Del ejemplo anterior veamos el resultado de aplicar el proceso.

Imagen original

Imagen comprimida
(cuadrante 2 en cero)

Transformada de Haar

- Del ejemplo anterior veamos el resultado de aplicar el proceso.

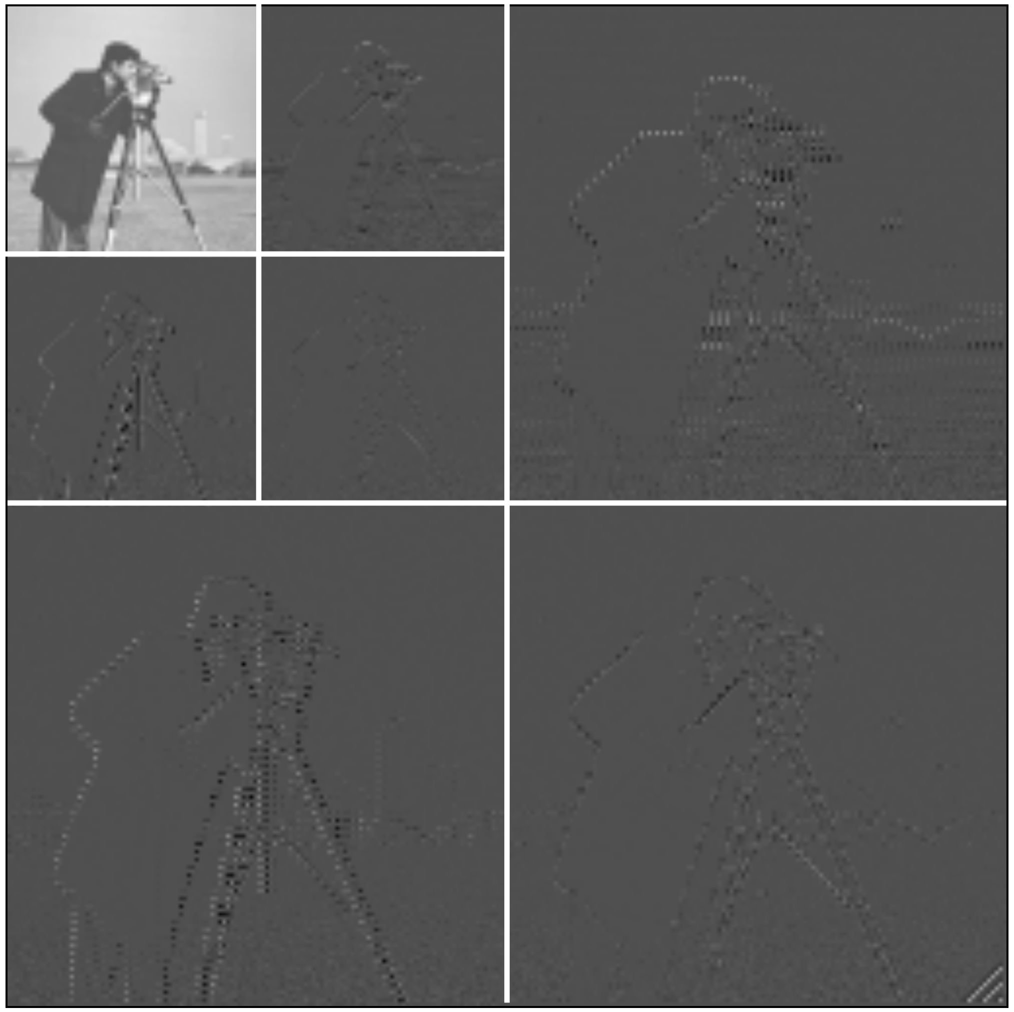
Imagen original

Imagen comprimida
(cuadrante 1 y 2 en cero)

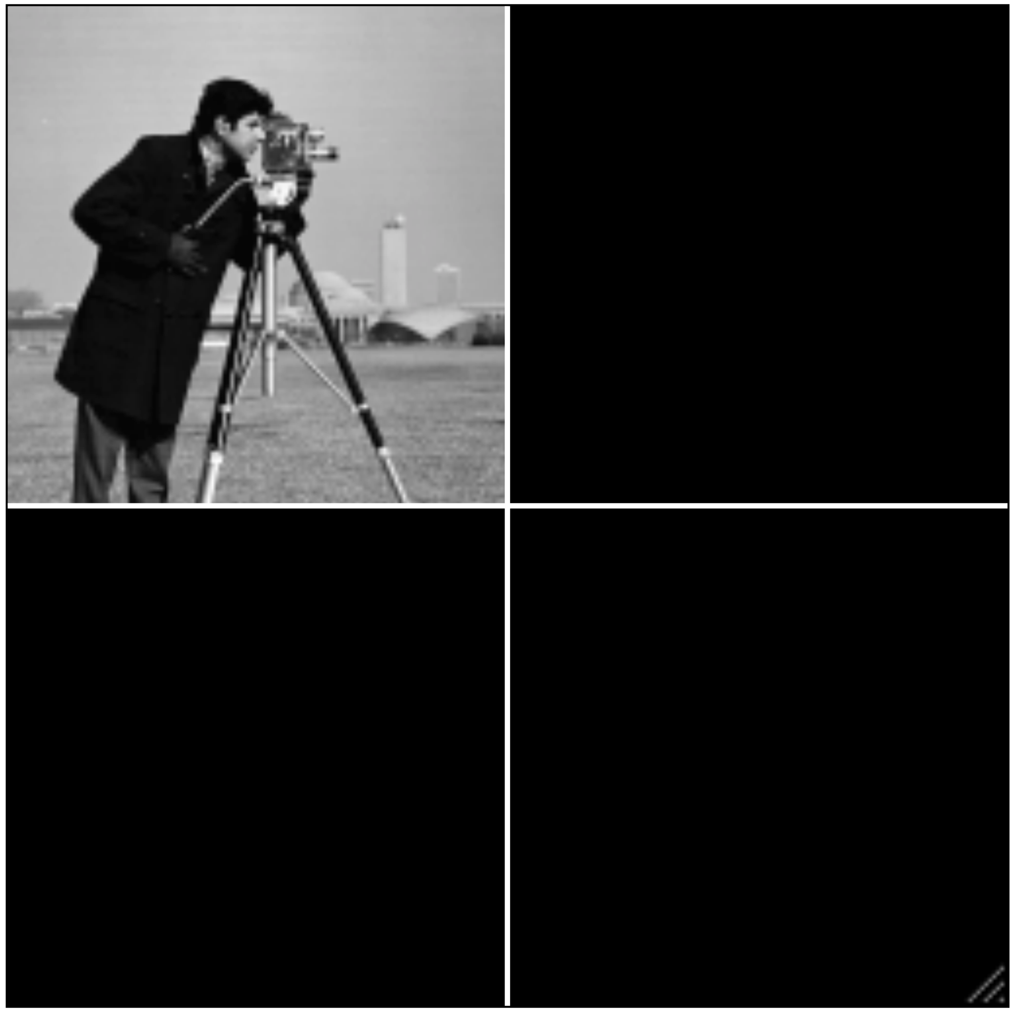
- Transformada de Haar
  - Podemos codificar nuevamente el cuadrante con más energía. SI, efectivamente esto se llama iterar nuevamente. (Para los computines es recursión).

Primera iteración

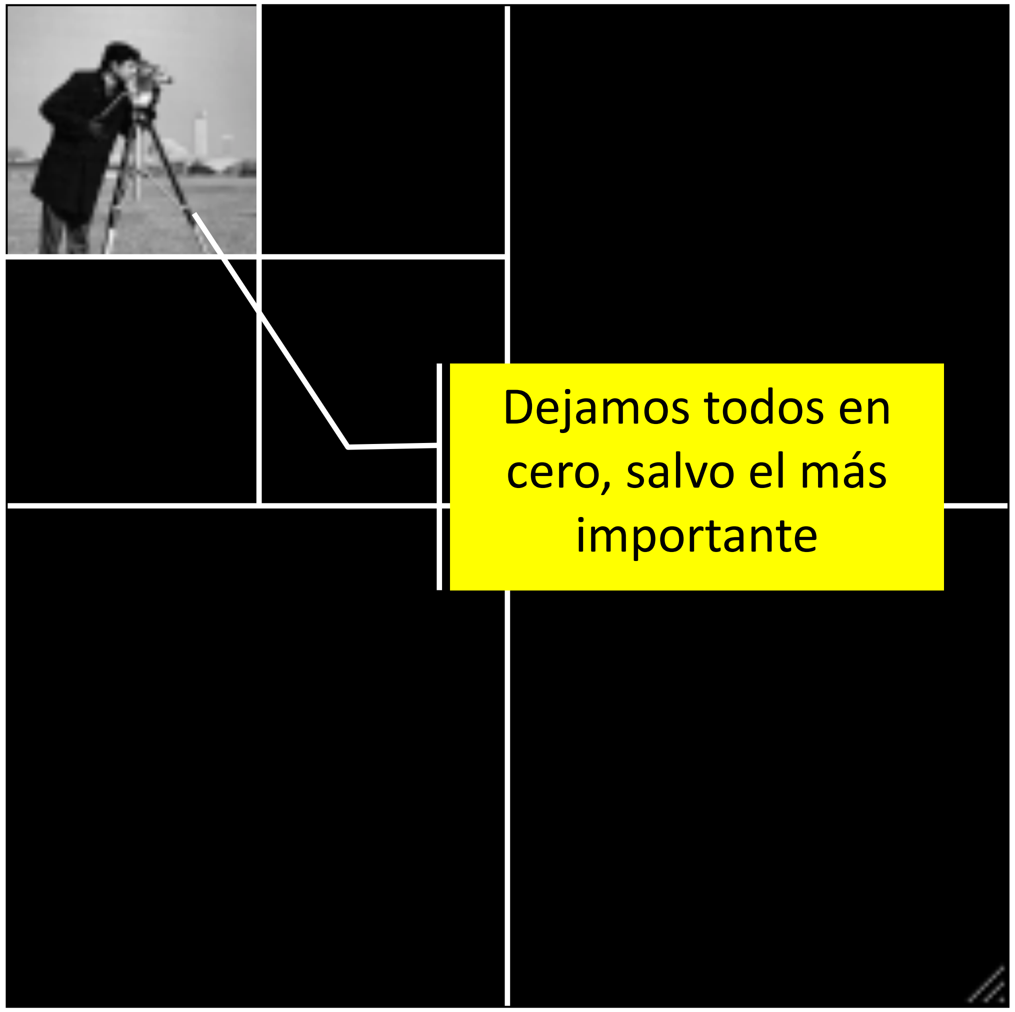
Segunda iteración



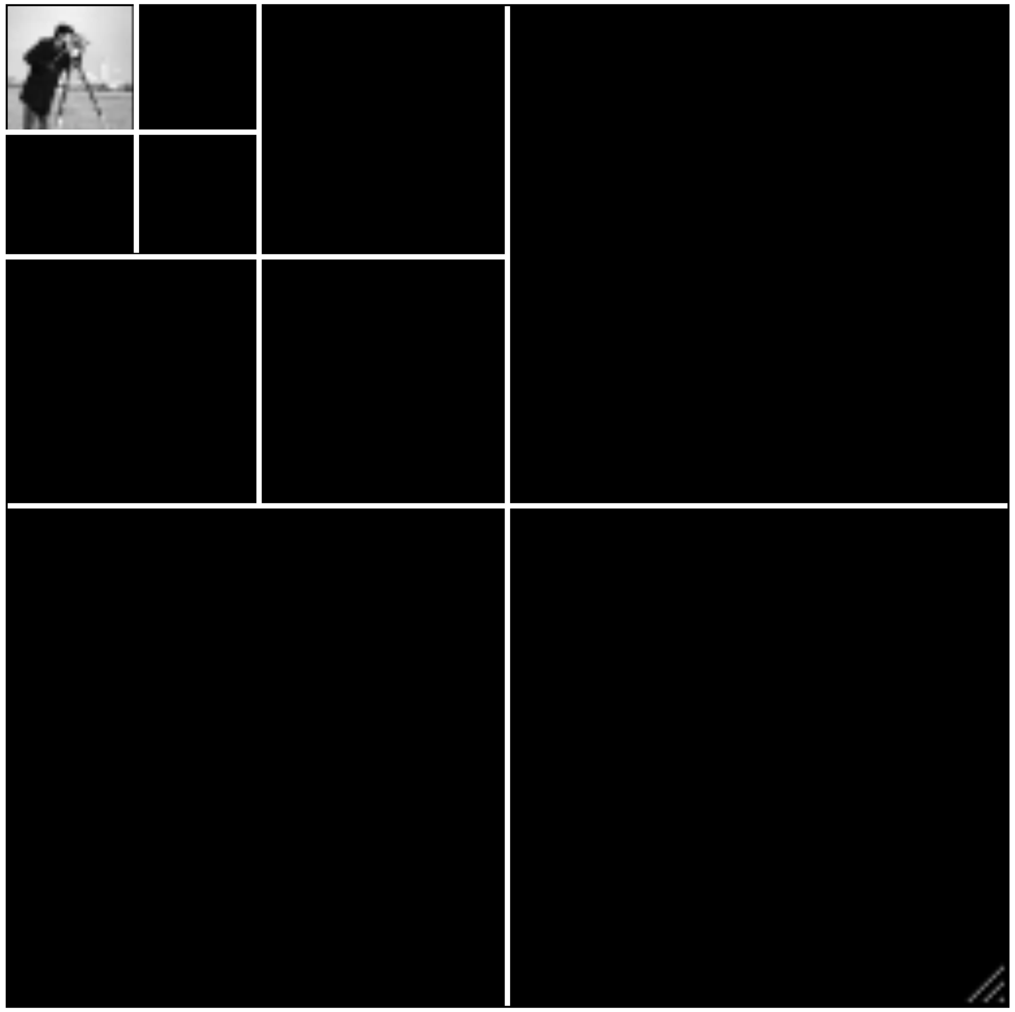
- Transformada de Haar
  - Podemos codificar nuevamente el cuadrante con más energía. SI, efectivamente esto se llama iterar nuevamente. (Para los computines es recursión).



- Transformada de Haar
  - Podemos codificar nuevamente el cuadrante con más energía. SI, efectivamente esto se llama iterar nuevamente. (Para los computines es recursión).



- Transformada de Haar
  - Podemos codificar nuevamente el cuadrante con más energía. SI, efectivamente esto se llama iterar nuevamente. (Para los computines es recursión).



Temario

- Multiresolución y Compresión
  - Filtro de Haar (Matricial)
  - Análisis vectorial
  - Ejemplo en imagenes

Filtro Haar (vectorial)

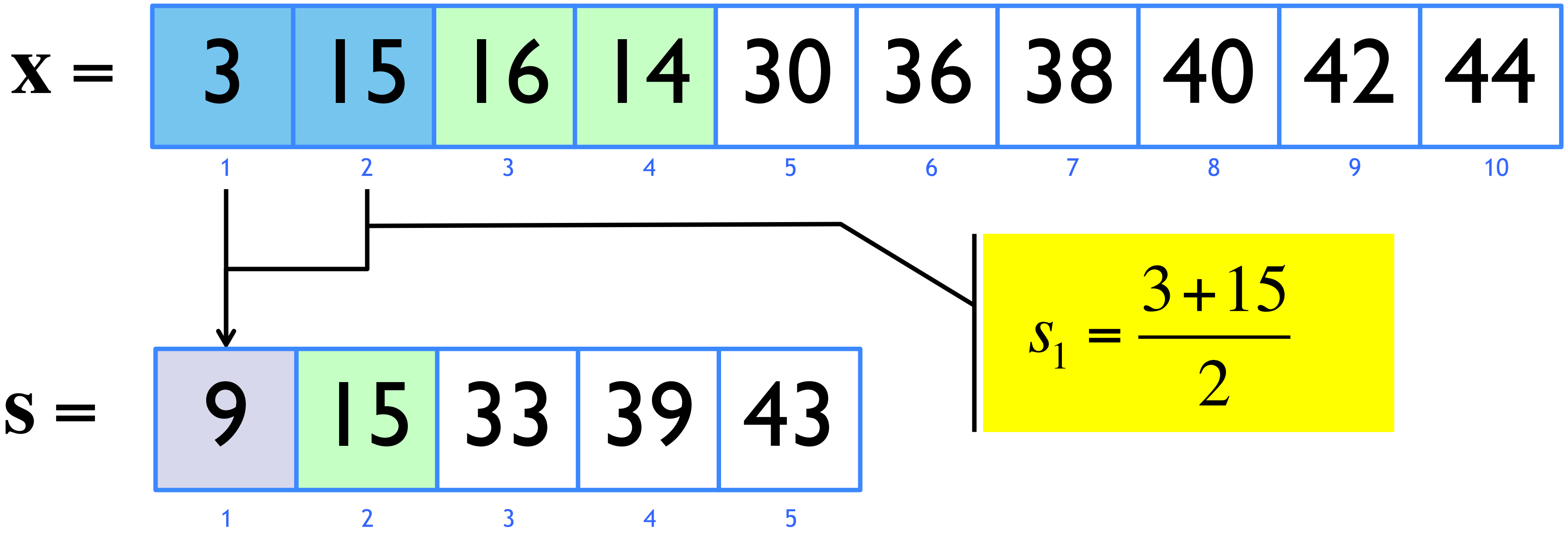
- Filtro de Haar
  - Normalmente el filtro de Haar es analizado desde su forma vectorial, ya que es más intuitivo y más eficiente su cálculo.
  - Suponga que usted tiene un vector $\mathbf{X}$ con N valores donde N es par.

$$\mathbf{x} = \begin{array}{cccccccccc} 3 & 15 & 16 & 14 & 30 & 36 & 38 & 40 & 42 & 44 \\ \scriptstyle 1 & \scriptstyle 2 & \scriptstyle 3 & \scriptstyle 4 & \scriptstyle 5 & \scriptstyle 6 & \scriptstyle 7 & \scriptstyle 8 & \scriptstyle 9 & \scriptstyle 10 \end{array}$$

Objetivo:
Enviar este vector en forma reducida (comprimida)

- Filtro de Haar (vectorial)
  - Una solución consiste en enviar dos secciones del vector. **Primero**: enviamos el promedio entre pares de valores.

$$s_k = \frac{x_{2k-1} + x_{2k}}{2}, \qquad k = 1, \ldots, \tfrac{N}{2}$$



Filtro Haar (vectorial)

- Filtro de Haar (vectorial)
  - Una solución consiste en enviar dos secciones del vector. **Segundo**: junto a la lista anterior enviamos la diferencia entre pares de valores.

$$d_k = \frac{x_{2k} - x_{2k-1}}{2}, \qquad k = 1, \ldots, \tfrac{N}{2}$$

x =

3	15	16	14	30	36	38	40	42	44
1	2	3	4	5	6	7	8	9	10

$$s_1 = \frac{15 - 3}{2}$$

d =

6	-1	3	1	1
1	2	3	4	5

Filtro Haar (vectorial)

- Filtro de Haar (vectorial)
  - Como vemos el proceso es completamente invertible al **restar** y sumar ambas listas:

$$\boxed{s_k - d_k = \frac{x_{2k-1} + x_{2k}}{2} - \frac{x_{2k} - x_{2k-1}}{2}} \longrightarrow \boxed{x_{2k-1} = s_k - d_k}$$

s =

9	15	33	39	43
1	2	3	4	5

d =

6	-1	3	1	1
1	2	3	4	5

$$s_1 - d_1 = 9 - 6$$

x =

3		16		30		38		42	
1	2	3	4	5	6	7	8	9	10

- Filtro de Haar (vectorial)
  - Como vemos el proceso es completamente invertible al restar y **sumar** ambas listas:

$$s_k + d_k = \frac{x_{2k-1} + x_{2k}}{2} + \frac{x_{2k} - x_{2k-1}}{2} \longrightarrow x_{2k} = s_k + d_k$$

s =

9	15	33	39	43
1	2	3	4	5

d =

6	-1	3	1	1
1	2	3	4	5

$$s_1 + d_1 = 9 + 6$$

x =

	15		14		36		40		44
1	2	3	4	5	6	7	8	9	10

Filtro Haar (vectorial)

- Filtro de Haar
  - En definitiva, a partir de un vector **X** podemos mapearlo en dos vectores.

$$\boxed{\mathbf{X} = \left(x_1, \ldots, x_N\right) \rightarrow \left(\mathbf{s} \mid \mathbf{d}\right) = \left(s_1, \ldots, s_{\frac{N}{2}} \mid d_1, \ldots, d_{\frac{N}{2}}\right)}$$

$\mathbf{x} =$

3	15	16	14	30	36	38	40	42	44
1	2	3	4	5	6	7	8	9	10

$\left(\mathbf{s} \mid \mathbf{d}\right) =$

9	15	33	39	43	6	-1	3	1	1
1	2	3	4	5	1	2	3	4	5

Filtro Haar (vectorial)

- Filtro de Haar
  - En definitiva, a partir de un vector $\mathbf{X}$ podemos mapearlo en dos vectores.
  - Ventajas:
    1. Identificar largas diferencias en el vector de diferencia $(\mathbf{d})$
    2. La transformación concentra la señal de energía en menos valores
    3. Y además tenemos menos dígitos que enviar (notar los números pequeños en la diferencia)

$\mathbf{x} =$

3	15	16	14	30	36	38	40	42	44
1	2	3	4	5	6	7	8	9	10

$(\mathbf{s} \mid \mathbf{d}) =$

9	15	33	39	43	6	-1	3	1	1
1	2	3	4	5	1	2	3	4	5

- Filtro de Haar
  - En forma matricial, cómo podemos obtener el mismo resultado.

$$\left[\begin{array}{c} \\ \\ \\ \hline \\ \\ \\ \end{array}\right] \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \frac{1}{2} \cdot \left[\begin{array}{c} x_1 + x_2 \\ x_3 + x_4 \\ x_5 + x_6 \\ x_7 + x_8 \\ \hline x_2 - x_1 \\ x_4 - x_3 \\ x_6 - x_5 \\ x_8 - x_7 \end{array}\right]$$

Qué coeficientes deben ir en la matriz

- Filtro de Haar
  - Facilitemos el trabajo, veamos el primer valor de la matriz.

$$\begin{bmatrix} \square & \square & \square & \square & \square & \square & \square & \square \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \frac{1}{2} \cdot \left(x_1 + x_2 \right)$$

Cuáles son los valores que deben ir

Filtro Haar (vectorial)

- Filtro de Haar
  - Super simple, en la primera fila tenemos los valores que multiplican los primeros dos términos del vector

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \frac{1}{2} \cdot (x_1 + x_2)$$

Ahora queda claro porque son estos los valores?

- Filtro de Haar
  - En forma matricial el resultado aplicado a los otros factores es el siguiente.

$$
\left[\begin{array}{cccccccc}
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\
\hdashline
-\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & -\frac{1}{2}
\end{array}\right]
\cdot
\left[\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{array}\right]
= \frac{1}{2} \cdot
\left[\begin{array}{c} x_1 + x_2 \\ x_3 + x_4 \\ x_5 + x_6 \\ x_7 + x_8 \\ \hdashline x_2 - x_1 \\ x_4 - x_3 \\ x_6 - x_5 \\ x_8 - x_7 \end{array}\right]
$$

Esta es el vector de entrada

- Filtro de Haar

INVERSA

  - En relación a la transformación inversa (obtener la señal original) ocupamos otra matriz **(simplemente transponer la matriz anterior**)

$$\frac{1}{2}\cdot\left[\begin{array}{cccc:cccc} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right]\cdot\begin{bmatrix} x_1+x_2 \\ x_3+x_4 \\ x_5+x_6 \\ x_7+x_8 \\ x_2-x_1 \\ x_4-x_3 \\ x_6-x_5 \\ x_8-x_7 \end{bmatrix}=\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix}$$

Esta es el vector de entrada

Filtro Haar (vectorial)

- Filtro de Haar
  - En forma matricial, como podemos obtener el mismo resultado.
  - Sea $\mathbf{W}_8$ la matriz discreta de Haar.

$$
\mathbf{W}_8 = \left[\begin{array}{cccccccc}
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\
\hline
-\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & -\frac{1}{2}
\end{array}\right] = \left[\begin{array}{c} \mathbf{H} \\ \hline \mathbf{G} \end{array}\right]
$$

iUAI
UNIVERSIDAD ADOLFO IBÁÑEZ
FACULTAD DE INGENIERÍA Y CIENCIAS

- Filtro de Haar
  - Del resultado anterior, podemos inferir que

$$\mathbf{W}_8 \cdot \mathbf{X} = \begin{bmatrix} \mathbf{H} \\ \mathbf{G} \end{bmatrix} \cdot \mathbf{x} = \begin{bmatrix} \mathbf{Hx} \\ \mathbf{Gx} \end{bmatrix}$$

$$\begin{array}{c} \mathbf{H} \\ \\ \\ \\ \\ \\ \\ \mathbf{G} \end{array} \underbrace{\left[\begin{array}{cccccccc} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \hdashline -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & -\frac{1}{2} \end{array}\right]}_{\mathbf{W}_8} \cdot \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix}}_{\mathbf{x}} = \frac{1}{2} \cdot \left[\begin{array}{c} x_1 + x_2 \\ x_3 + x_4 \\ x_5 + x_6 \\ x_7 + x_8 \\ \hdashline x_2 - x_1 \\ x_4 - x_3 \\ x_6 - x_5 \\ x_8 - x_7 \end{array}\right]$$

Filtro Haar (vectorial)

- Filtro de Haar
  - Otra forma de decir lo mismo es:

$$\mathbf{H}\cdot\mathbf{x}=\begin{bmatrix}\frac{1}{2}&\frac{1}{2}&0&0&0&0&0&0\\0&0&\frac{1}{2}&\frac{1}{2}&0&0&0&0\\0&0&0&0&\frac{1}{2}&\frac{1}{2}&0&0\\0&0&0&0&0&0&\frac{1}{2}&\frac{1}{2}\end{bmatrix}\cdot\begin{bmatrix}x_1\\x_2\\x_3\\x_4\\x_5\\x_6\\x_7\\x_8\end{bmatrix}$$

$$=\frac{1}{2}\cdot\begin{bmatrix}x_1+x_2\\x_3+x_4\\x_5+x_6\\x_7+x_8\end{bmatrix}=\boxed{\begin{bmatrix}x_1&x_2\\x_3&x_4\\x_5&x_6\\x_7&x_8\end{bmatrix}\cdot\begin{bmatrix}\frac{1}{2}\\\frac{1}{2}\end{bmatrix}}$$

Otra forma de expresar el mismo resultado

- Filtro de Haar
  - Otra forma de decir lo mismo es:

$$\mathbf{G}\cdot\mathbf{x}=\begin{bmatrix} -\tfrac{1}{2} & \tfrac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\tfrac{1}{2} & \tfrac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\tfrac{1}{2} & \tfrac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\tfrac{1}{2} & \tfrac{1}{2} \end{bmatrix}\cdot\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix}$$

$$=\frac{1}{2}\cdot\begin{bmatrix} x_2-x_1 \\ x_4-x_3 \\ x_6-x_5 \\ x_8-x_7 \end{bmatrix}=\boxed{\begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \\ x_5 & x_6 \\ x_7 & x_8 \end{bmatrix}\cdot\begin{bmatrix} -\tfrac{1}{2} \\ \tfrac{1}{2} \end{bmatrix}}$$

Otra forma de expresar el mismo resultado

- Filtro de Haar
  - Si ordenamos el vector de entrada de otra forma, podemos codificar **W** en forma vectorial.

$$\begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \\ x_5 & x_6 \\ x_7 & x_8 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} = \frac{1}{2} \cdot \begin{bmatrix} x_1 + x_2 \\ x_3 + x_4 \\ x_5 + x_6 \\ x_7 + x_8 \end{bmatrix} \longrightarrow s$$

$$\begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \\ x_5 & x_6 \\ x_7 & x_8 \end{bmatrix} \cdot \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} = \frac{1}{2} \cdot \begin{bmatrix} x_2 - x_1 \\ x_4 - x_3 \\ x_6 - x_5 \\ x_8 - x_7 \end{bmatrix} \longrightarrow d$$

Importante: Reodenar el vector x

BASES DE HAAR. Ahora son vectores

- Filtro de Haar
  - En forma genérica. Si tenemos una columna de tamaño N debemos reordenar los vectores de entrada de la siguiente forma

$$\underbrace{\begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \\ \vdots & \vdots \\ x_{N-1} & x_N \end{bmatrix}}_{\mathbf{W}_N} \cdot \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} = \frac{1}{2} \cdot \begin{bmatrix} x_1 + x_2 \\ x_3 + x_4 \\ \vdots \\ x_{N-1} + x_N \end{bmatrix} \longrightarrow \boxed{s}$$

$$\begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \\ \vdots & \vdots \\ x_{N-1} & x_N \end{bmatrix} \cdot \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} = \frac{1}{2} \cdot \begin{bmatrix} x_2 - x_1 \\ x_4 - x_3 \\ \vdots \\ x_N - x_{N-1} \end{bmatrix} \longrightarrow \boxed{d}$$

Importante: Reodenar el vector x

BASES DE HAAR. Ahora son vectores

- Filtro de Haar

INVERSA

  - Ahora suponga que ya calculamos el vector **s** y **d**; y deseamos calcular la inversa. Supuesto, vamos a pensar que los valores están en un vector **y**.

$$\left[\begin{array}{cccc:cccc} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right] \cdot \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \end{bmatrix} = \begin{bmatrix} y_1 - y_5 \\ y_1 + y_5 \\ y_2 - y_6 \\ y_2 + y_6 \\ y_3 - y_7 \\ y_3 + y_7 \\ y_4 - y_8 \\ y_4 + y_8 \end{bmatrix}$$

Esta es el vector de entrada

iUAI
UNIVERSIDAD ADOLFO IBÁÑEZ
FACULTAD DE INGENIERÍA Y CIENCIAS

INVERSA

- Filtro de Haar
  - Ahora suponga que ya calculamos el vector **s** y **d**; y deseamos calcular la inversa. Supuesto, vamos a pensar que los valores están en un vector **y**.

$$\begin{bmatrix} y_1 & y_5 \\ y_2 & y_6 \\ y_3 & y_7 \\ y_4 & y_8 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} y_1 - y_5 \\ y_2 - y_6 \\ y_3 - y_7 \\ y_4 - y_8 \end{bmatrix} \longrightarrow \boxed{a}$$

$$\begin{bmatrix} y_1 & y_5 \\ y_2 & y_6 \\ y_3 & y_7 \\ y_4 & y_8 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} y_1 + y_5 \\ y_2 + y_6 \\ y_3 + y_7 \\ y_4 + y_8 \end{bmatrix} \longrightarrow \boxed{b}$$

BASES DE HAAR DE INVERSA.

- Filtro de Haar

INVERSA

  - Ahora suponga que ya calculamos el vector **s** y **d**; y deseamos calcular la inversa. Supuesto, vamos a pensar que los valores están en un vector **y**.

$$\begin{bmatrix} y_1 & y_5 \\ y_2 & y_6 \\ y_3 & y_7 \\ y_4 & y_8 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} y_1 - y_5 \\ y_2 - y_6 \\ y_3 - y_7 \\ y_4 - y_8 \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}$$

$$\begin{bmatrix} y_1 & y_5 \\ y_2 & y_6 \\ y_3 & y_7 \\ y_4 & y_8 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} y_1 + y_5 \\ y_2 + y_6 \\ y_3 + y_7 \\ y_4 + y_8 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

BASES DE HAAR DE INVERSA.

iUAI
UNIVERSIDAD ADOLFO IBÁÑEZ
FACULTAD DE INGENIERÍA Y CIENCIAS

- Filtro de Haar

INVERSA

  - Ahora suponga que ya calculamos el vector **s** y **d**; y deseamos calcular la inversa. Supuesto, vamos a pensar que los valores están en un vector **y**.

$$\begin{bmatrix} y_1 - y_5 \\ y_2 - y_6 \\ y_3 - y_7 \\ y_4 - y_8 \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}$$

$$\begin{bmatrix} y_1 + y_5 \\ y_2 + y_6 \\ y_3 + y_7 \\ y_4 + y_8 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

Valores obtenidos

$$\mathbf{x} = \begin{bmatrix} y_1 - y_5 \\ y_1 + y_5 \\ y_2 - y_6 \\ y_2 + y_6 \\ y_3 - y_7 \\ y_3 + y_7 \\ y_4 - y_8 \\ y_4 + y_8 \end{bmatrix} \Rightarrow \mathbf{x} = \begin{bmatrix} a_1 \\ b_1 \\ a_2 \\ b_2 \\ a_3 \\ b_3 \\ a_4 \\ b_4 \end{bmatrix}$$

Objetivo

- Filtro de Haar

INVERSA

  - Ahora suponga que tenemos un vector de largo N

$$\begin{bmatrix} y_1 & y_{\frac{N}{2}+1} \\ \vdots & \vdots \\ y_{\frac{N}{2}} & y_N \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} y_1 - y_5 \\ y_2 - y_6 \\ \vdots \\ y_{\frac{N}{2}} - y_N \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{\frac{N}{2}} \end{bmatrix}$$

$$\begin{bmatrix} y_1 & y_{\frac{N}{2}+1} \\ \vdots & \vdots \\ y_{\frac{N}{2}} & y_N \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} y_1 + y_5 \\ y_2 + y_6 \\ \vdots \\ y_{\frac{N}{2}} + y_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{\frac{N}{2}} \end{bmatrix}$$

Importante: Reodenar el vector x

iUAI
UNIVERSIDAD ADOLFO IBÁÑEZ
FACULTAD DE INGENIERÍA Y CIENCIAS

INVERSA

- Filtro de Haar
  - Ahora suponga que tenemos un vector de largo N

$$\overbrace{\begin{bmatrix} y_1 - y_{\frac{N}{2}+1} \\ y_2 - y_{\frac{N}{2}+2} \\ \vdots \\ y_{\frac{N}{2}} - y_N \end{bmatrix}}^{\mathbf{W}_N} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{\frac{N}{2}} \end{bmatrix}$$

$$\begin{bmatrix} y_1 + y_{\frac{N}{2}+1} \\ y_2 + y_{\frac{N}{2}+2} \\ \vdots \\ y_{\frac{N}{2}} + y_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{\frac{N}{2}} \end{bmatrix}$$

Valores obtenidos

$$\mathbf{x} = \begin{bmatrix} y_1 - y_{\frac{N}{2}+1} \\ y_1 + y_{\frac{N}{2}+1} \\ y_2 - y_{\frac{N}{2}+2} \\ y_2 + y_{\frac{N}{2}+2} \\ \vdots \\ y_{\frac{N}{2}} - y_N \\ y_{\frac{N}{2}} + y_N \end{bmatrix} \Rightarrow \mathbf{x} = \begin{bmatrix} a_1 \\ b_1 \\ a_2 \\ b_2 \\ \vdots \\ a_{\frac{N}{2}} \\ b_{\frac{N}{2}} \end{bmatrix}$$

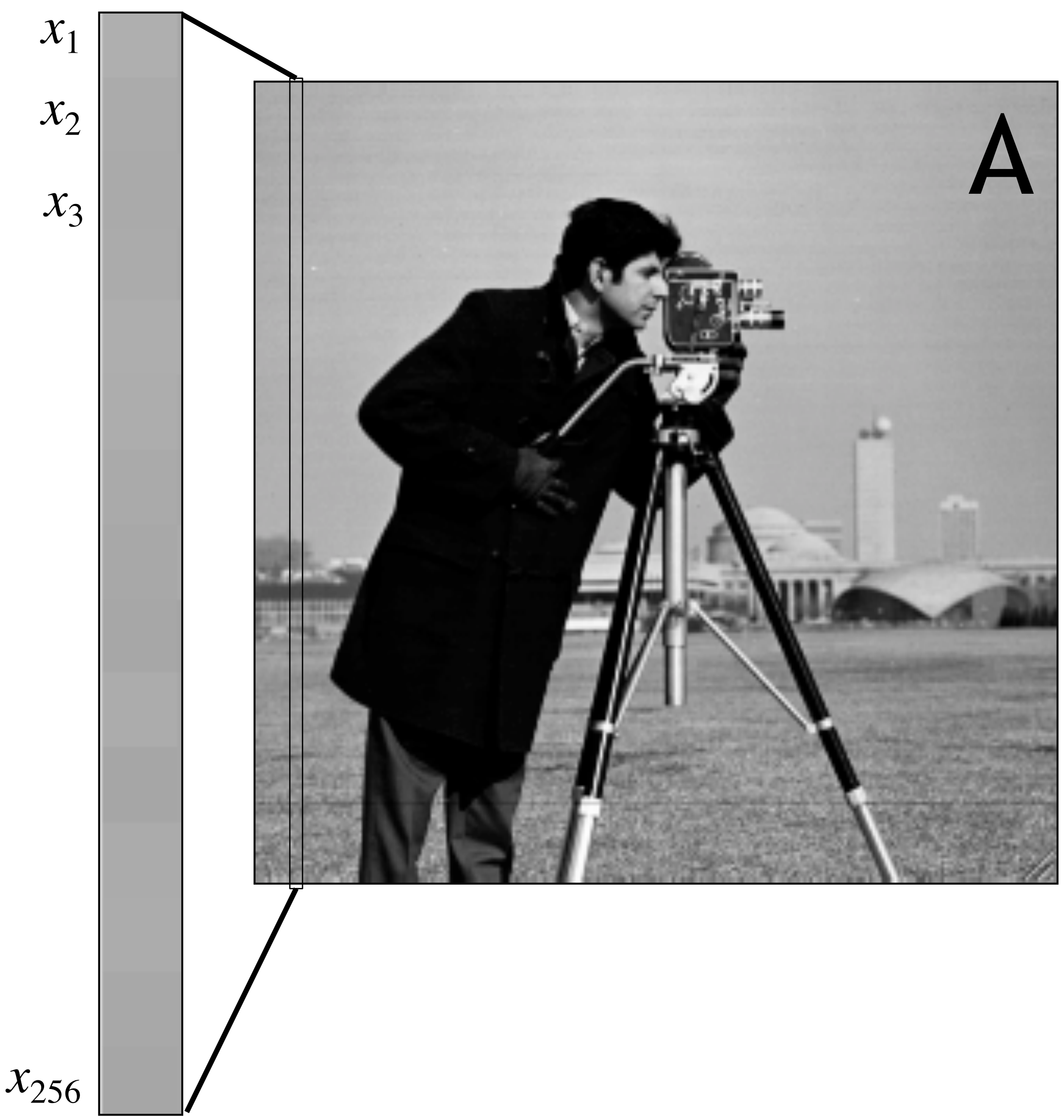
Objetivo

Temario

- Multiresolución y Compresión
  - Ejemplo en imágenes
  - Multidescomposición
  - Reducción del ruido
  - Resultado

Codificación de Haar

- Aplicación en imágenes
  - Si x_1, x_2, x_{256} son las columnas de la imagen A, debemos transformar cada columna en una matriz w_i de la siguiente forma:

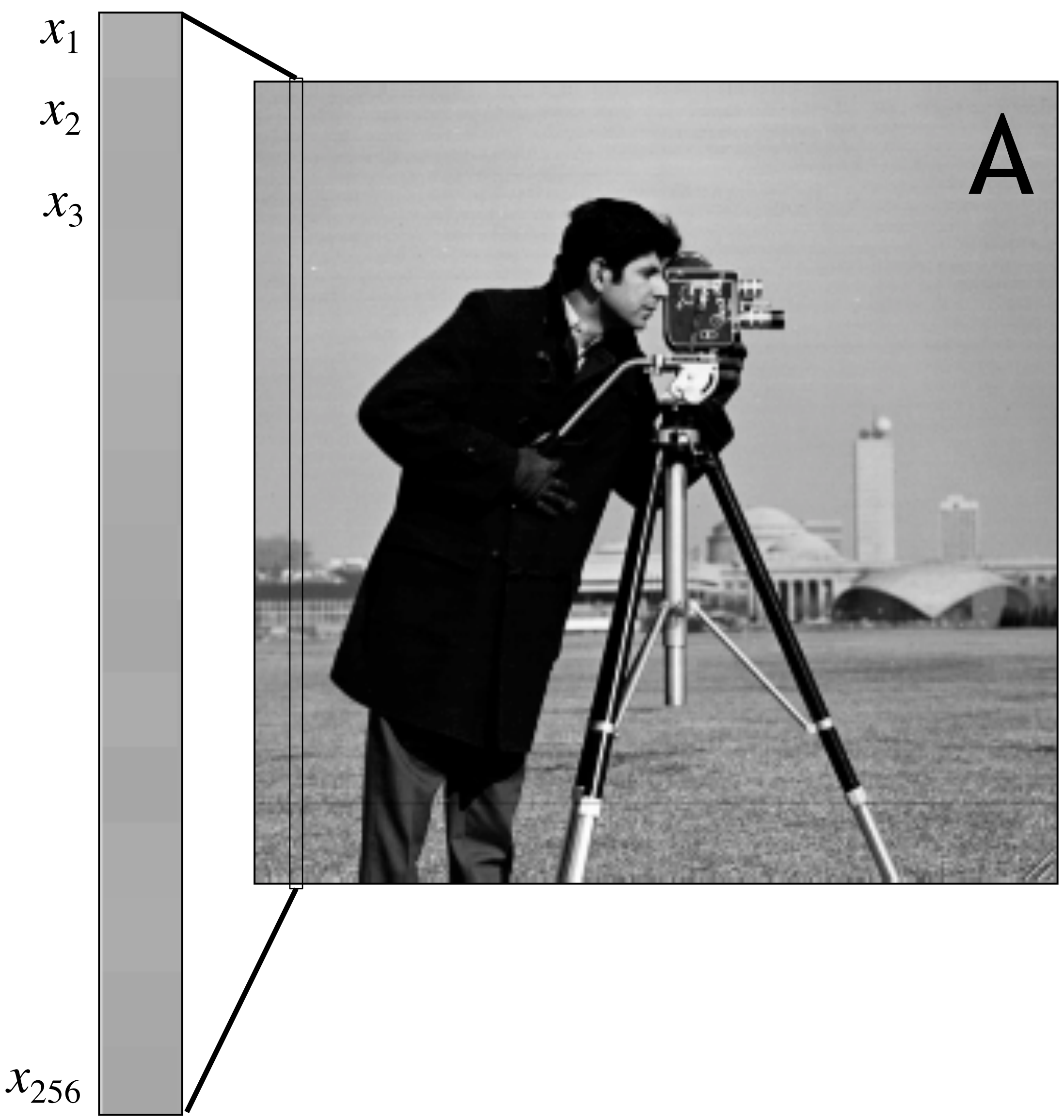


$$\mathbf{w}_i = \begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \\ \vdots & \vdots \\ x_{N-1} & x_N \end{bmatrix}$$

Cada columna se reordena de la siguiente forma

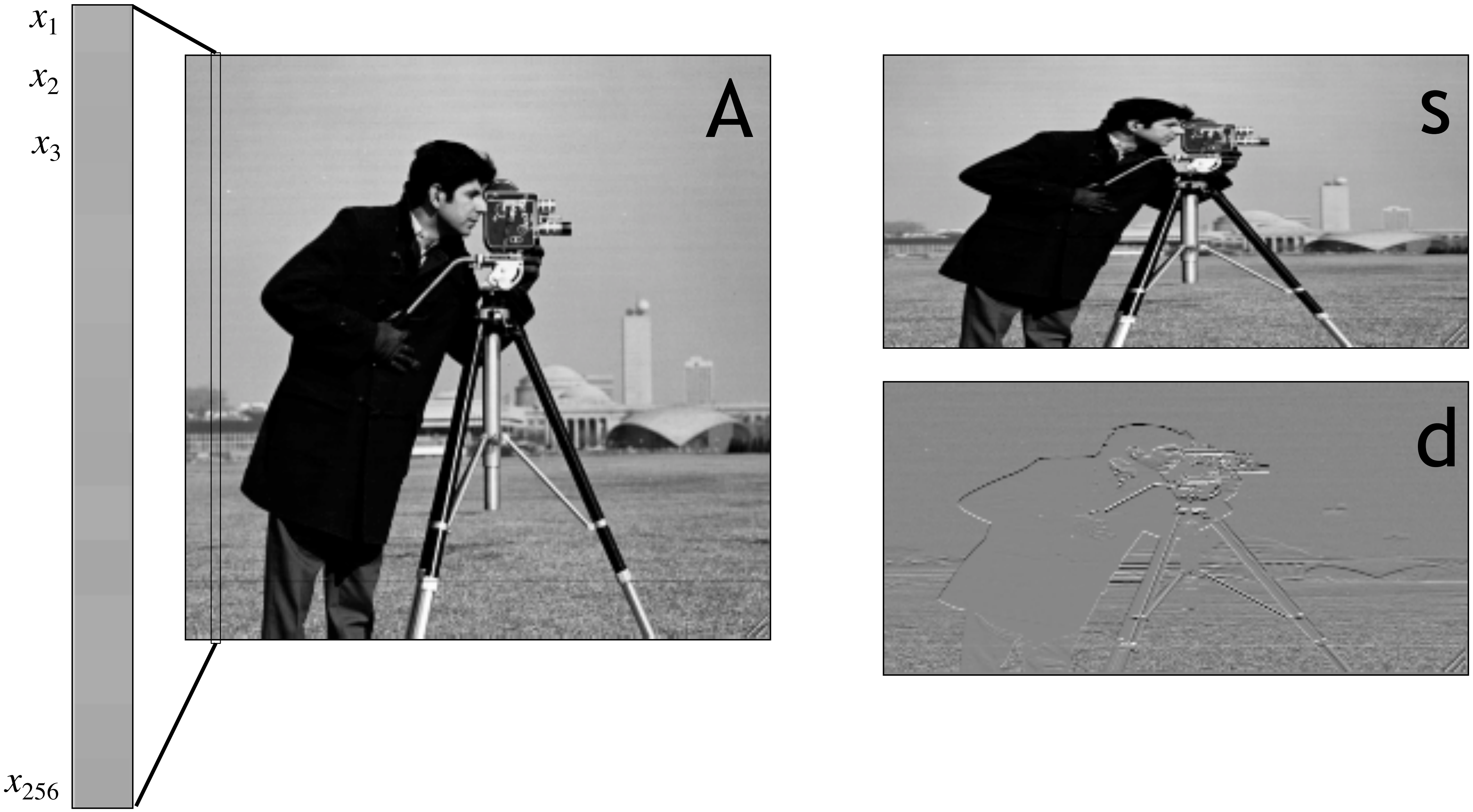
Codificación de Haar

- Aplicación en imágenes
  - Por cada columna determinamos los vectores **s** y **d** simplemente multiplicándolas por las bases



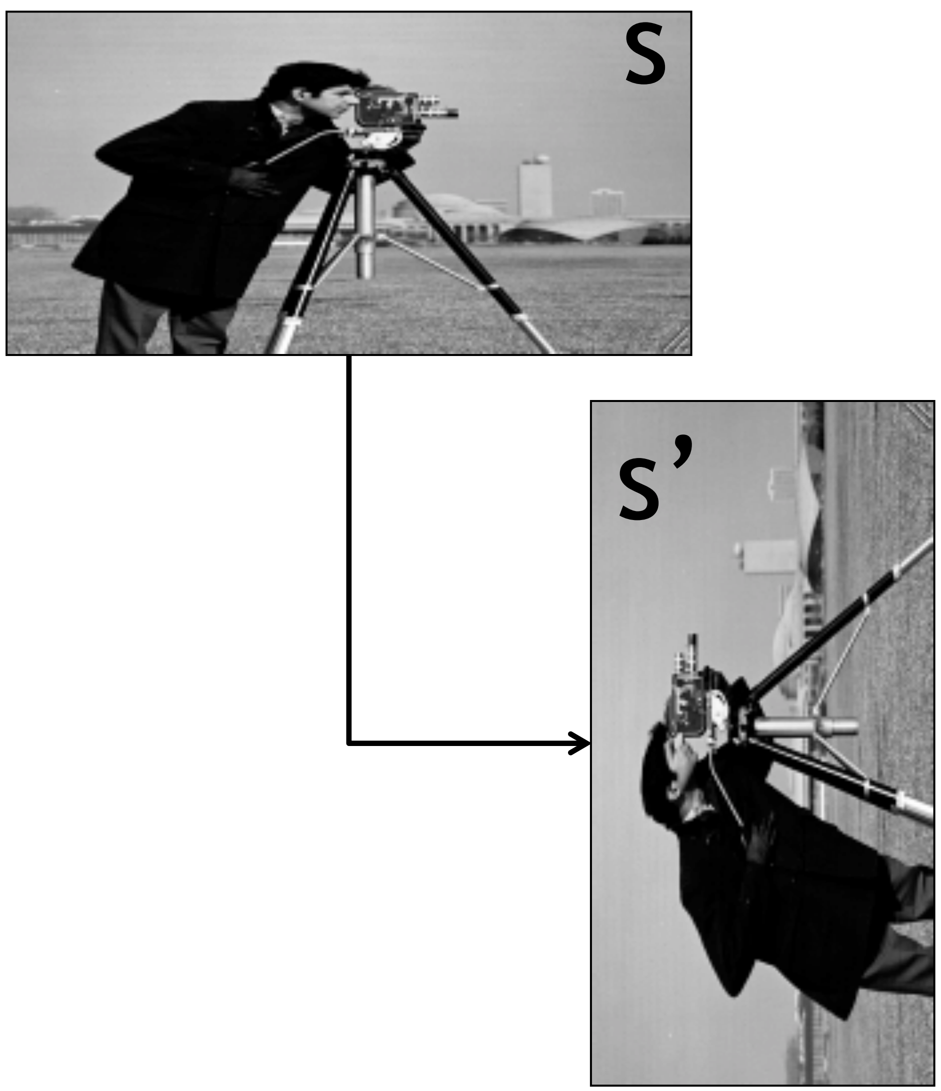
$$\mathbf{s}_i = \mathbf{w}_i \cdot \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} \qquad \mathbf{d}_i = \mathbf{w}_i \cdot \begin{bmatrix} \frac{1}{2} \\ -\frac{1}{2} \end{bmatrix}$$

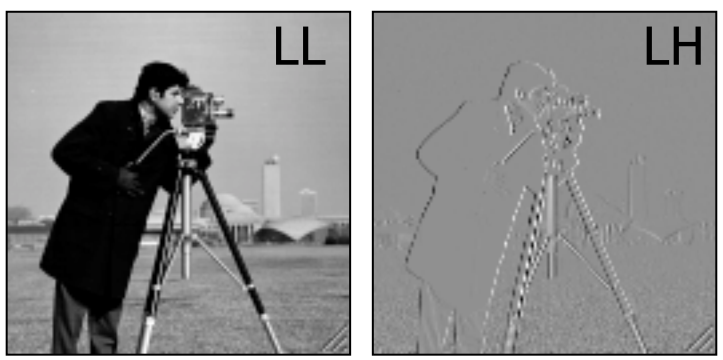
- Aplicación en imágenes
  - ¿Qué resultado genera las matrices **s** y **d** ?



Codificación de Haar

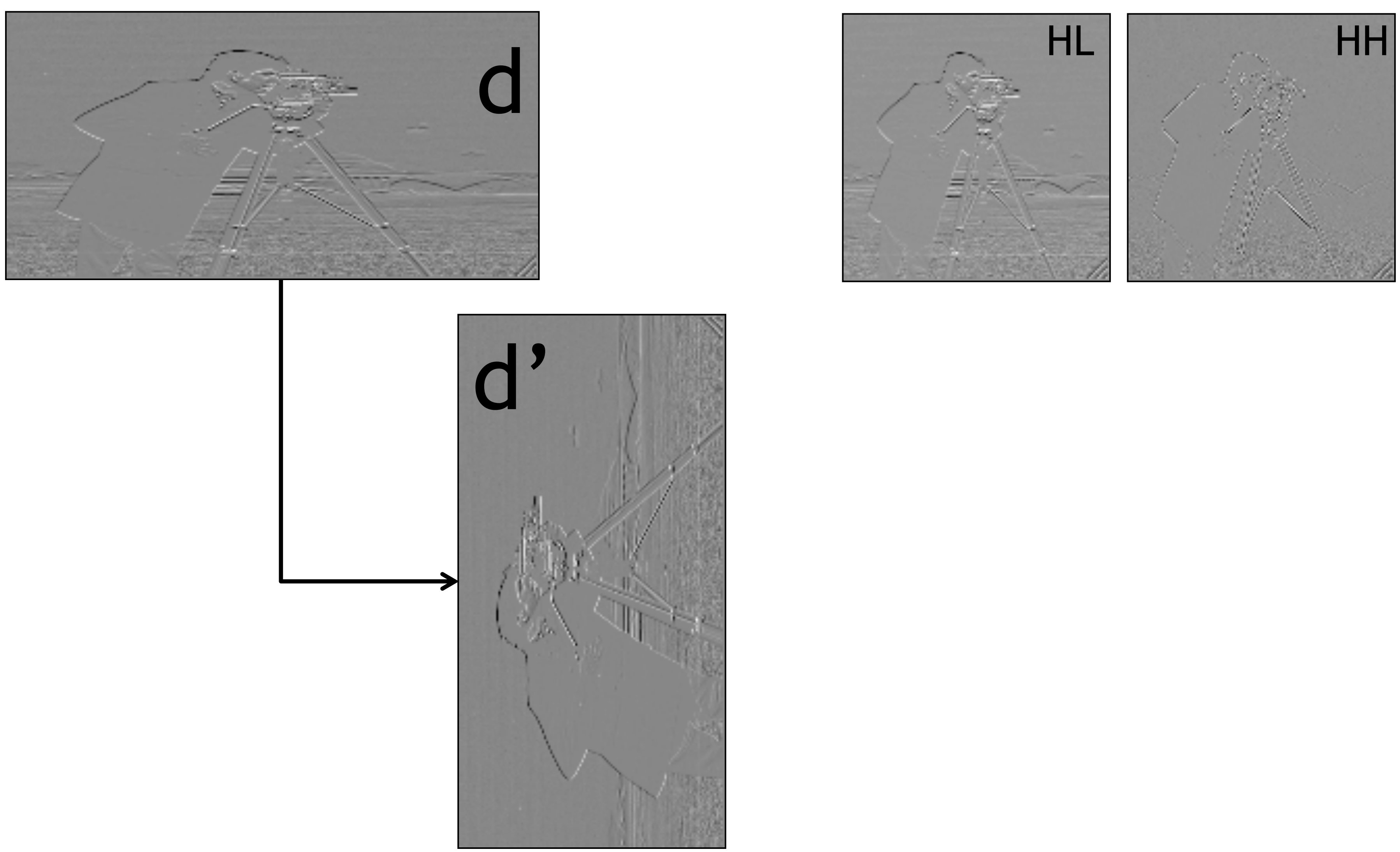
- Aplicación en imágenes
  - Posteriormente aplicamos el mismo proceso en la matriz **s** y la matriz **d** , lo cual genera la imagen de Haar. Para ello debemos rotar la imagen **s** y aplicar nuevamente el mismo algoritmo





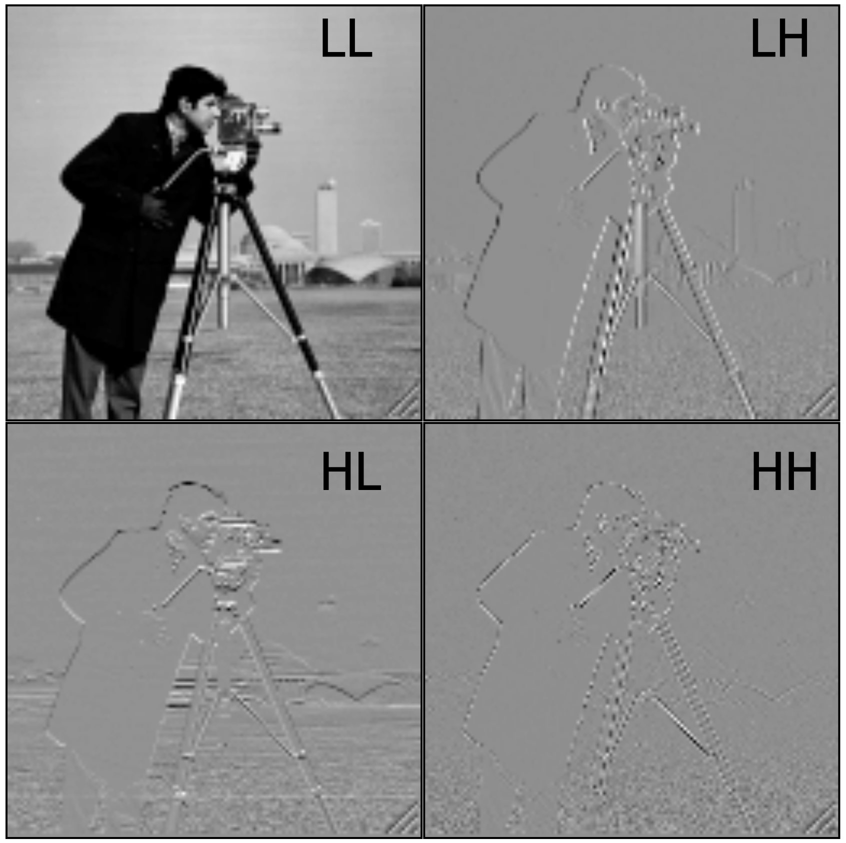
Codificación de Haar

- Aplicación en imágenes
  - Posteriormente aplicamos el mismo proceso en la matriz **s** y la matriz **d** , lo cual genera la imagen de Haar. Para ello debemos rotar la imagen **s** y aplicar nuevamente el mismo algoritmo



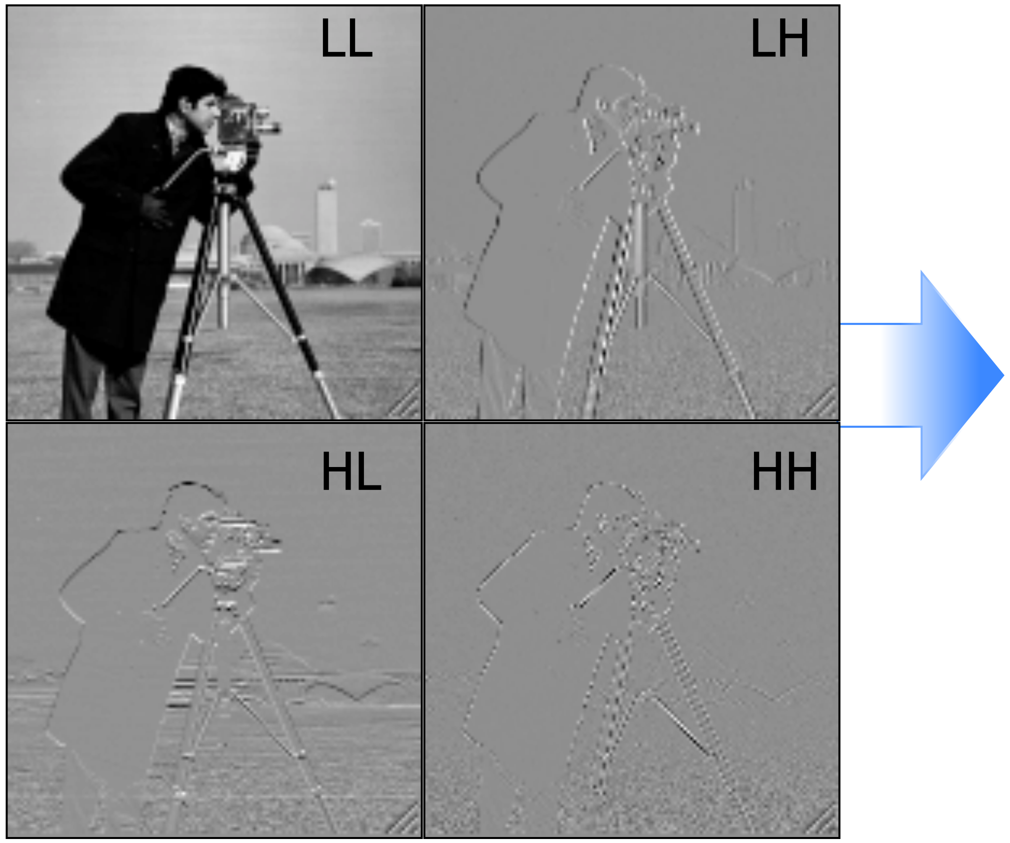
Codificación de Haar

- Aplicación en imágenes
  - El resultado de la descomposición se muestra de la siguiente forma:



INVERSA

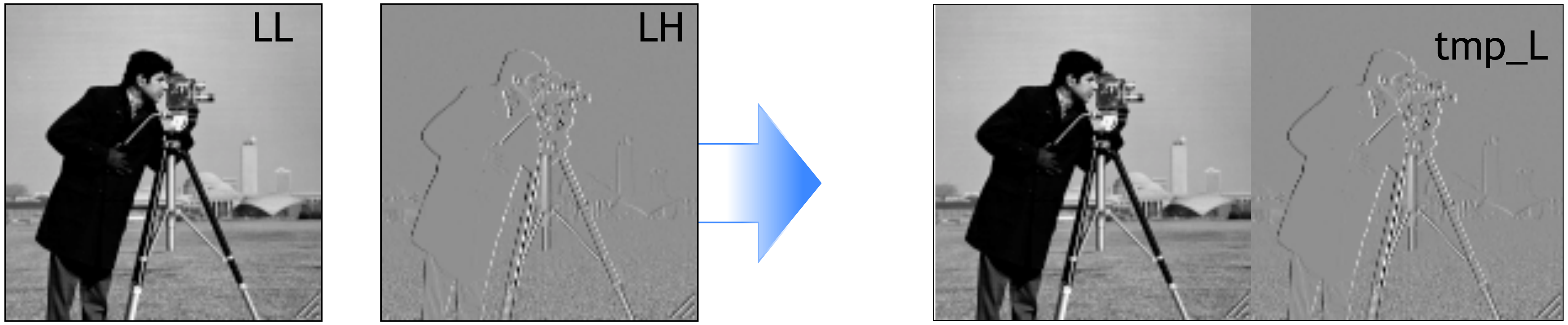
- Aplicación en imágenes
  - Ahora analicemos el proceso inverso. Es decir, a partir de la descomposición restauremos la imagen original



Objetivo

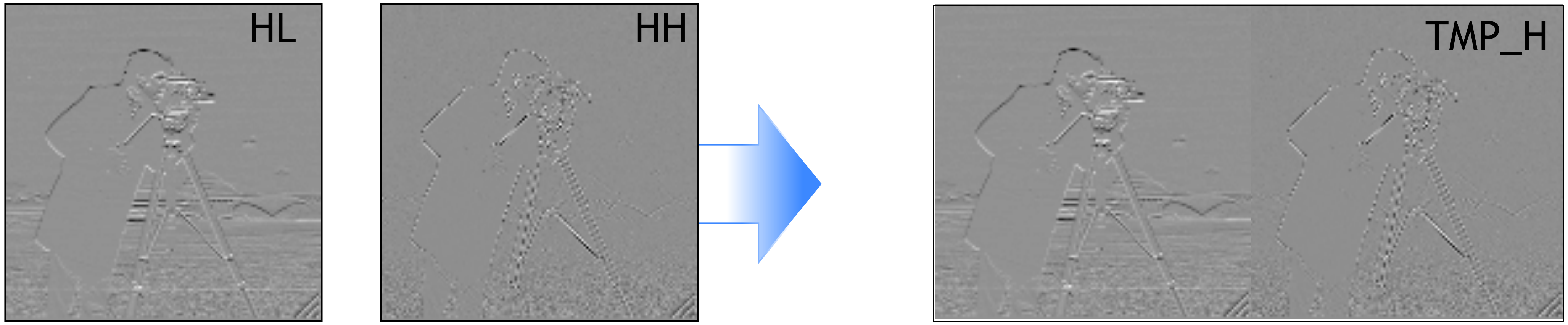
INVERSA

- Aplicación en imágenes
  - El primer paso es componer las imágenes LL y LH en una matriz



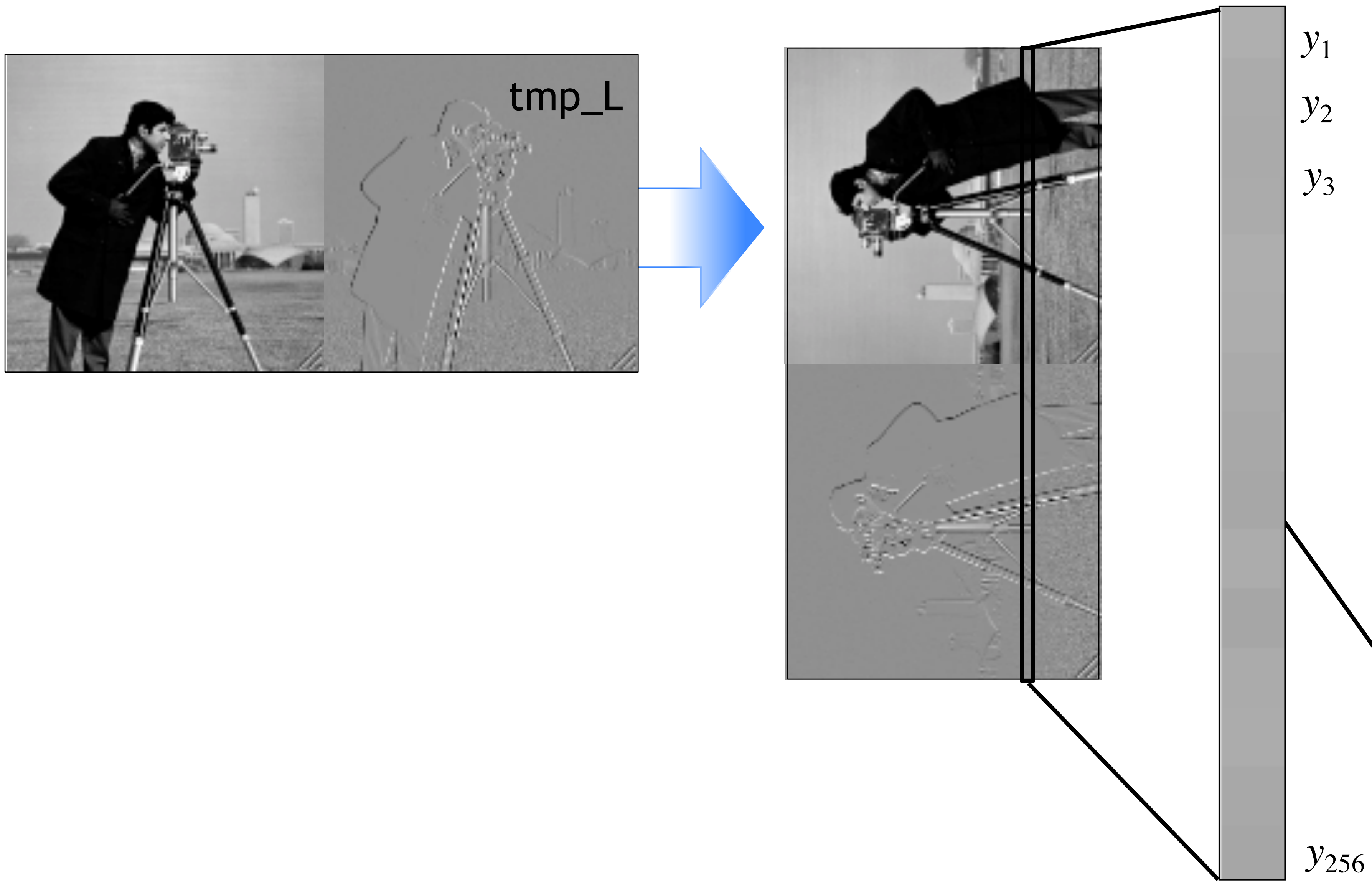
INVERSA

- Aplicación en imágenes
  - El primer paso es componer las imágenes HL y HH en una matriz



INVERSA

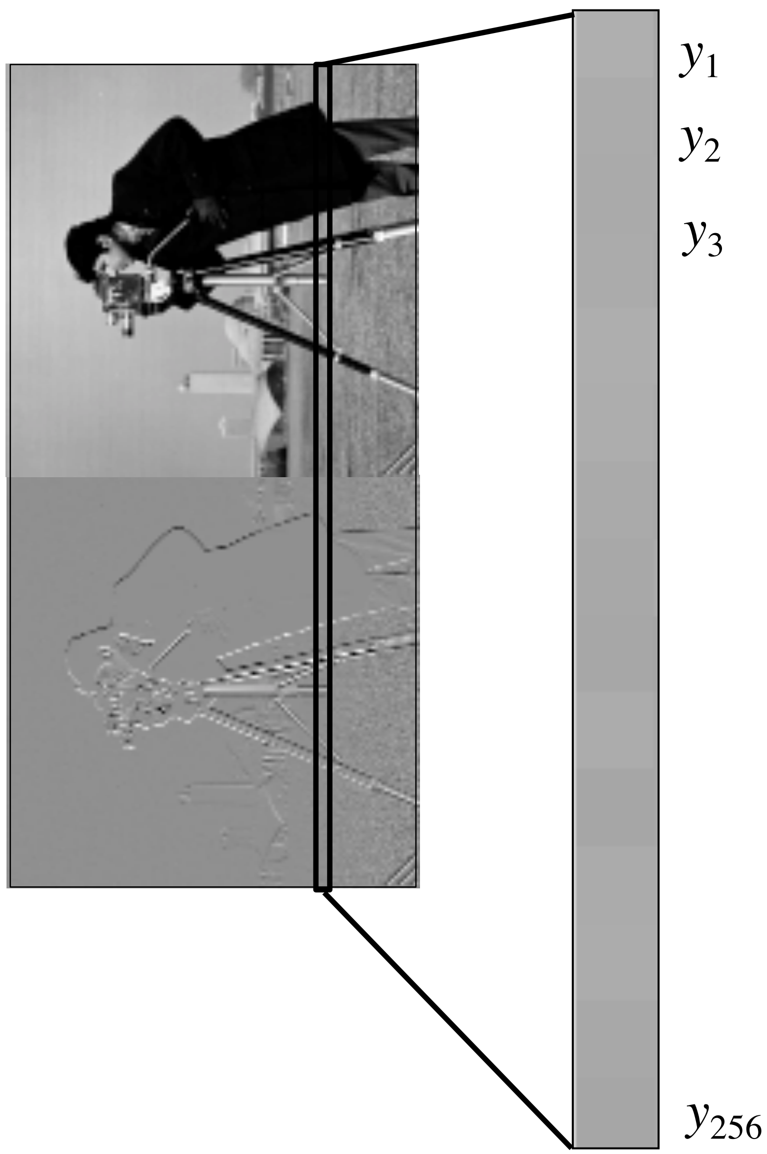
- Aplicación en imágenes
  - El segundo paso es componer emplear el algoritmo de transformación inversa. Para ello debemos rotar la matriz



El algoritmo ahora va a operar por columnas, al igual que al inicio

INVERSA

- Aplicación en imágenes
  - Por cada columna de la nueva imagen, debemos reordenarla de la siguiente forma:



Bases

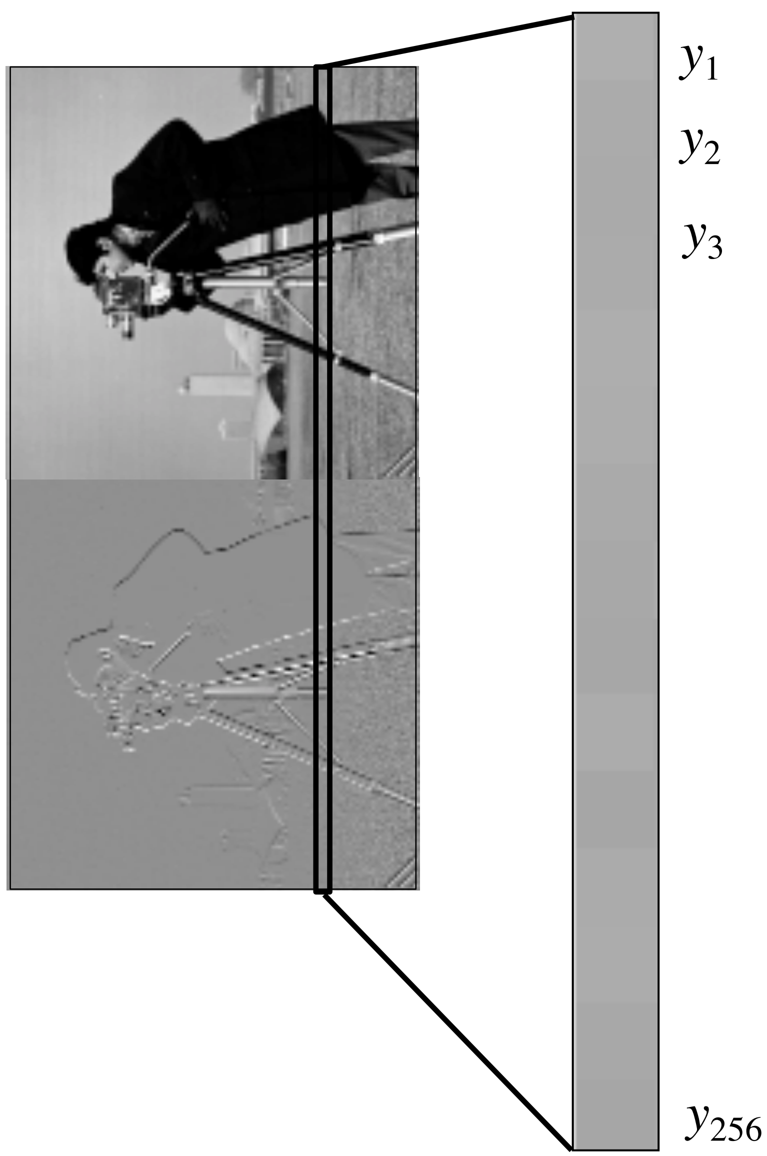
$$\begin{bmatrix} y_1 & y_5 \\ y_2 & y_6 \\ \vdots & \vdots \\ y_{\frac{N}{2}} & y_N \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} y_1 - y_5 \\ y_2 - y_6 \\ \vdots \\ y_{\frac{N}{2}} - y_N \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{\frac{N}{2}} \end{bmatrix}$$

$$\begin{bmatrix} y_1 & y_5 \\ y_2 & y_6 \\ \vdots & \vdots \\ y_{\frac{N}{2}} & y_N \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} y_1 + y_5 \\ y_2 + y_6 \\ \vdots \\ y_{\frac{N}{2}} + y_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{\frac{N}{2}} \end{bmatrix}$$

Debemos reodenar la matriz de esta forma

INVERSA

- Aplicación en imágenes
  - El único problema es que el vector de salida tiene los valores intercalados

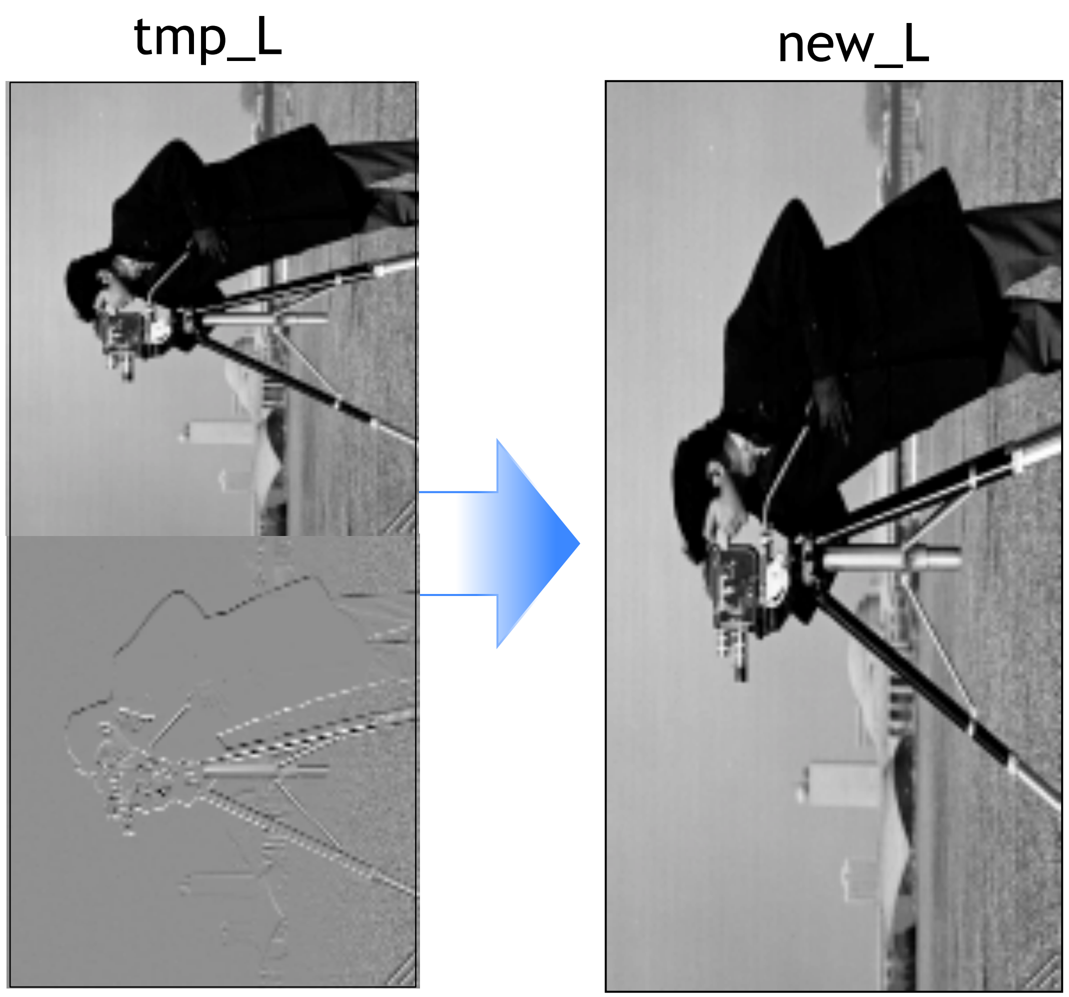


$$\mathbf{x} = \begin{bmatrix} y_1 - y_{\frac{N}{2}+1} \\ y_1 + y_{\frac{N}{2}+1} \\ y_2 - y_{\frac{N}{2}+2} \\ y_2 + y_{\frac{N}{2}+2} \\ \vdots \\ y_{\frac{N}{2}} - y_N \\ y_{\frac{N}{2}} + y_N \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} a_1 \\ b_1 \\ a_2 \\ b_2 \\ \vdots \\ a_{\frac{N}{2}} \\ b_{\frac{N}{2}} \end{bmatrix}$$

La solución es intercalar los valores obtenidos anteriormente

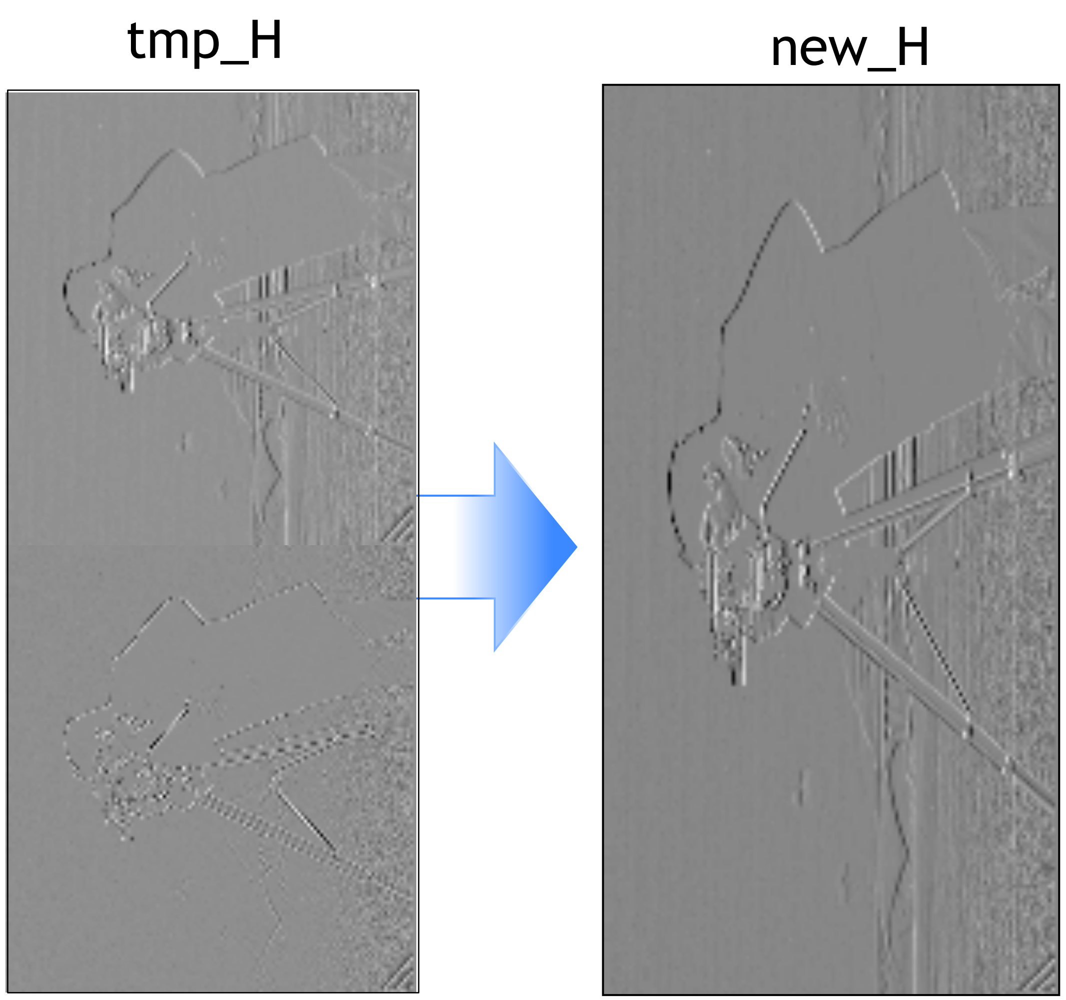
INVERSA

- Aplicación en imágenes
  - El tercer paso es componer las dos matrices empleando las bases de composición.



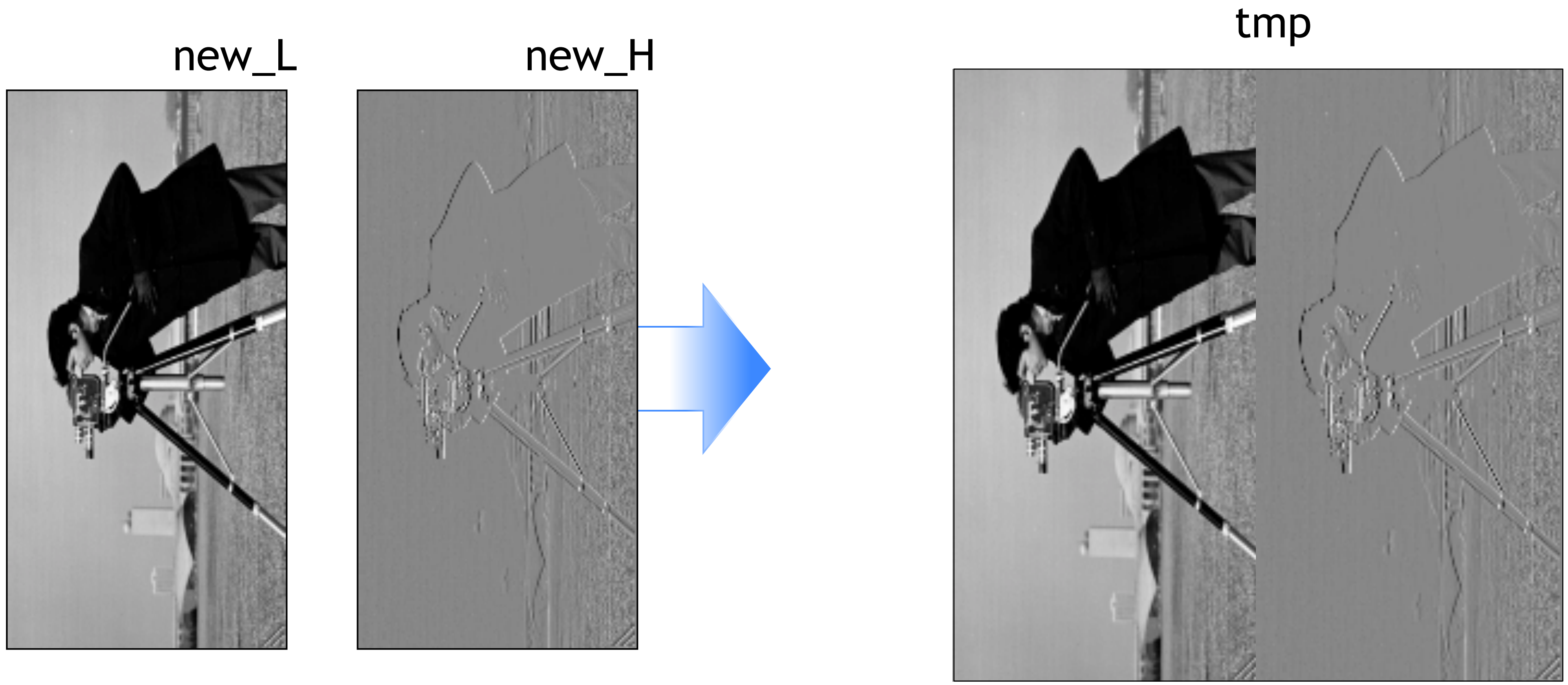
INVERSA

- Aplicación en imágenes
  - El tercer paso es componer las dos matrices empleando las bases de composición



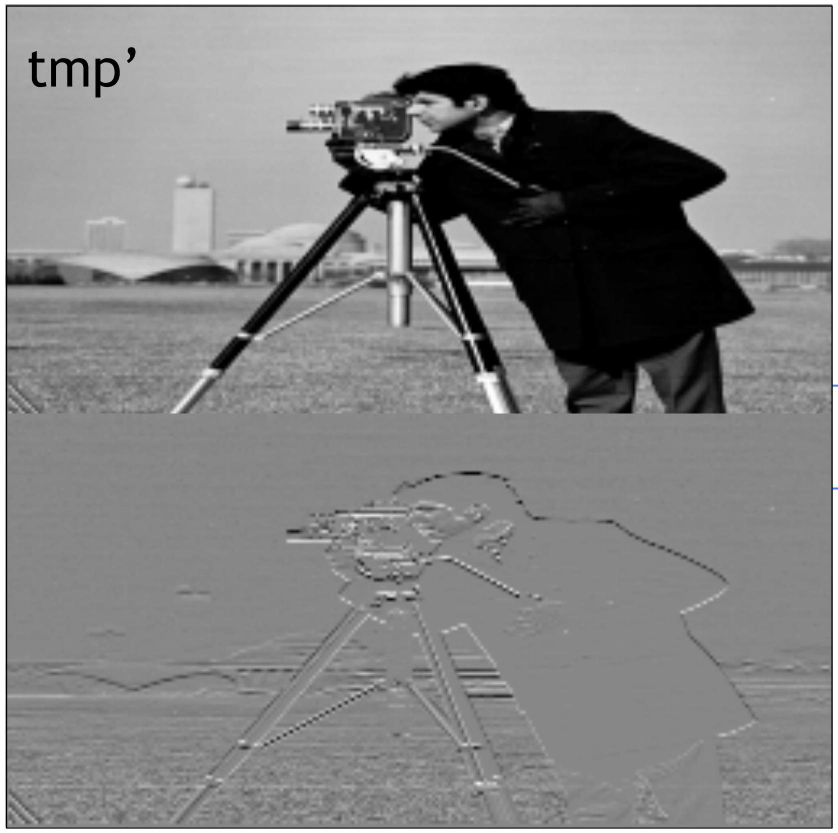
INVERSA

- Aplicación en imágenes
  - El cuarto paso es unir las dos matrices en una sola empleando el algoritmo del primer paso



INVERSA

- Aplicación en imágenes
  - El quinto paso es rotar la matriz y emplear el mismo algoritmo del paso 3

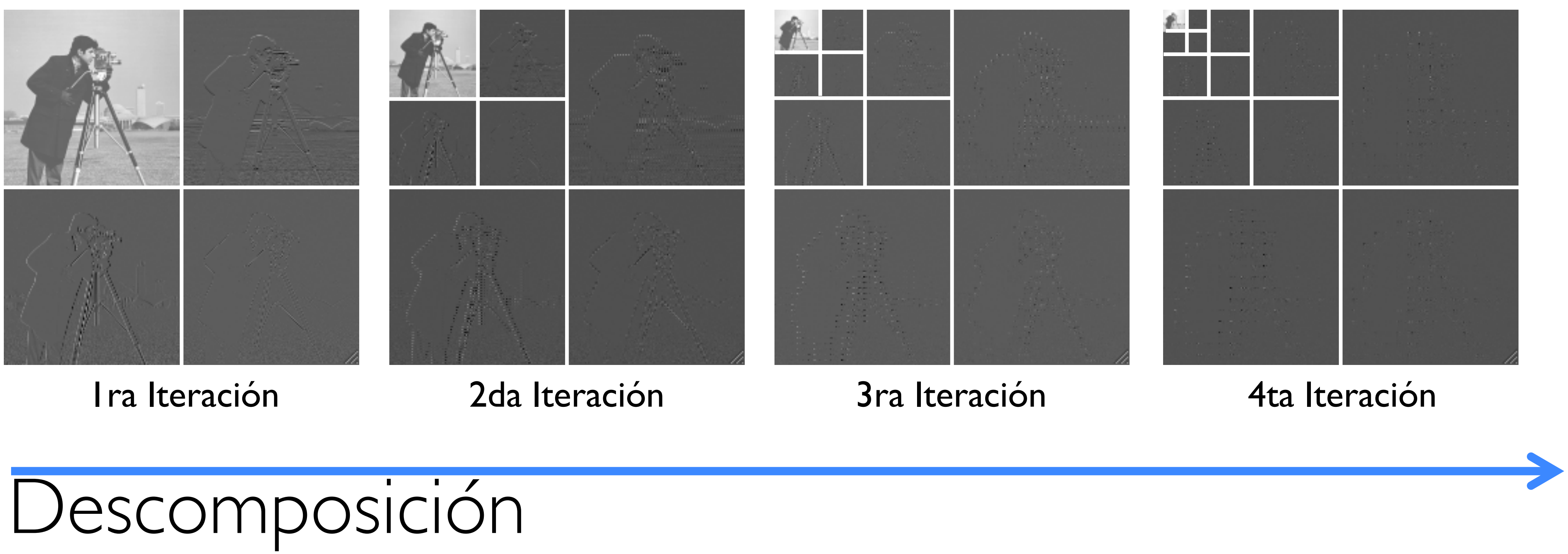


Temario

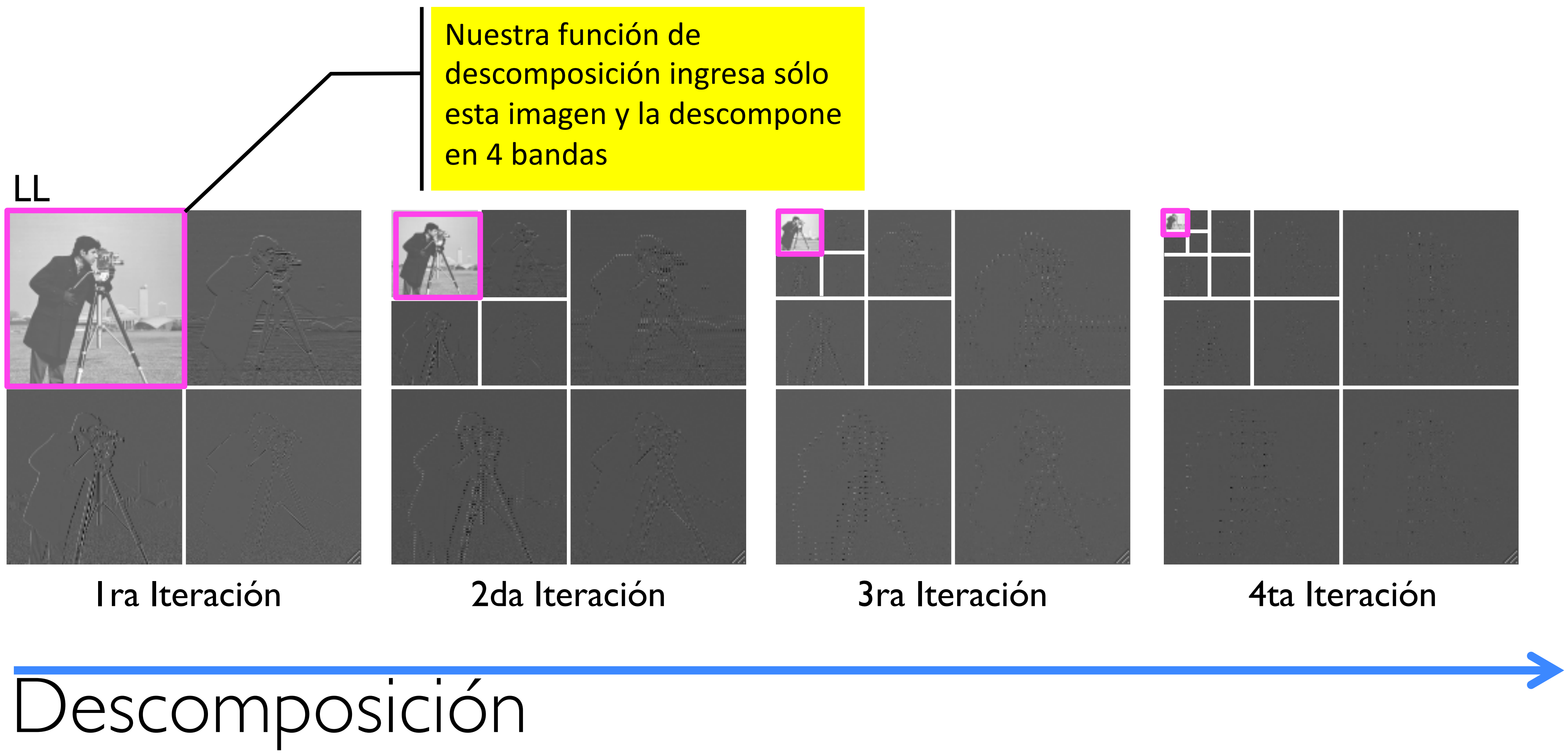
- Multiresolución y Compresión
    - Ejemplo en imágenes
    - Multidescomposición
    - Reducción del ruido
    - Resultado

Descomposición múltiple

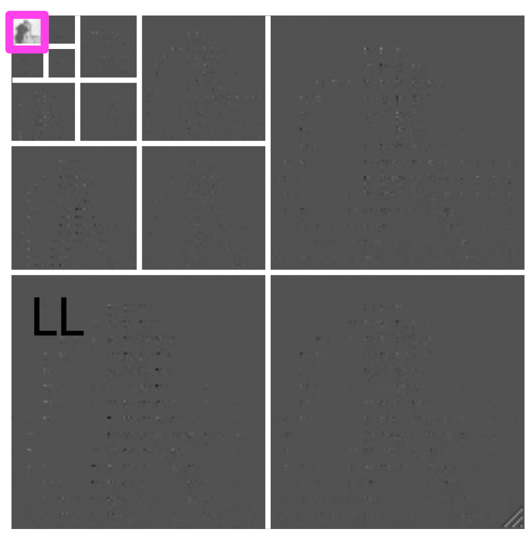
- Procesamiento múltiple
  - La división anterior nos permite efectuar un análisis en una división.
  - Para mejorar el rendimiento de nuestro filtro, debemos seguir expandiendo la descomposición en múltiples subniveles



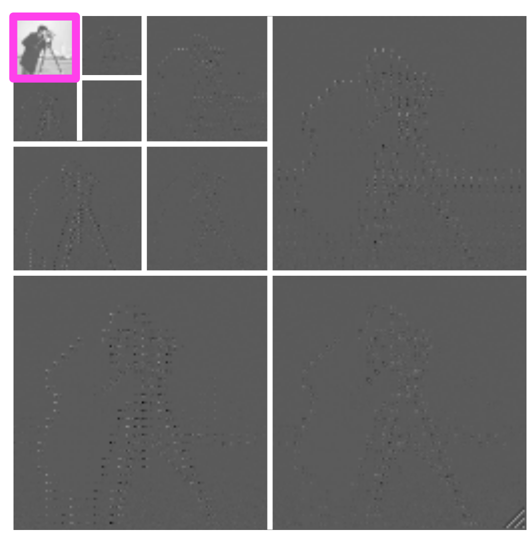
- Procesamiento múltiple
  - La solución simplemente consiste en descomponer nuevamente el cuadro LL en nuevas subbandas



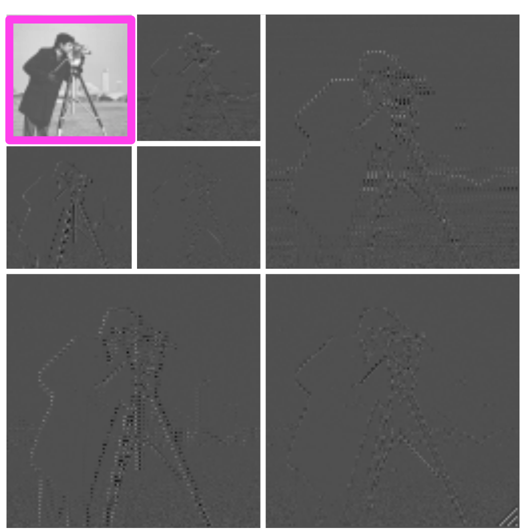
- Procesamiento múltiple
  - Ahora vamos a realizar el proceso inverso para componer los cuadrantes.



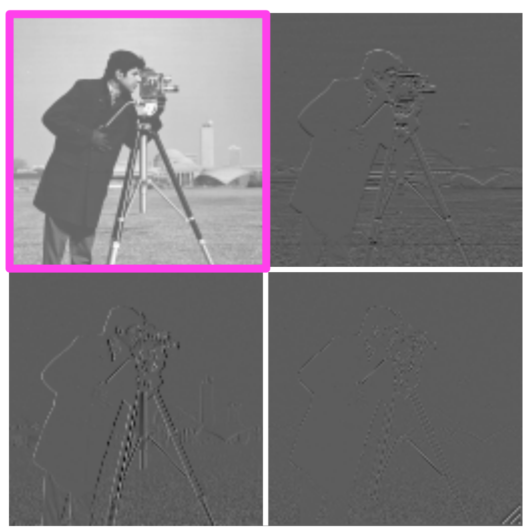
4ta Iteración



3ra Iteración



2da Iteración



1ra Iteración

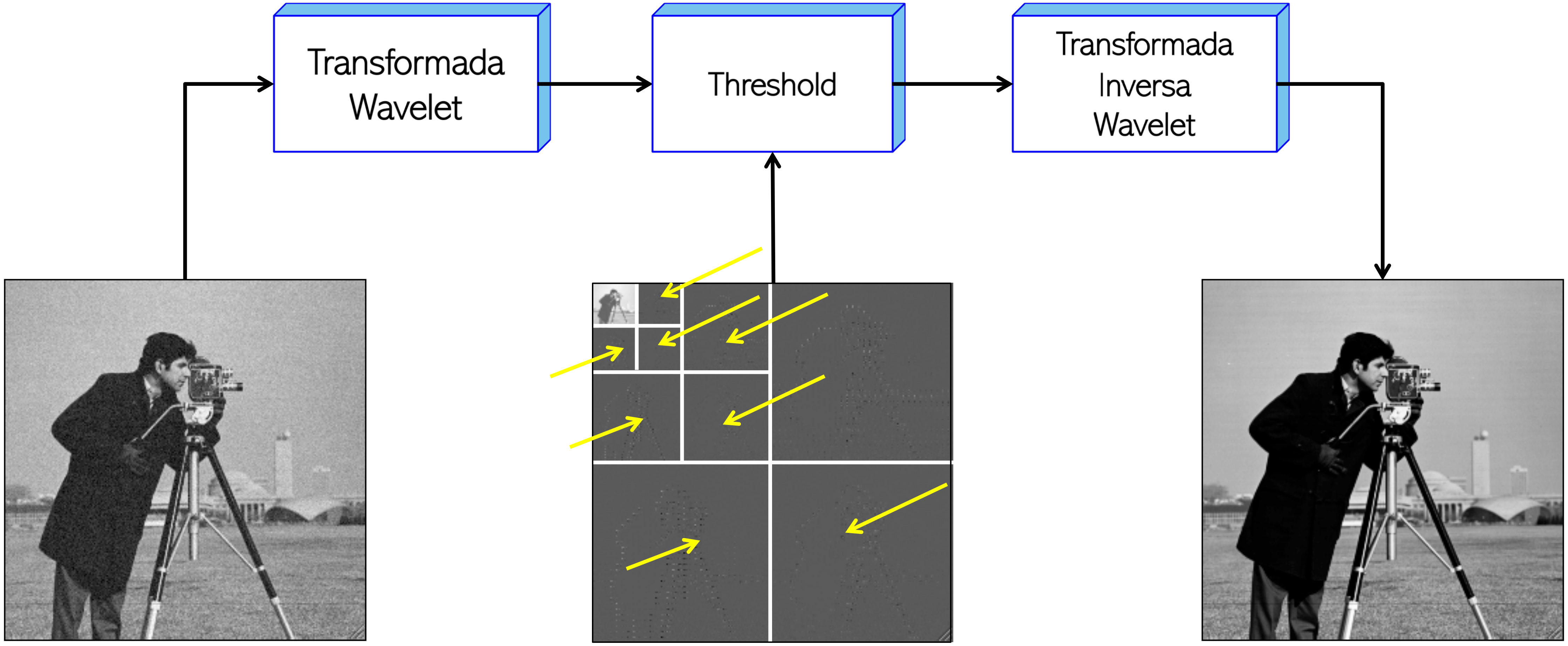
Composición

Resultado.

Temario

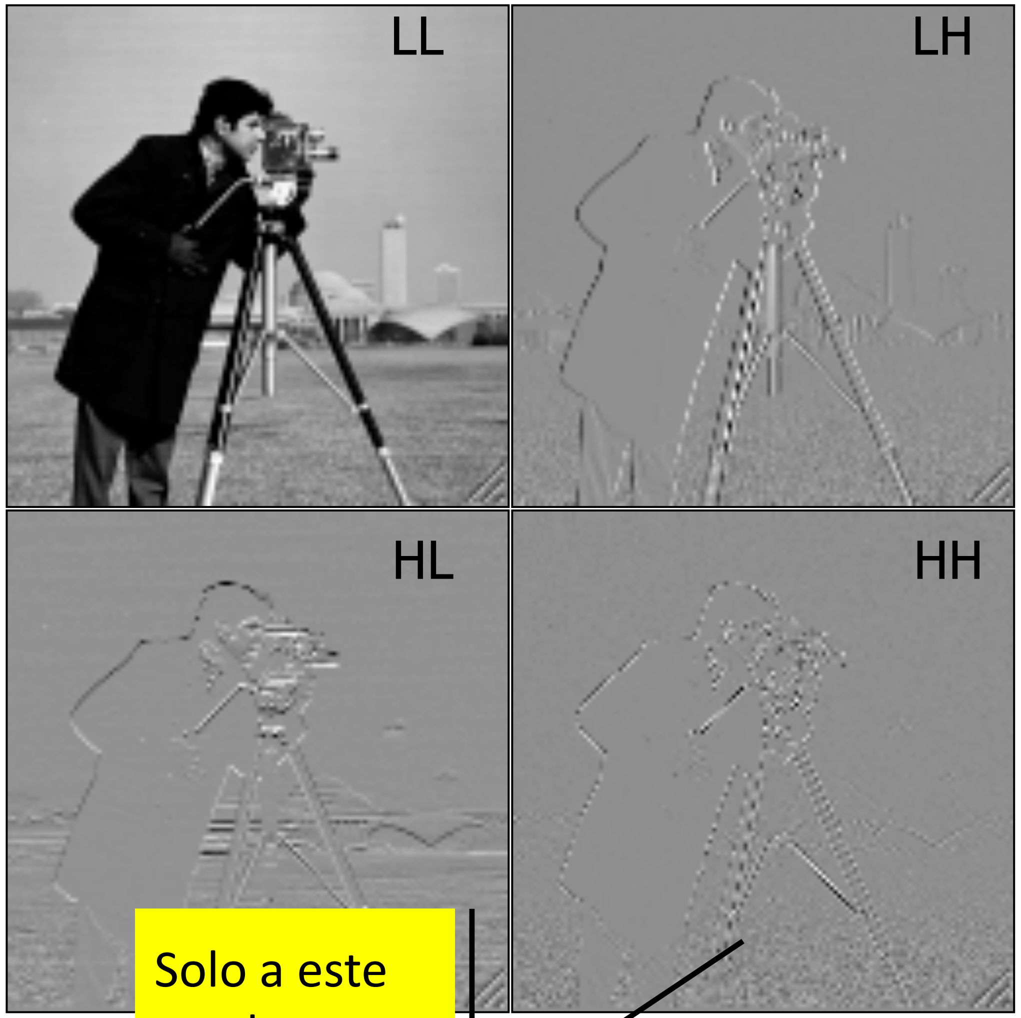
- Multiresolución y Compresión
  - Ejemplo en imágenes
  - Multidescomposición
  - Reducción del ruido
  - Resultado

- Reducción de ruido gaussiano blanco
  - Para reducir el ruido el proceso consiste en filtrar las sub-bandas de los factores HH, LH, HL. En la literatura existen distintas formas para efectuar este proceso, pasando desde los más simples a otros muy complejos.
  - El proceso en general es el siguiente.



Reducción del ruido

- Reducción de ruido gaussiano blanco
  - El primer paso es estimar el ruido de la imagen. Gracias al trabajo de los investigadores Donoho y Jonhstone (1993) existe una forma para estimar el ruido. Este ruido solo se determina en el cuadro HH de la primera descomposición.



Solo a este cuadro

$$\sigma = \left(\frac{\text{mediana}\left(\left|\text{HH}_1\right|\right)}{0.6745} \right)$$

- Reducción de ruido gaussiano blanco
  - Para filtrar vamos a ocupar la misma idea que realizamos para descomponer. Vamos a comenzar desde la subbanda más pequeña y ocuparemos el siguiente filtro por cada imagen (excepto el bloque LL)

imagen del bloque

estimación del ruido

```
y = np.maximum(np.abs(tmp)-nse,0)
output = y/(y+nse)* tmp
```

$$y = \max(|img| - \sigma, 0)$$

Esto genera una nueva matriz

$$output = \frac{y}{(y+\sigma)} \cdot img$$

Esto es una imagen filtrada

Temario

- Multiresolución y Compresión
  - Ejemplo en imágenes
  - Multidescomposición
  - Reducción del ruido
  - Resultado

- Resultado del proceso
  - Como podemos notar la imagen mejora notablemente. Además podemos aplicar otros filtros si queremos mejorar aún mas.

Imagen original
sigma 0.01

Imagen filtrada con Haar (8 iter)
Bases dividida $\sqrt{2}$

Post procesamiento
con filtro adaptivo local