Universidad
Popular del Cesar

Tratamiento de Señales

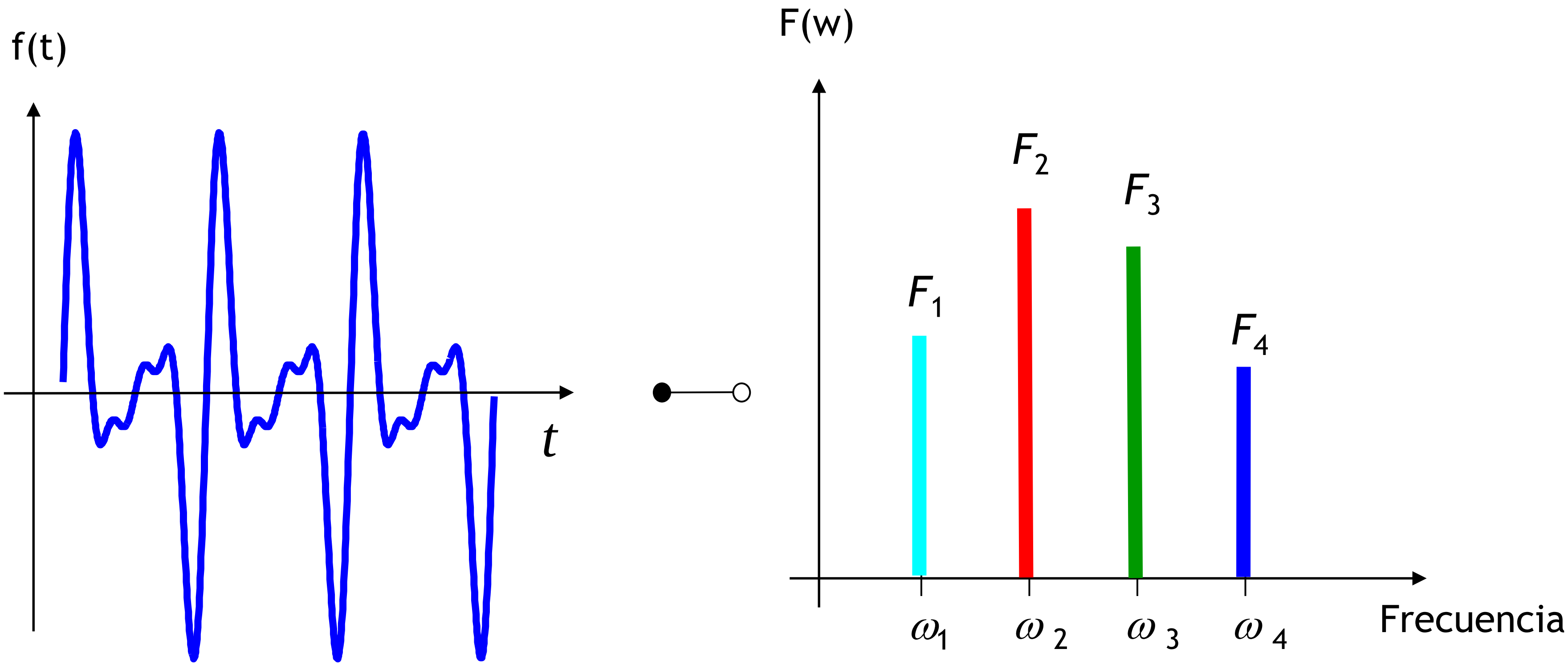
Version 2024-I

Transformada de Fourier en 2D

[Capítulo 4]

Dr. José Ramón Iglesias
DSP-ASIC BUILDER GROUP
Director Semillero TRIAC
Ingenieria Electronica
Universidad Popular del Cesar

Transformada de Fourier en 1D



Señal $f(t)$

Espectro de la señal $F(\mathrm{w})$

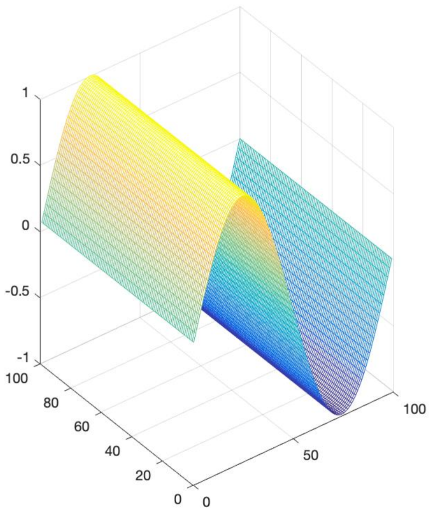
$$f(t) \;\bullet\!\!-\!\!\!-\!\!\circ\; F(\mathrm{w})$$

¿Cómo sería la Transformada de Fourier en 2D?

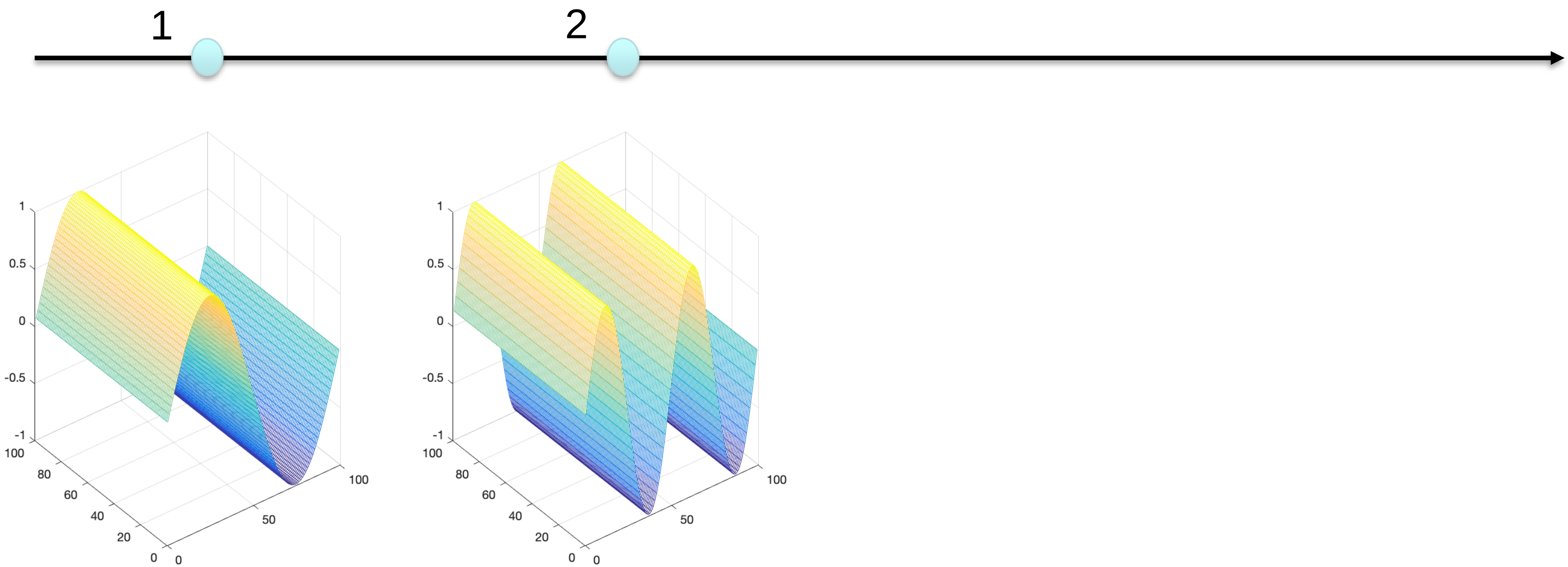
Se define una frecuencia horizontal

Se define una frecuencia horizontal

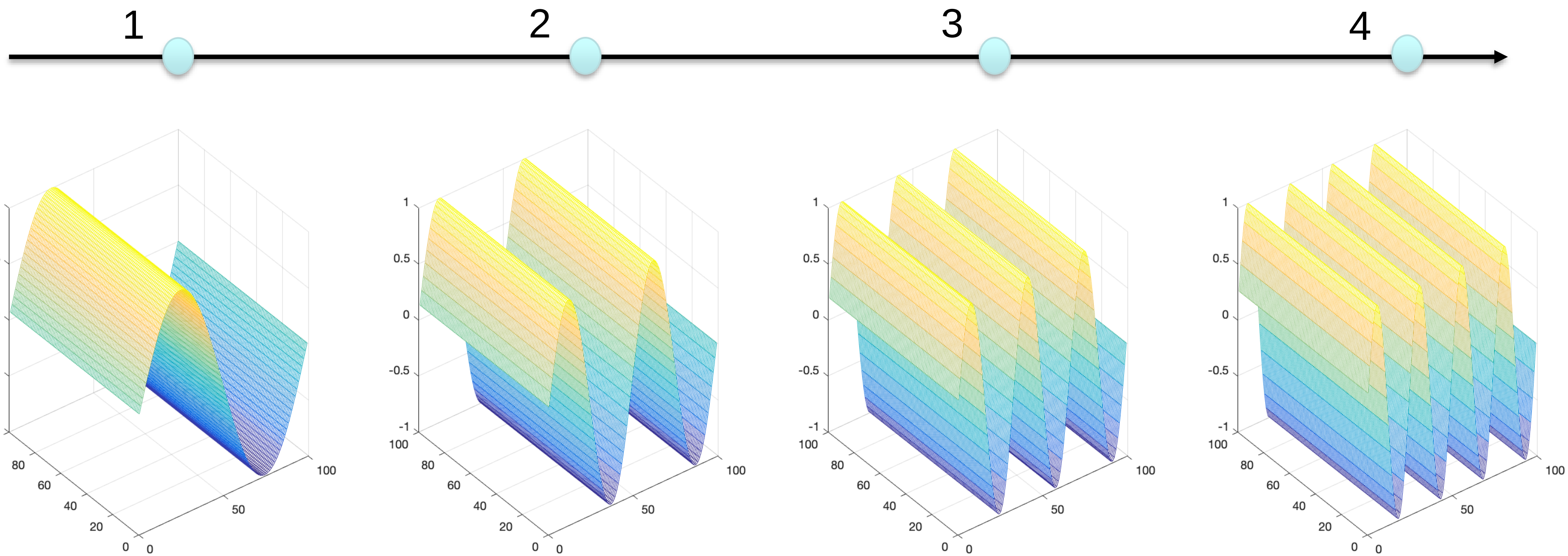
1



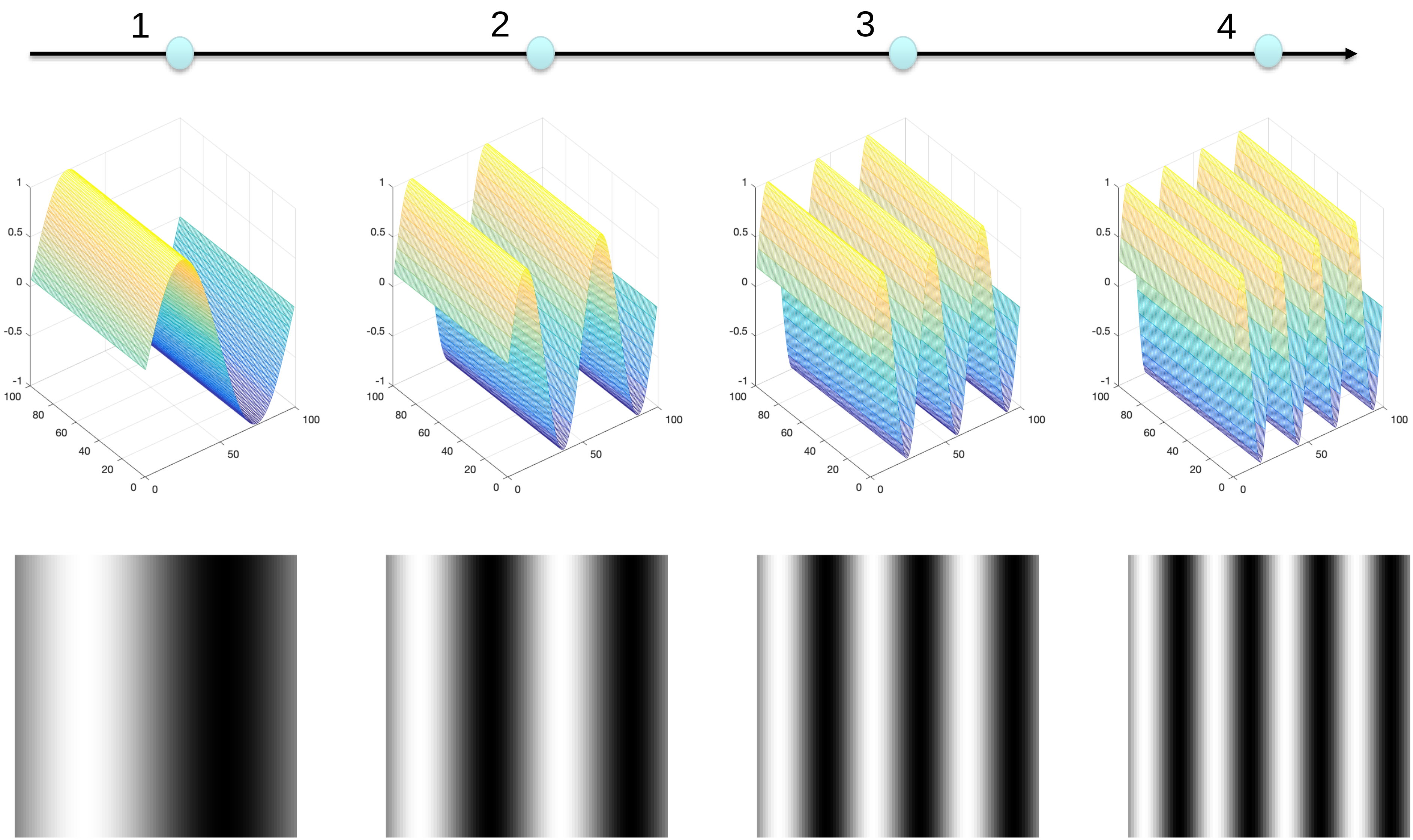
Se define una frecuencia horizontal



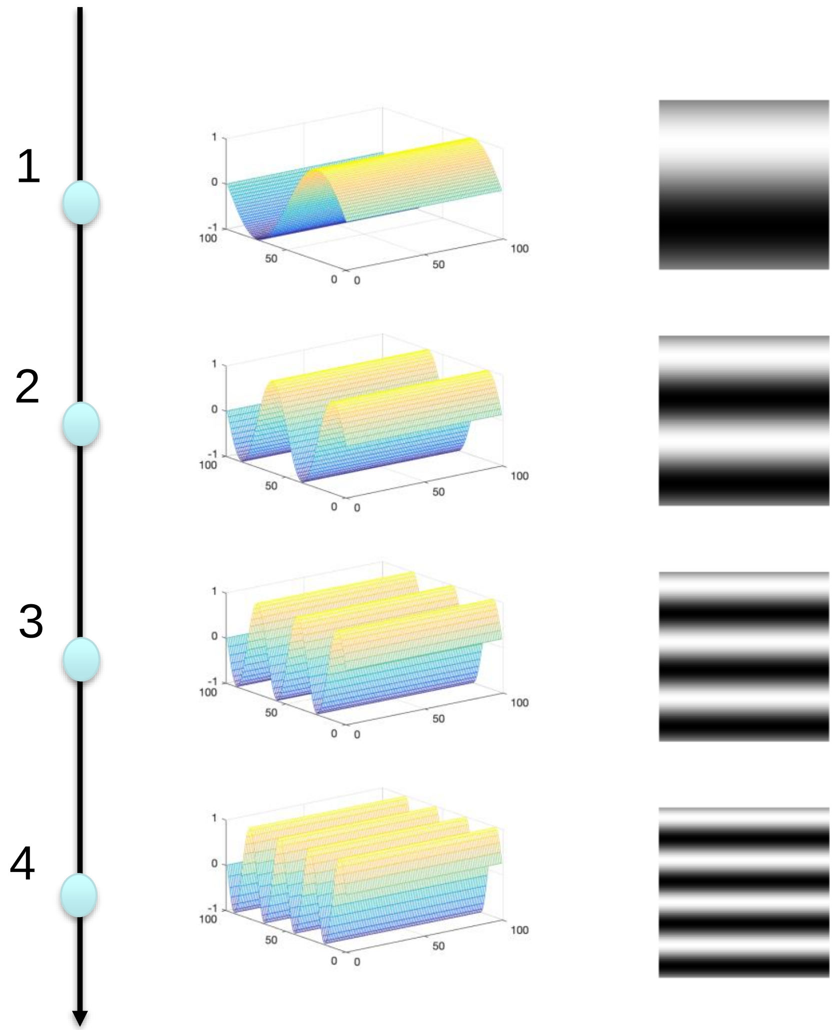
Se define una frecuencia horizontal



Se define una frecuencia horizontal

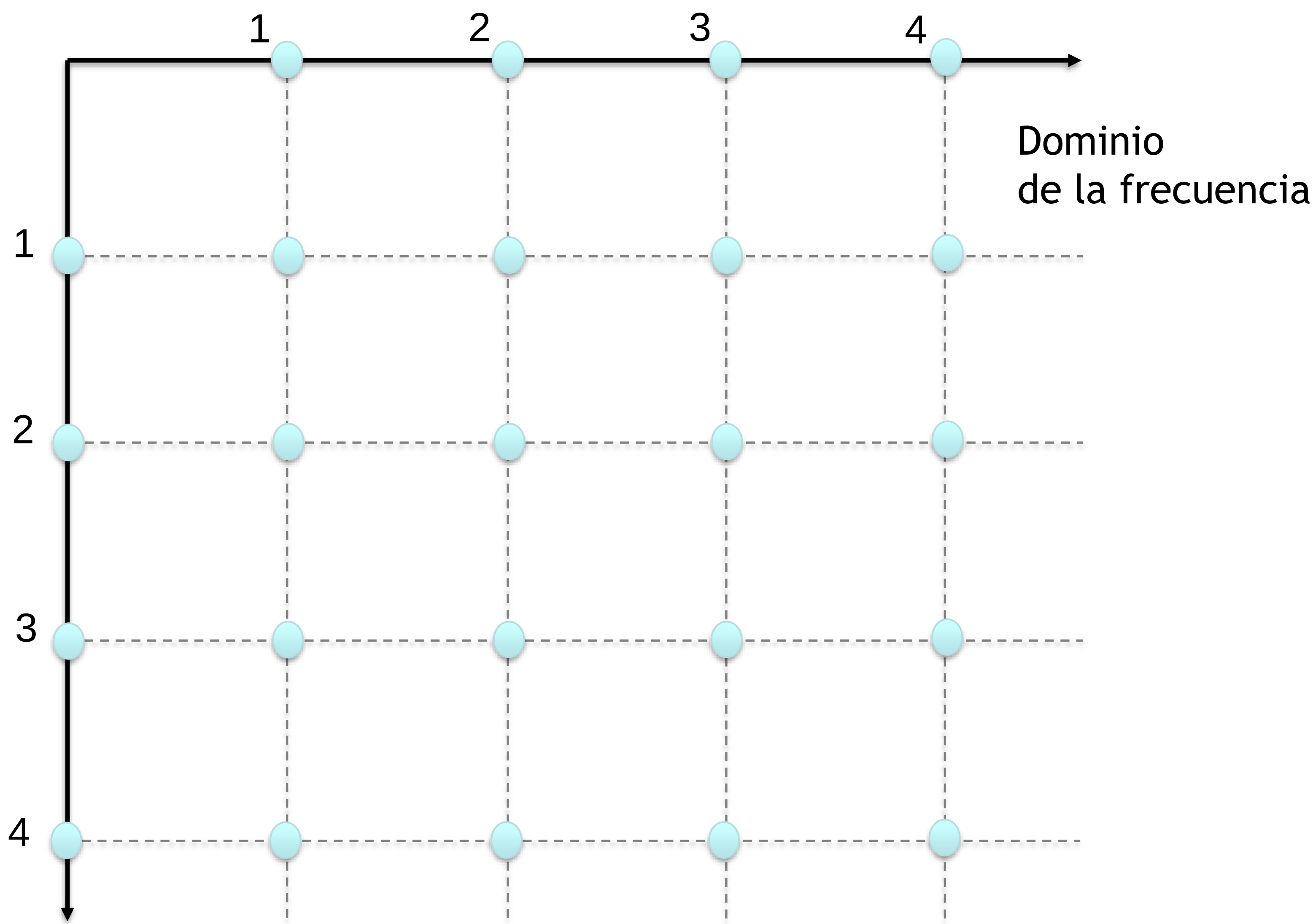


De la misma manera, se define una frecuencia vertical

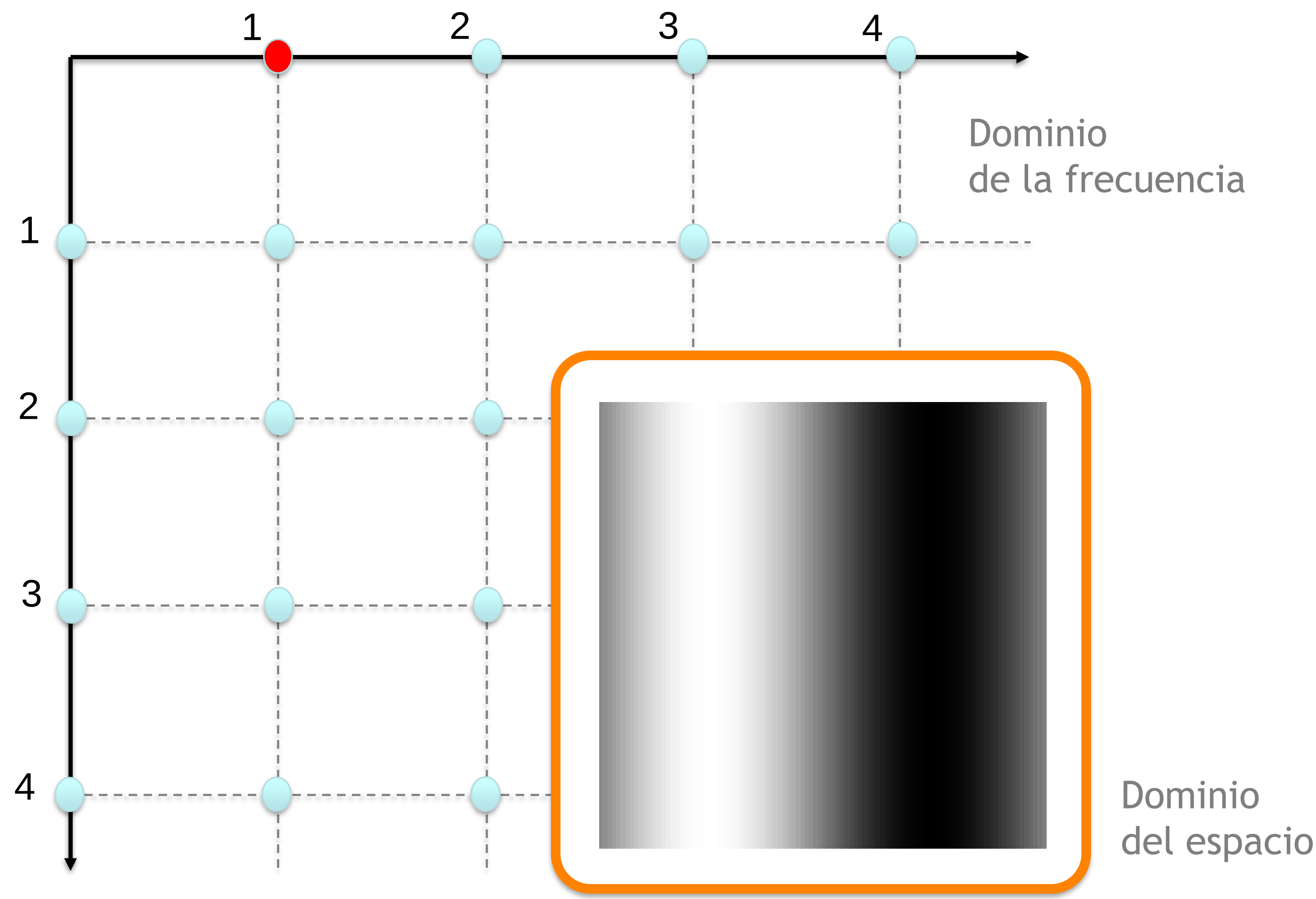


Ahora definimos un espacio de dos frecuencias

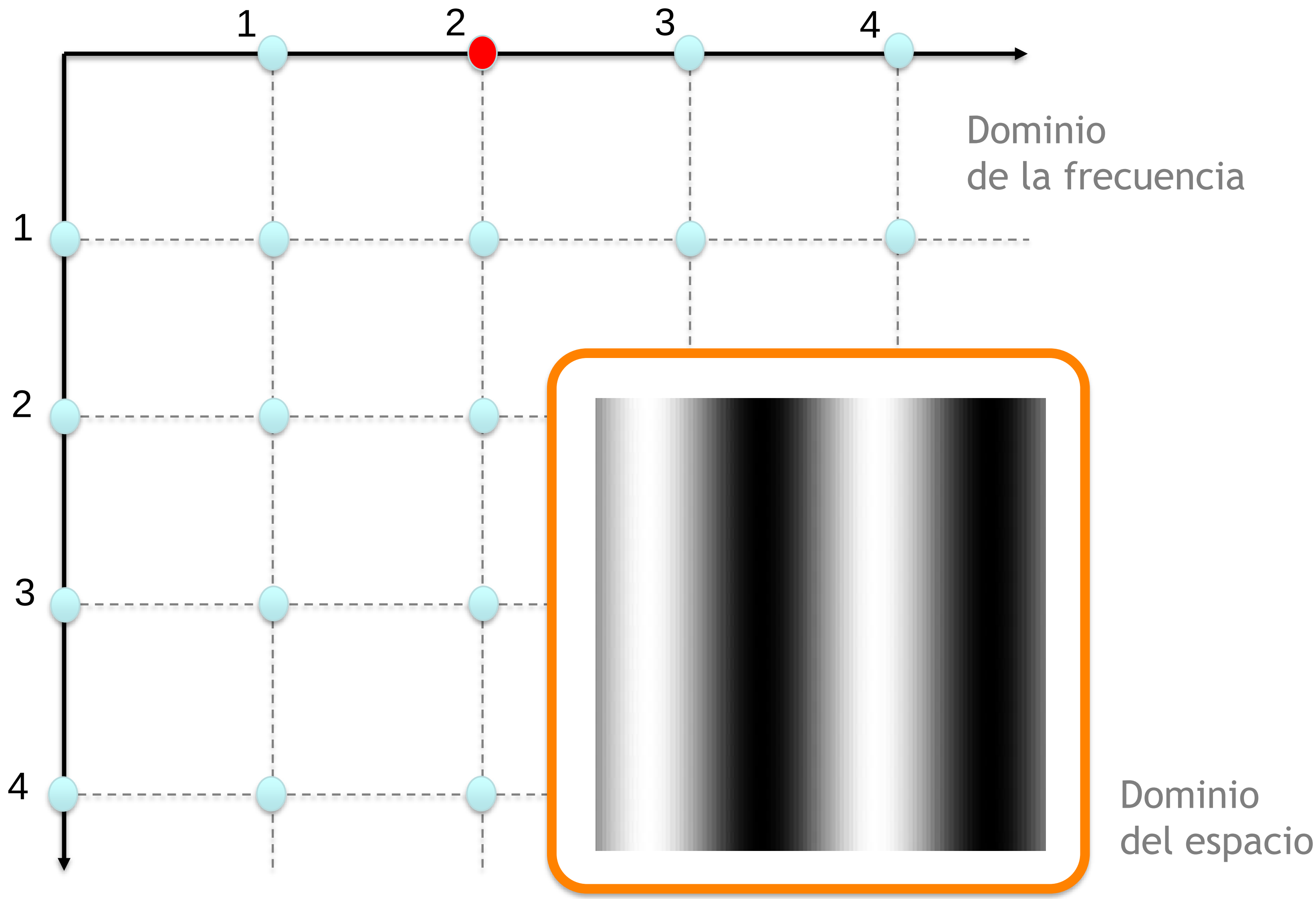
Espacio de dos frecuencias



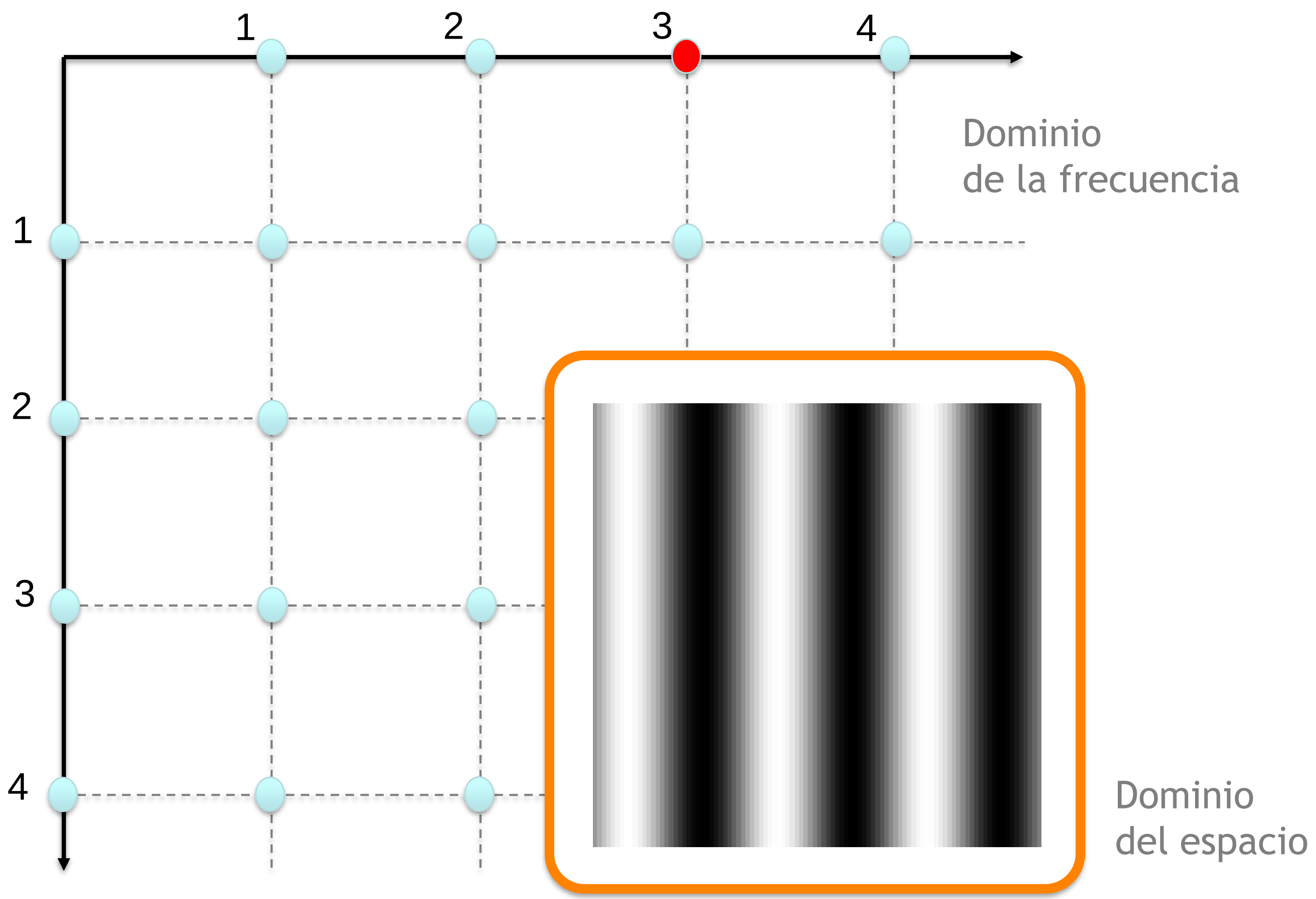
Espacio de dos frecuencias



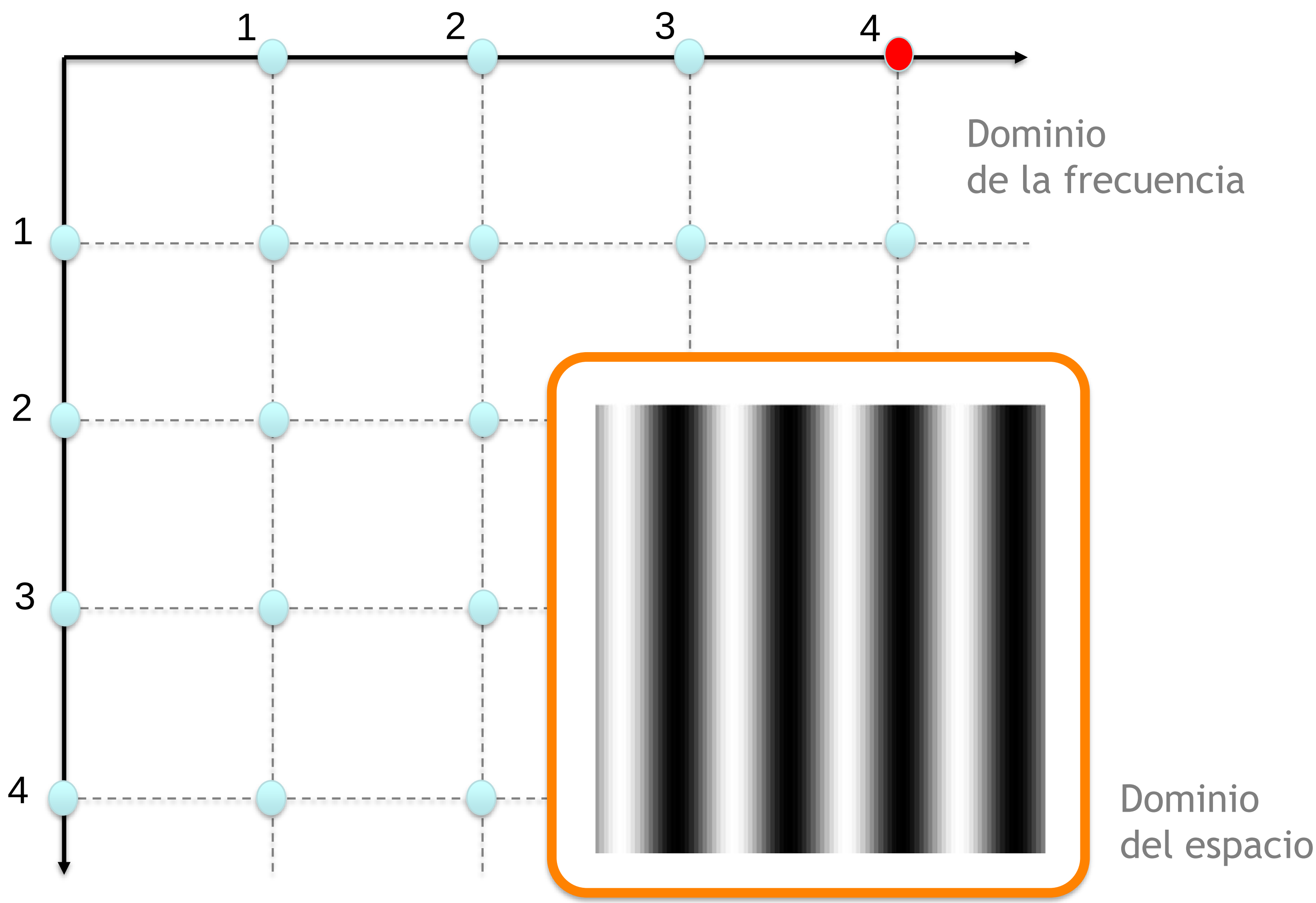
Espacio de dos frecuencias



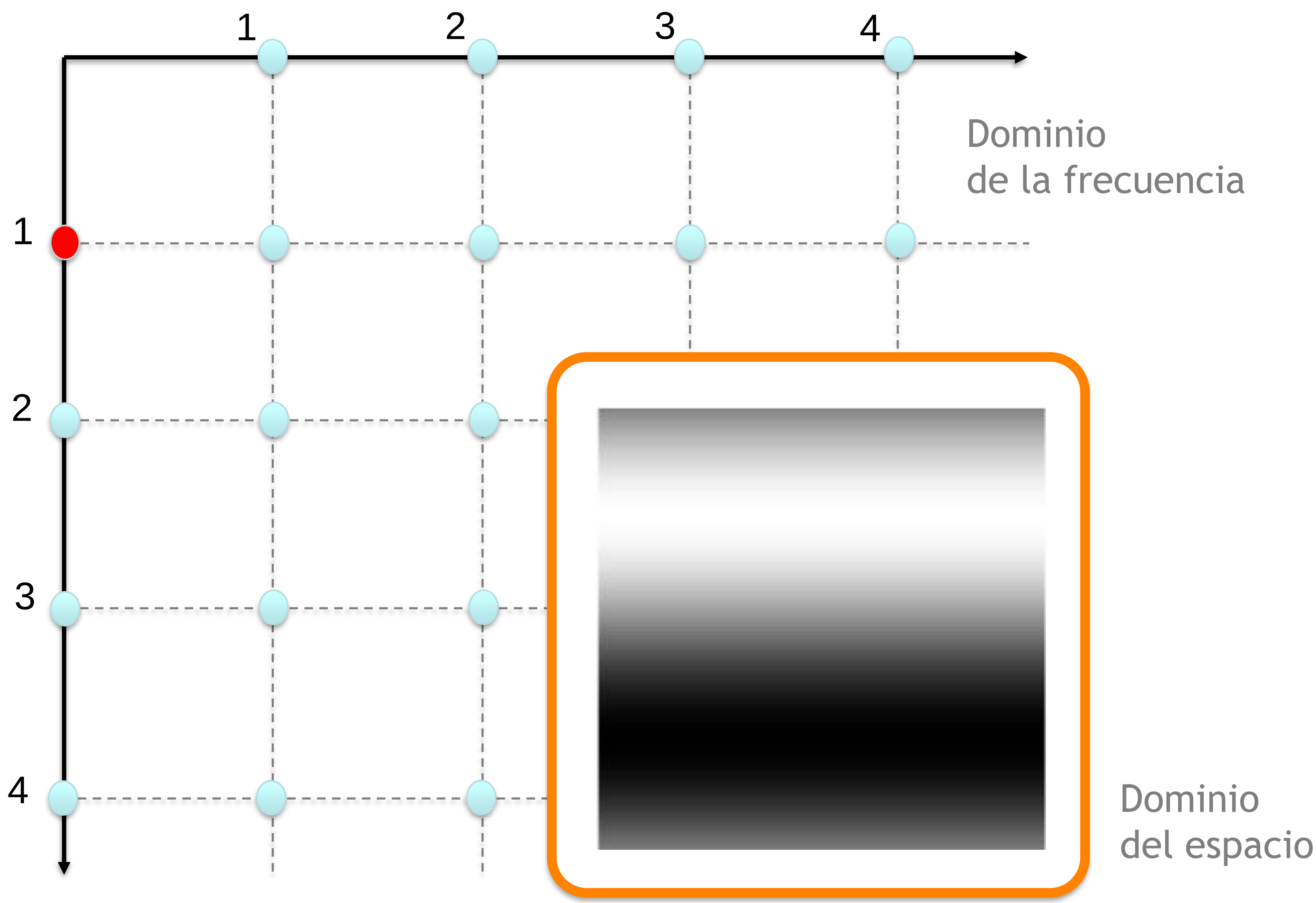
Espacio de dos frecuencias



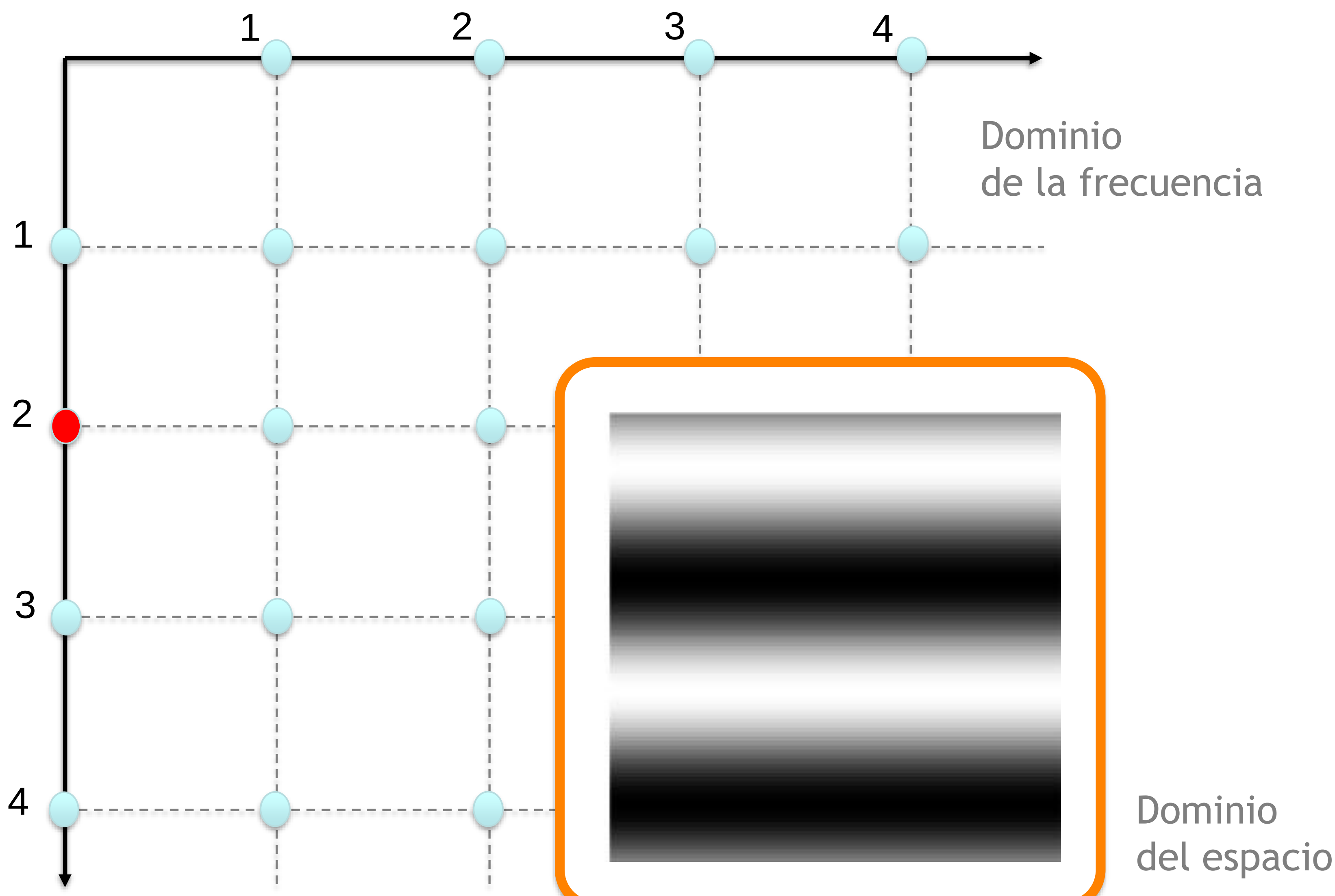
Espacio de dos frecuencias



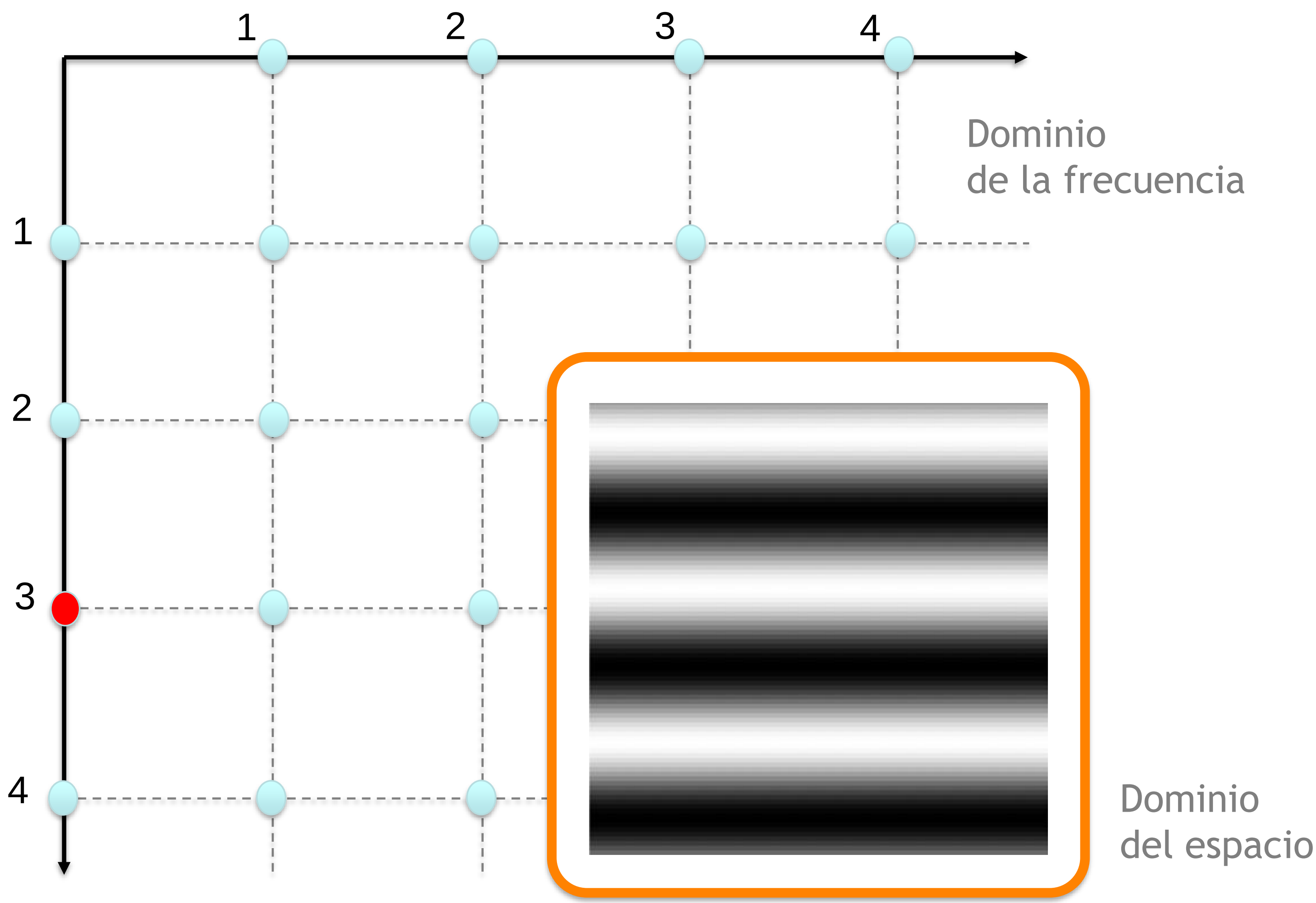
Espacio de dos frecuencias



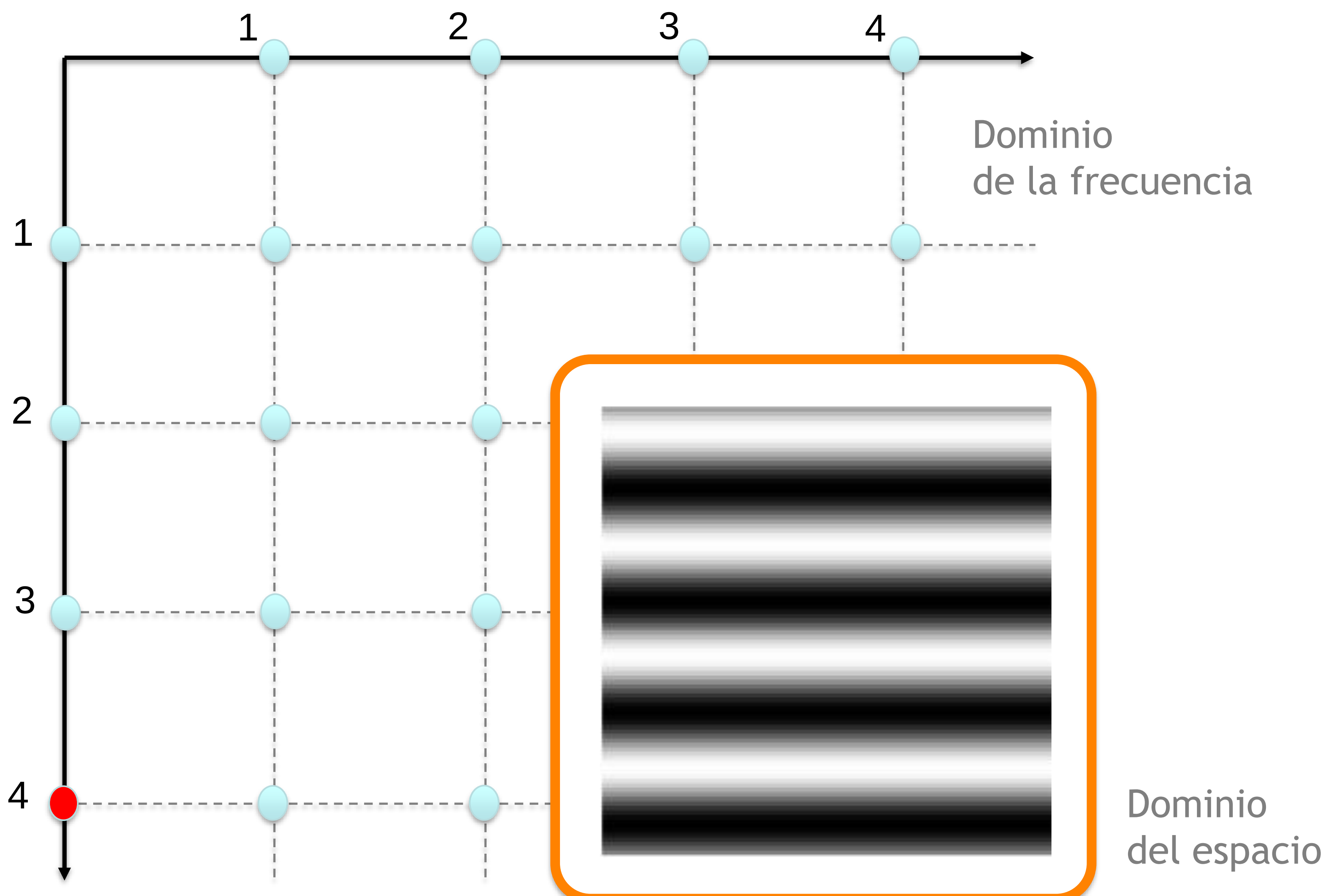
Espacio de dos frecuencias



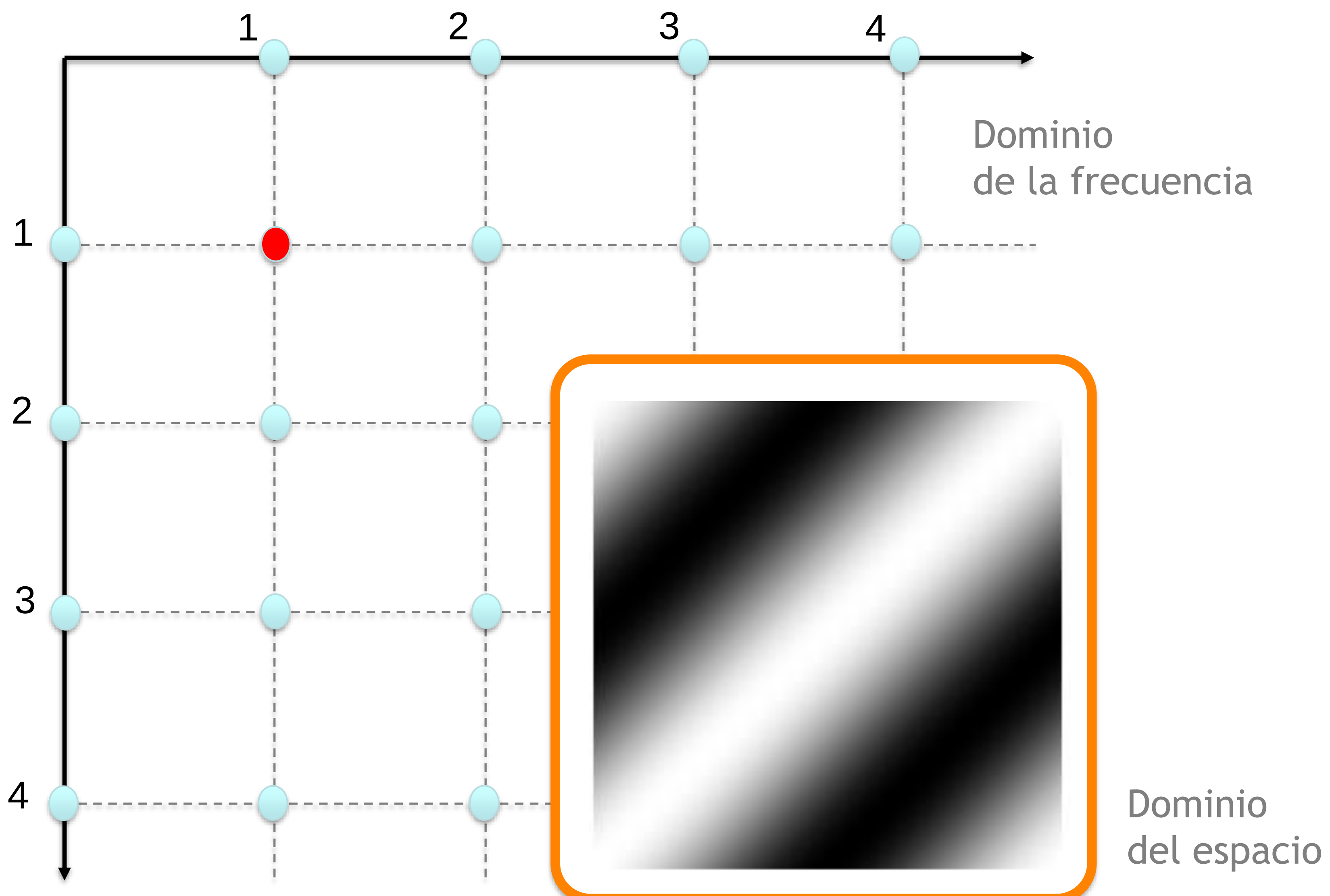
Espacio de dos frecuencias



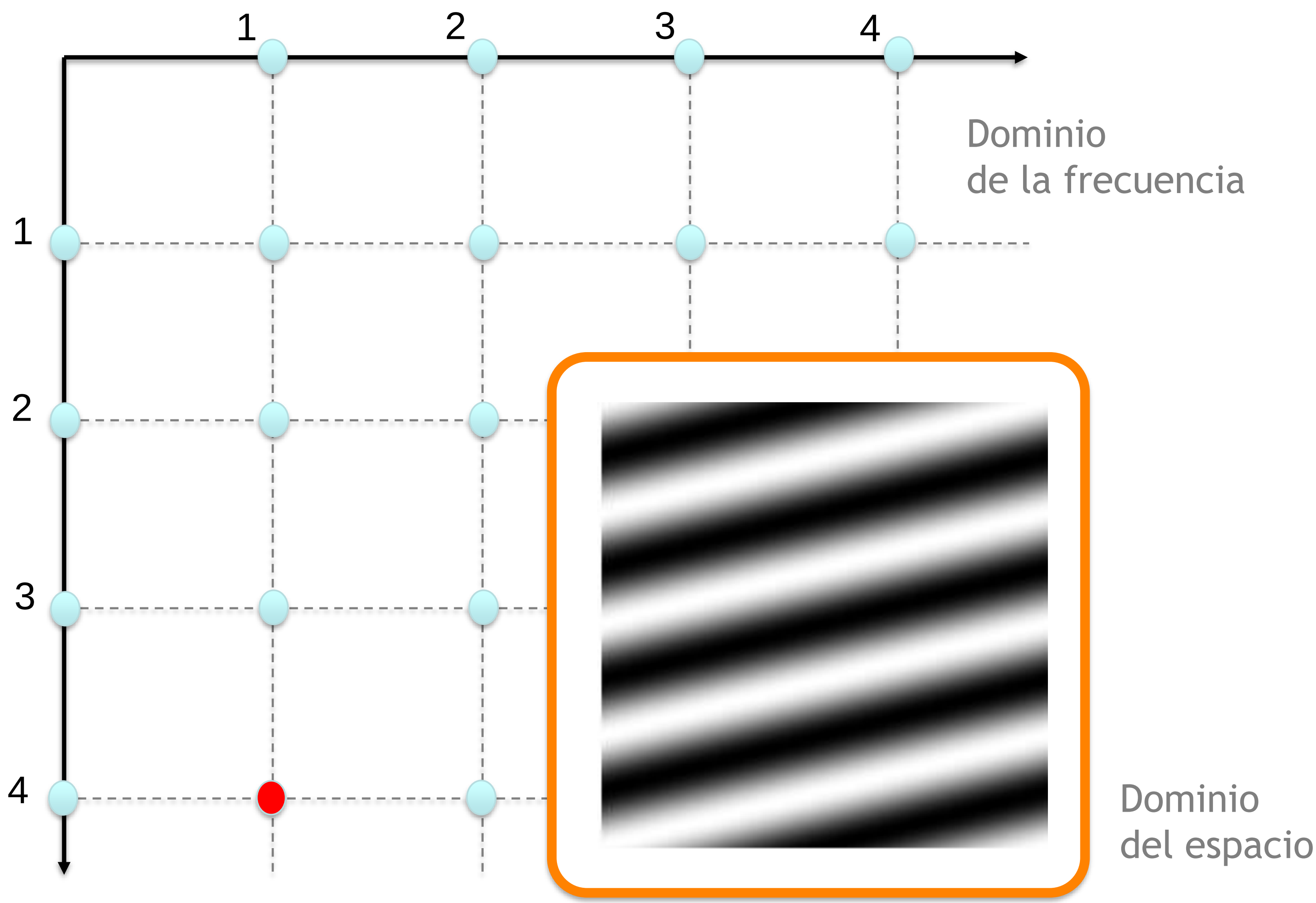
Espacio de dos frecuencias



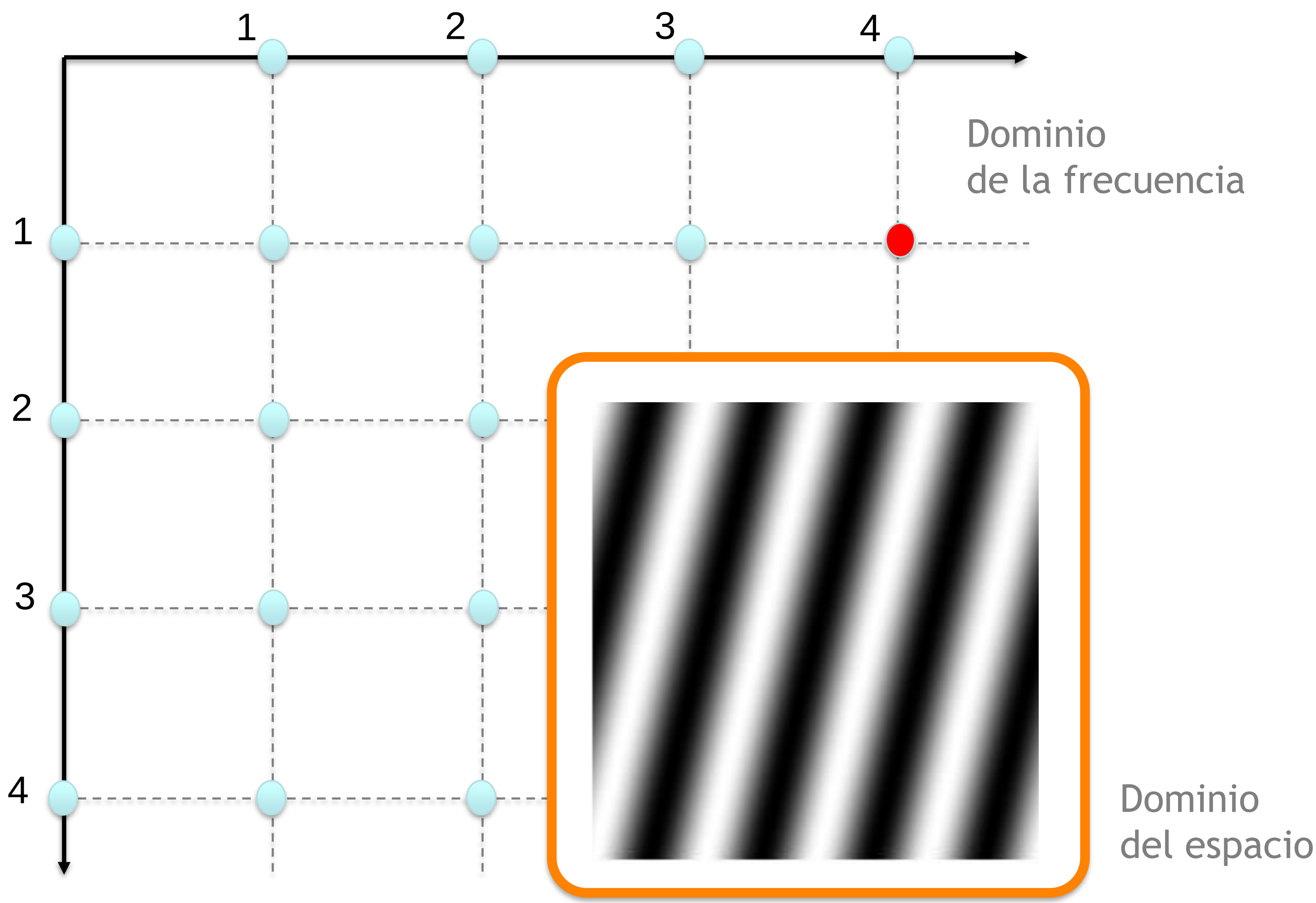
Espacio de dos frecuencias



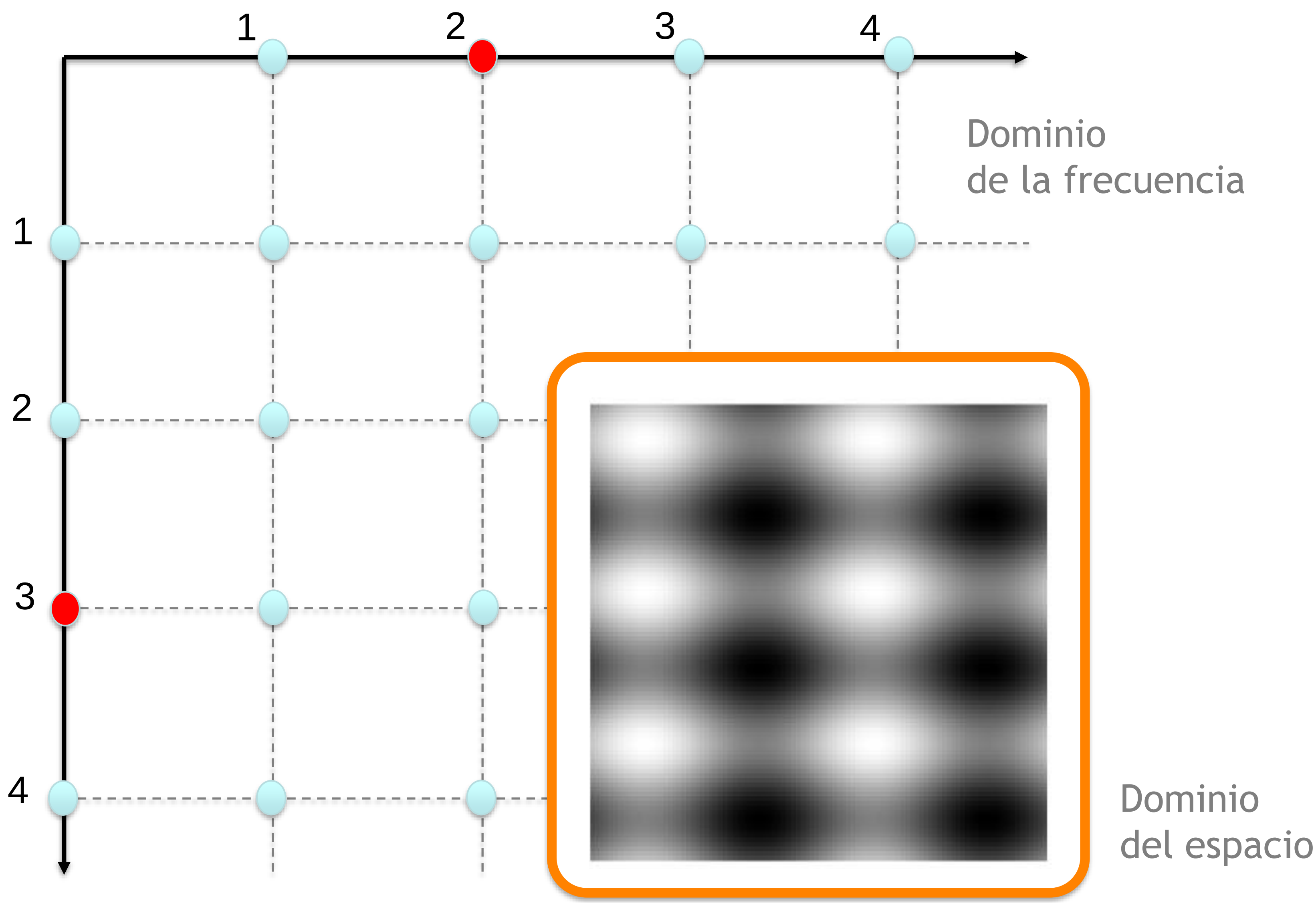
Espacio de dos frecuencias



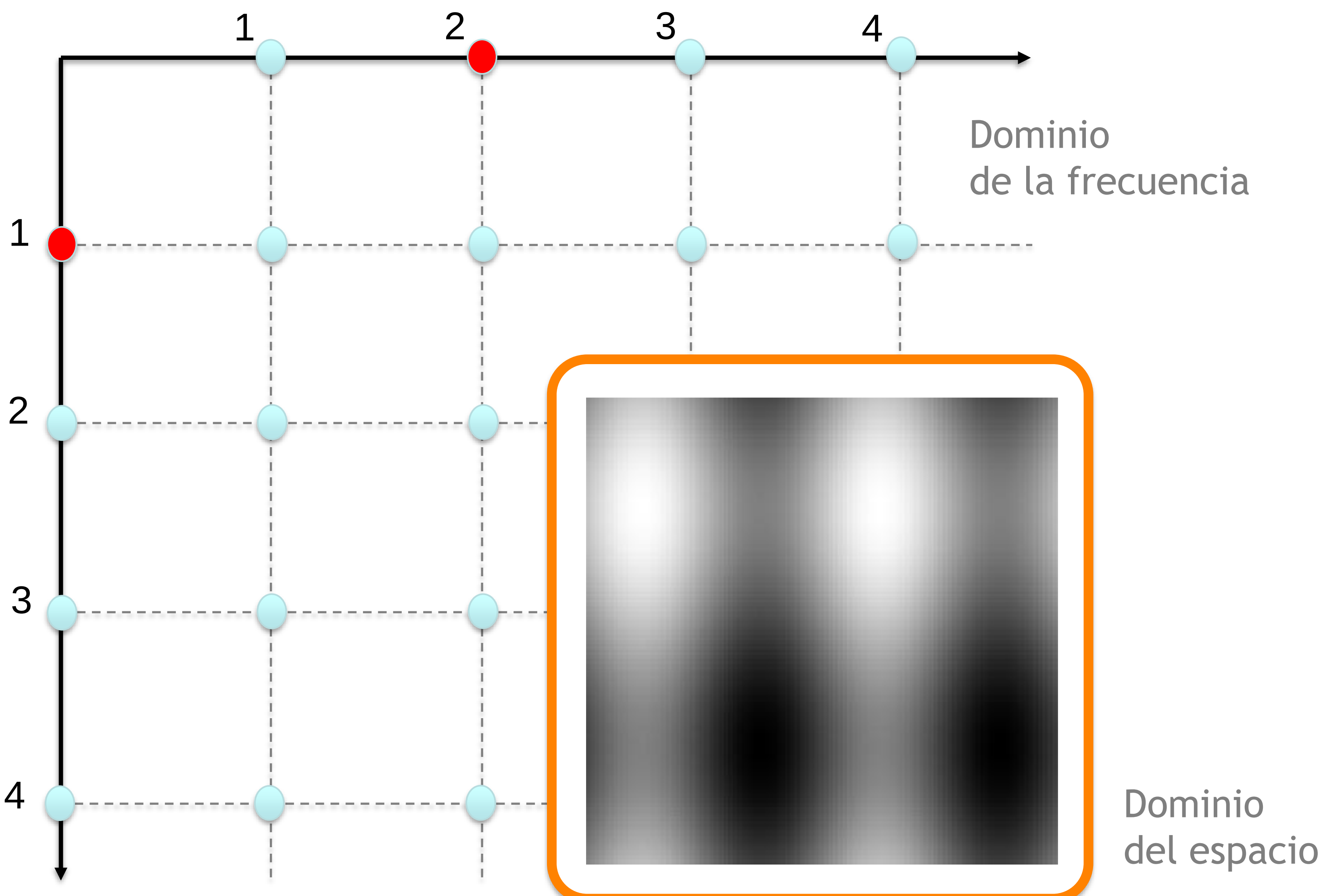
Espacio de dos frecuencias



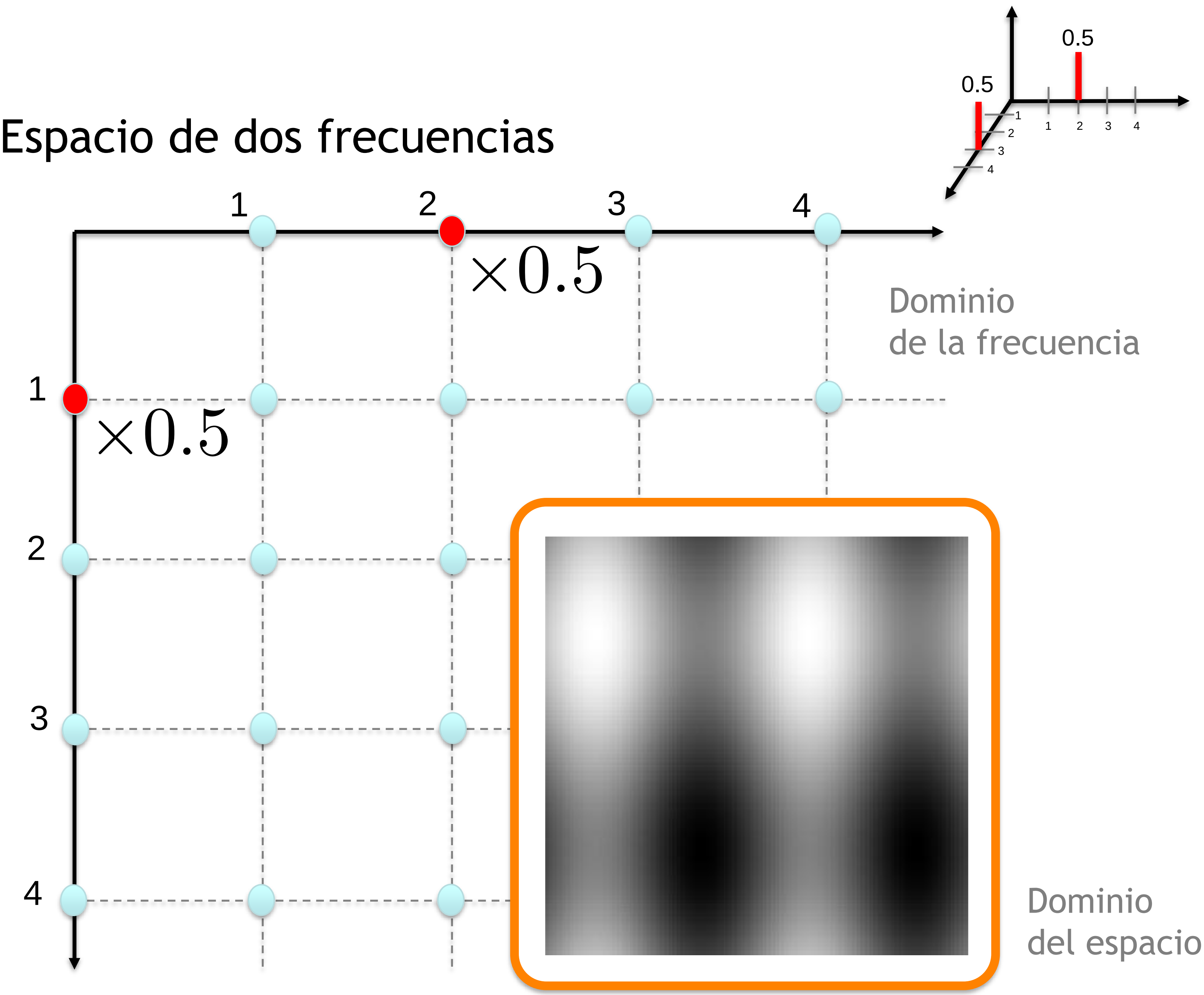
Espacio de dos frecuencias



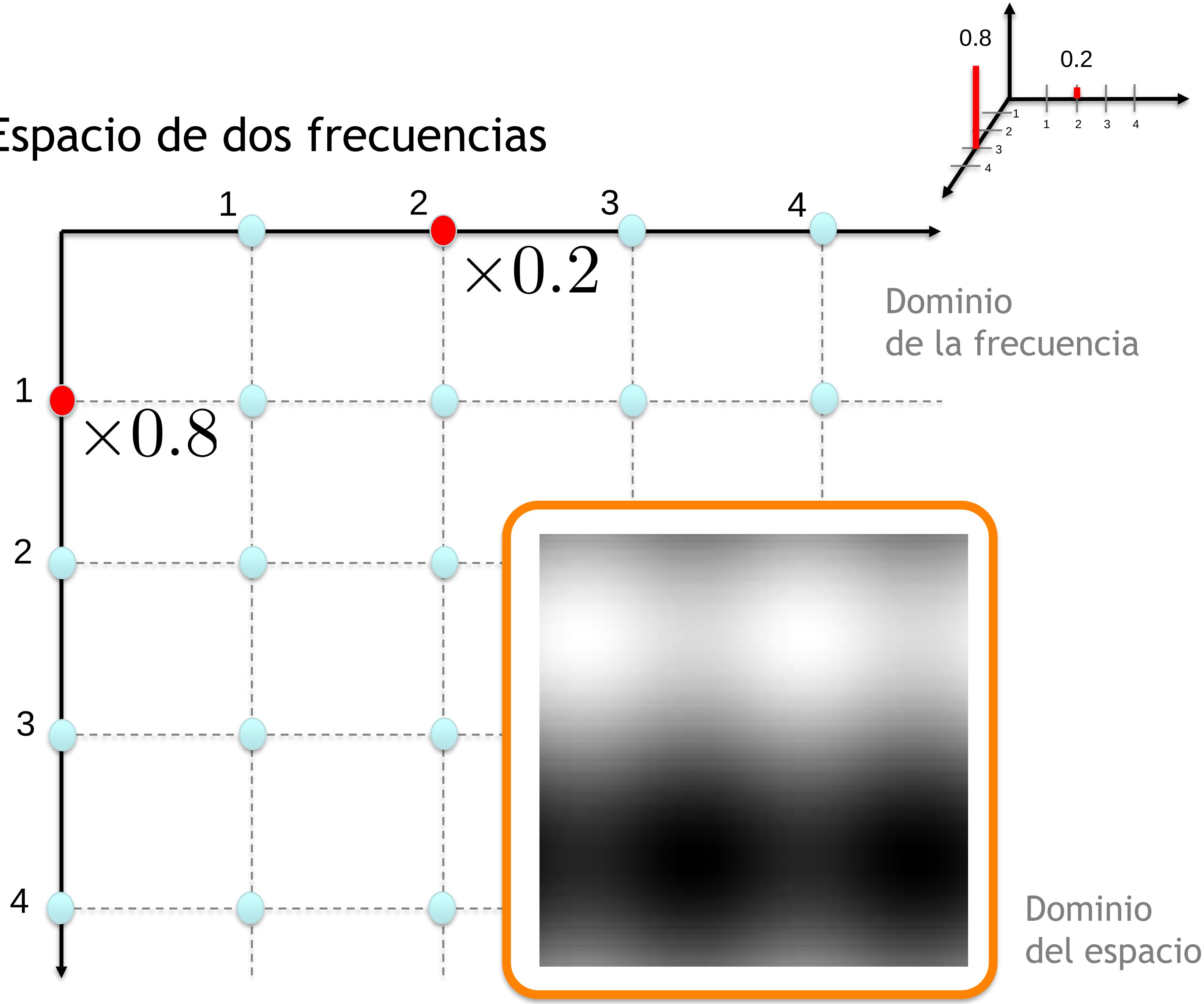
Espacio de dos frecuencias



Espacio de dos frecuencias



Espacio de dos frecuencias



Transformada de Fourier en 2D

Un sinusoide discreto en 2D se define como

$$f(m,n) = \sin\left[2\pi\left(Um + Vn\right)\right]$$

donde

U: frecuencia vertical (por filas) $0 < U < 1/2$

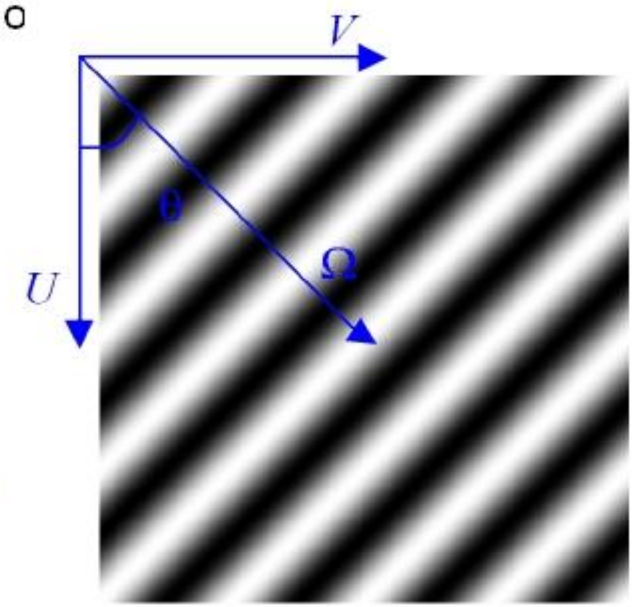
V: frecuencia horizontal (por columnas) $0 < V < 1/2$

La frecuencia más alta de oscilación del sinusoide se conoce como **frecuencia radial** y está definida como

$$\Omega = \sqrt{U^2 + V^2}$$

que ocurre en la **dirección**

$$\theta = \tan^{-1}\left(\frac{V}{U}\right)$$



El sinusoide oscila en todas las direcciones excepto en la dirección perpendicular a θ!

Un exponencial complejo discreto en 2D puede escribirse como

$$e^{j2\pi(Um+Vn)} = \cos\left[2\pi(Um+Vn)\right] + j\sin\left[2\pi(Um+Vn)\right]$$

Para el caso de una imagen finita de tamaño $M \times N$ se definen las frecuencias normalizadas, dadas en ciclos/imagen

$$u = \underbrace{M}_{\#\,filas} \times \underbrace{U}_{Freq.Ver} \qquad 0 \le u \le M/2$$

$$v = \underbrace{N}_{\#\,col's} \times \underbrace{V}_{Freq.Hor} \qquad 0 \le v \le N/2$$

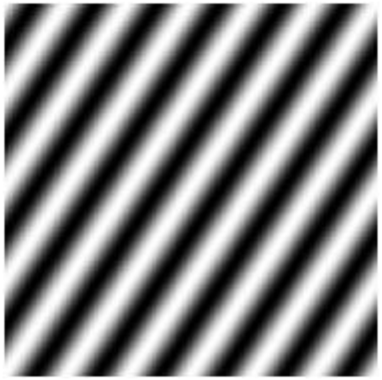
Por tal motivo, la función sinusoidal 2D puede escribirse como

$$f(m,n) = \sin\left[2\pi\left(\frac{u}{M}m + \frac{v}{N}n\right)\right] \qquad \begin{array}{l} 0 \le m \le M-1 \\ 0 \le n \le N-1 \end{array}$$

Ejemplo. Algunos sinusoides discretos ($M = N = 200$).

- $u = 4$ y $v = 6$

$$\theta = \mathbf{tan}^{-1}\left(\frac{6}{4}\right) \approx 56^\circ$$

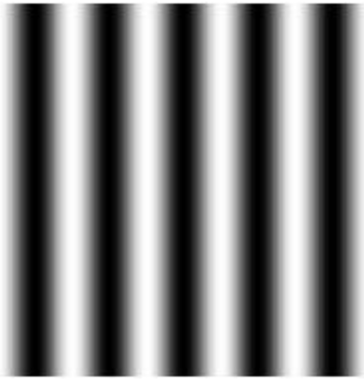


- $u = 20$ y $v = 2$

$$\theta = \mathbf{tan}^{-1}\left(\frac{2}{20}\right) \approx 5^\circ$$

- $u = 0$ y $v = 5$

$$\theta = \mathbf{tan}^{-1}\left(\frac{5}{0}\right) \approx 90^\circ$$



La **transformada de Fourier** de tiempo discreto 2D se define como

$$\Im\{f(m,n)\}_{2D} = F(U,V) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} \underbrace{f(m,n)e^{-j2\pi(Um+Vn)}}_{comparación!} \qquad 0 \le U,V \le 1$$

$F(U,V)$ da **información de la similitud** de la imagen $f(m,n)$ con un exponencial complejo de frecuencia (U,V).

- $F(U,V)$ es periódica con periodo 1 en filas y en columnas, i.e

$$F(U+k_1, V+k_2) = F(U,V), \quad \forall k_1, k_2 \in \mathbf{Z}$$

- $F(0,0)$ da información del nivel DC de la imagen

$$F(0,0) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)$$

- $F(0.5,0.5)$ da información sobre la componente de frecuencia más alta de la imagen

$$F\left(\tfrac{1}{2},\tfrac{1}{2}\right) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)e^{-j(\pi m+\pi V n)}$$

Transformada de Fourier 2D

CONTINUA

$$F(\omega, \nu) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(t, z)e^{-j2\pi(\omega t + \nu z)}dt\, dz$$

Transformada Directa

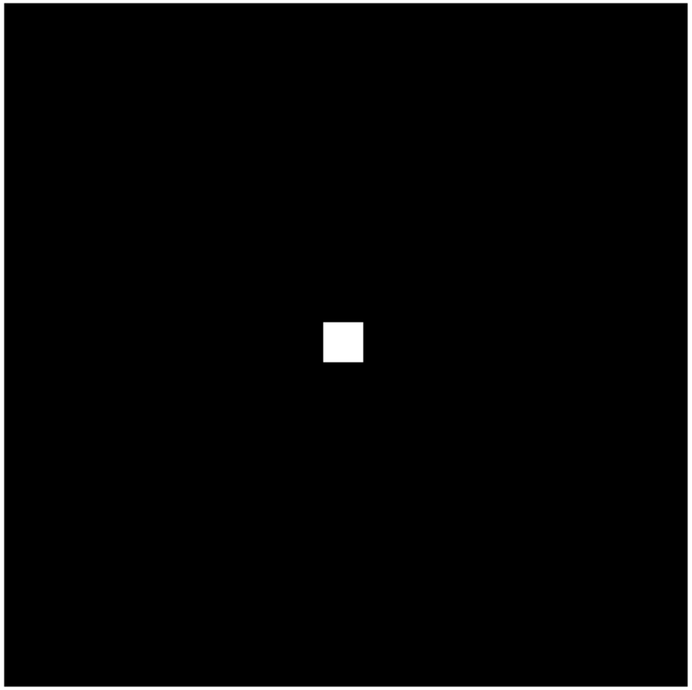
$$f(t, z) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(\omega, \nu)e^{j2\pi(\omega t + \nu z)}d\omega\, d\nu$$

Transformada Inversa

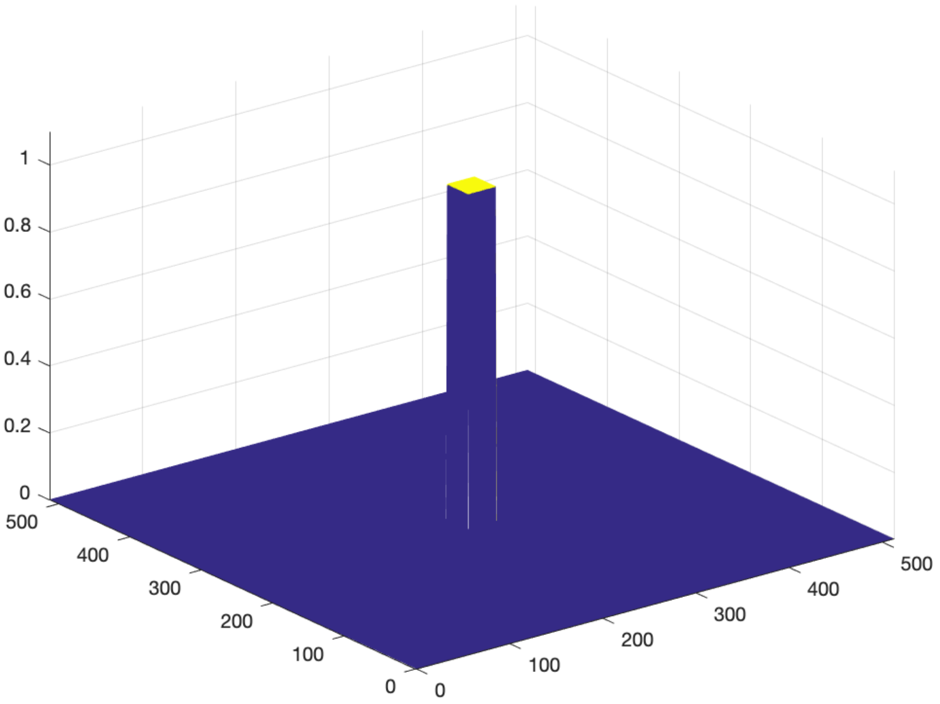
$$j = \sqrt{-1}$$

Transformada de Fourier 2D

EJEMPLO



(escala de grises)

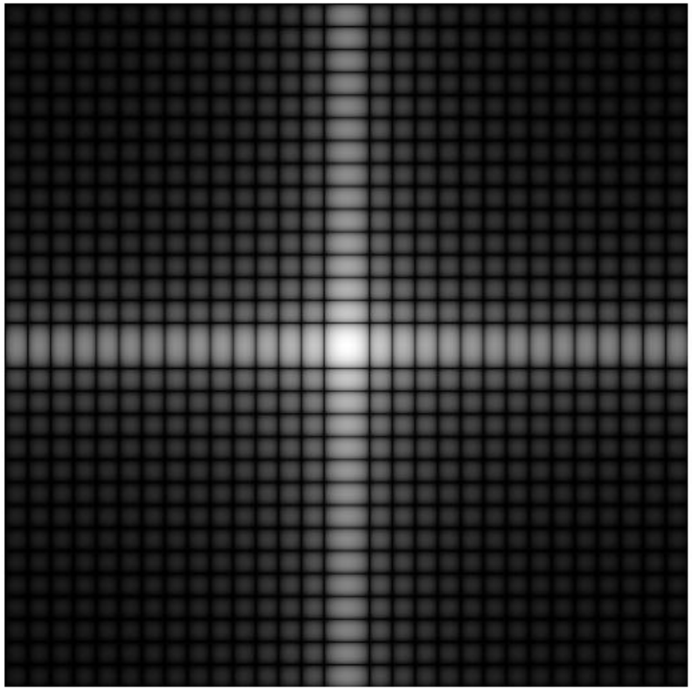


(representación 3D)

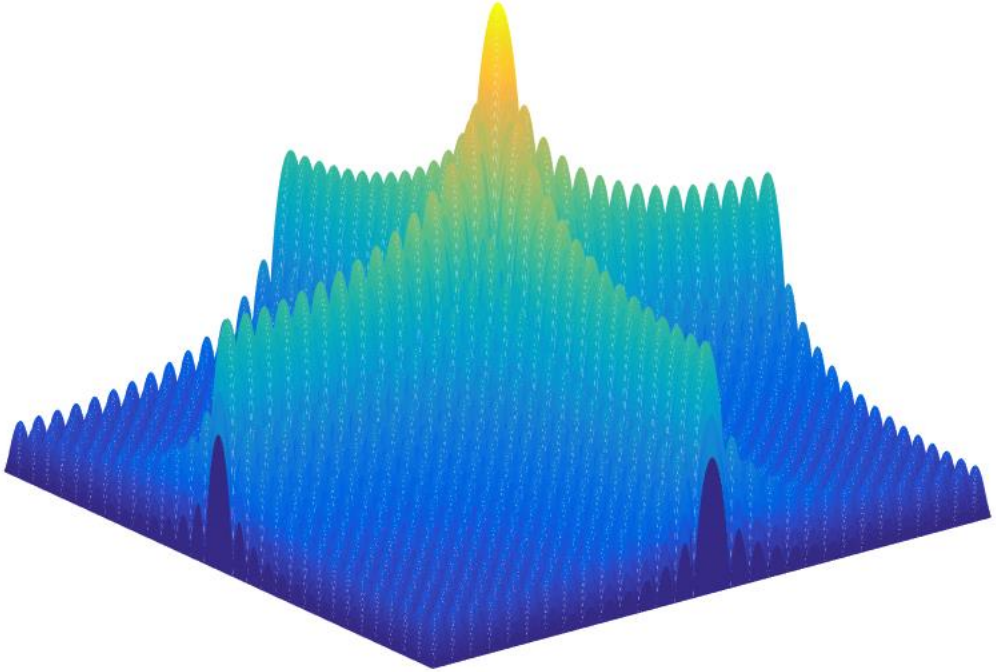
Dominio del Espacio

Transformada de Fourier 2D

EJEMPLO



(escala de grises)



(representación 3D)

Dominio de la Frecuencia

Transformada de Fourier 2D

DISCRETA

$$F(\omega,\nu) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(\omega x/M + \nu y/N)}$$

Transformada Directa

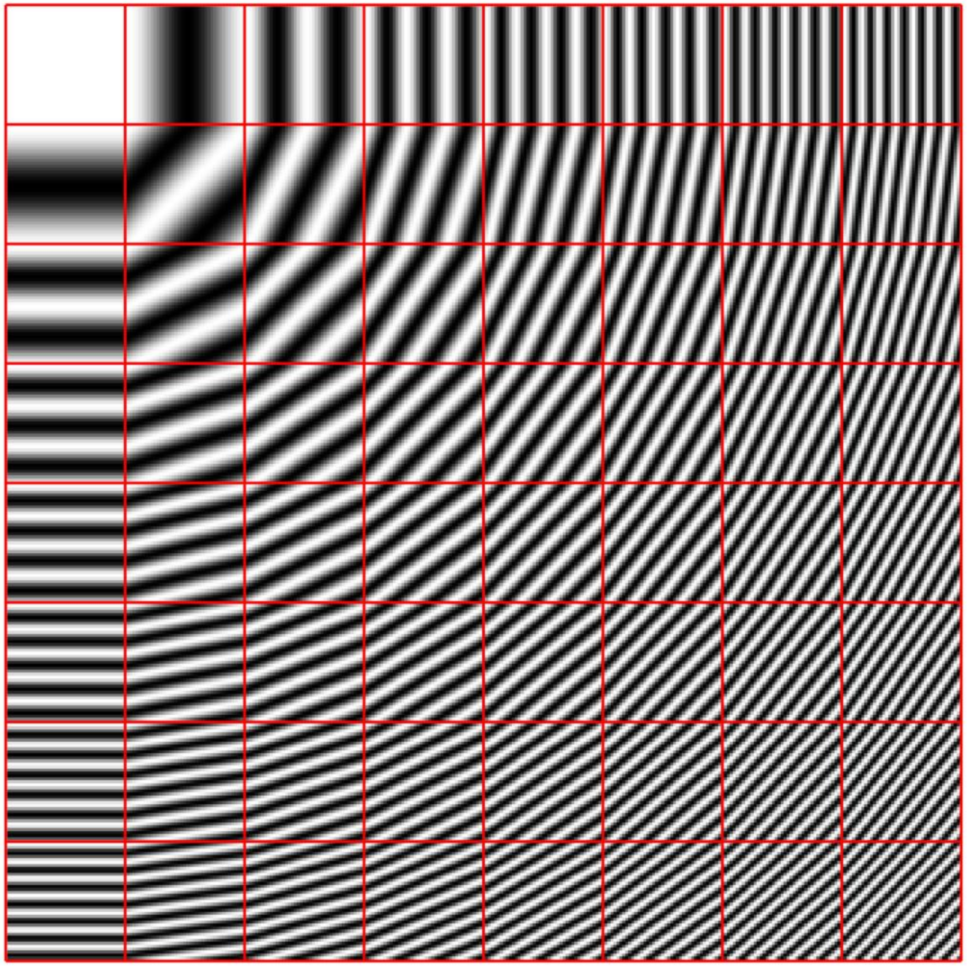
$$f(x,y) = \frac{1}{MN} \sum_{\omega=0}^{M-1} \sum_{\nu=0}^{N-1} F(\omega,\nu) \overbrace{e^{j2\pi(\omega x/M + \nu y/N)}}^{\text{Funciones Base}}$$

Transformada Inversa

Coeficientes

Transformada de Fourier 2D

Funciones Base



Transformada de Fourier 2D

DISCRETA

$$F(\omega,\nu) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(\omega x/M+\nu y/N)}$$

Transformada Directa

$$f(x,y) = \frac{1}{MN}\sum_{\omega=0}^{M-1}\sum_{\nu=0}^{N-1} \underbrace{F(\omega,\nu)}_{\text{Coeficientes}}\overbrace{e^{j2\pi(\omega x/M+\nu y/N)}}^{\text{Funciones Base}}$$

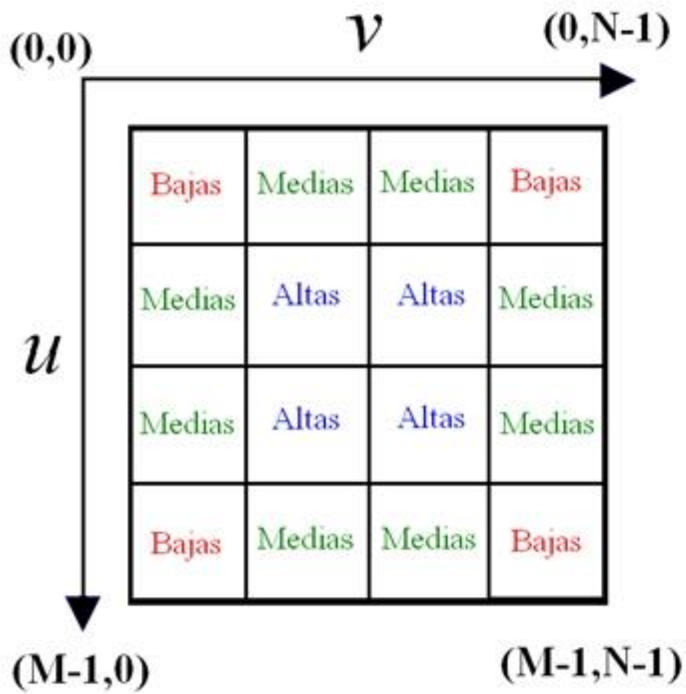
Transformada Inversa

Visualización de la DFT en 2D

La información contenida en la DFT se encuentra sobre un rango muy amplio de valores, por lo que se visualiza haciendo

$$\log\left[1+\left|F(u,v)\right|\right]$$

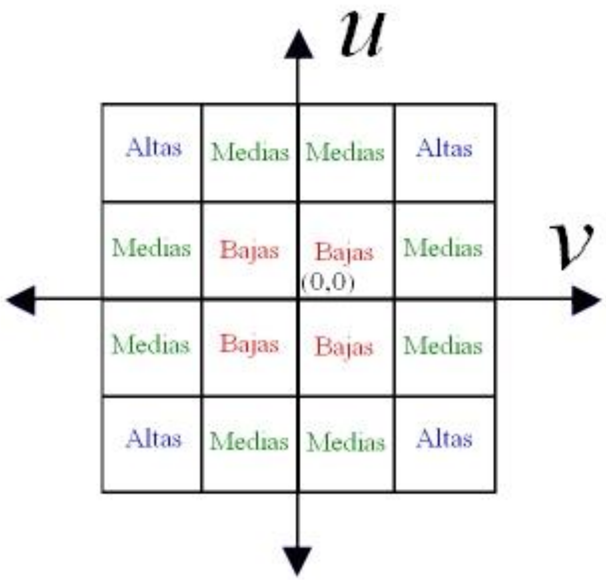
Al graficar se tiene la información de las frecuencias así



$$U=\frac{u}{M}$$

$$V=\frac{v}{N}$$

Intercambiando los cuadrantes 1 y 4, y 2 y 3 se obtiene una información más intuitiva y fácil de interpretar

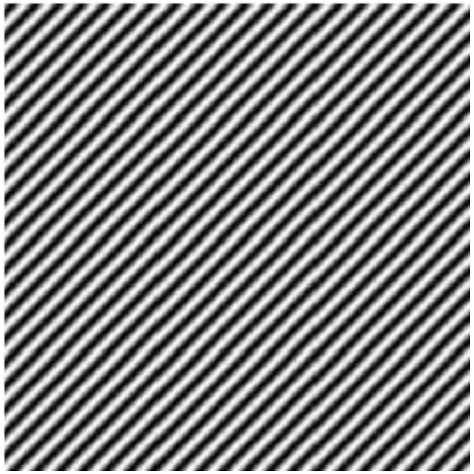
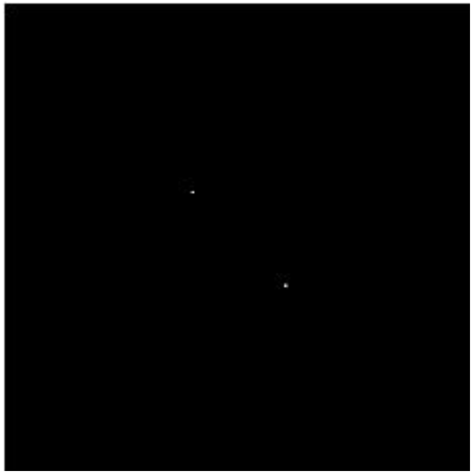


Ejemplo. En Python

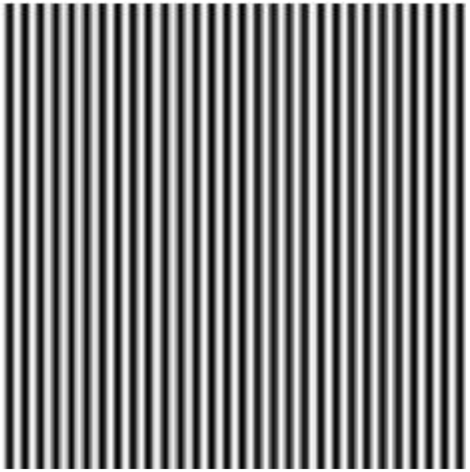
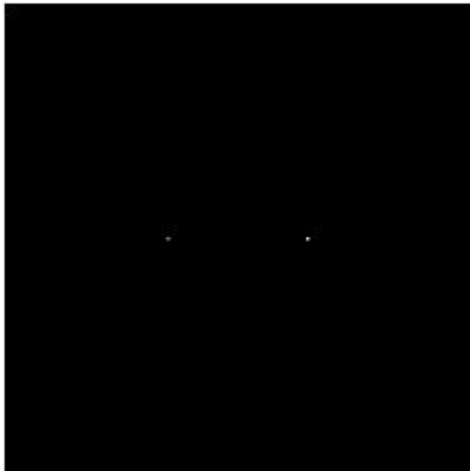
```
def calculate_2dft(input):
    ft = np.fft.ifftshift(input)
    ft = np.fft.fft2(ft)
    return np.fft.fftshift(ft)

def calculate_2dift(input):
    ift = np.fft.ifftshift(input)
    ift = np.fft.ifft2(ift)
    ift = np.fft.fftshift(ift)
    return ift.real
```

Ejemplo. Sinusoides en el tiempo y en la frecuencia

- $u = 20$ y $v = 20$

- $u = 0$ y $v = 30$

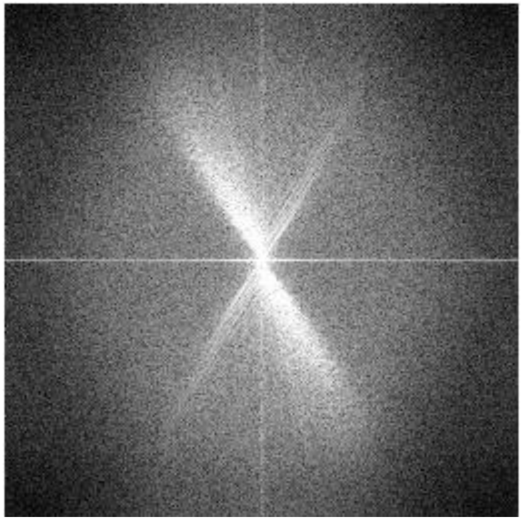
Se puede observar la **orientación** de la frecuencia radial!

Interpretación de la DFT

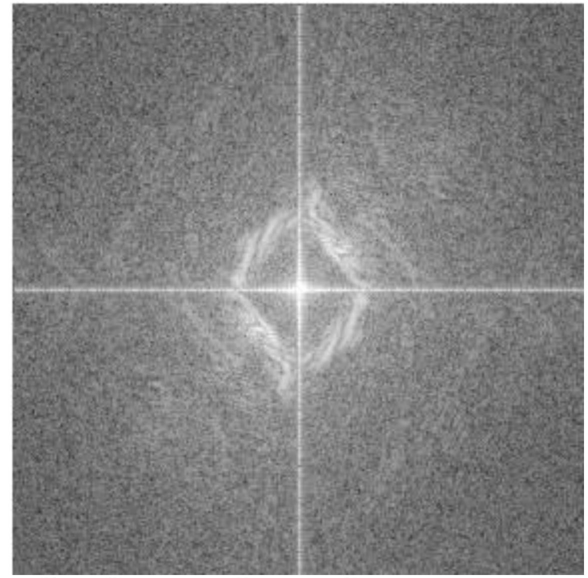
Cada punto de la DFT da 2 tipos de información:

- **Frecuencia radial:** la distancia de los puntos al centro indica la magnitud de las frecuencias presentes en la imagen.
- **Orientación:** el ángulo formado por los puntos con respecto al eje vertical indica la orientación de la frecuencia.

Ejemplo. Algunas imágenes y su correspondiente DFT

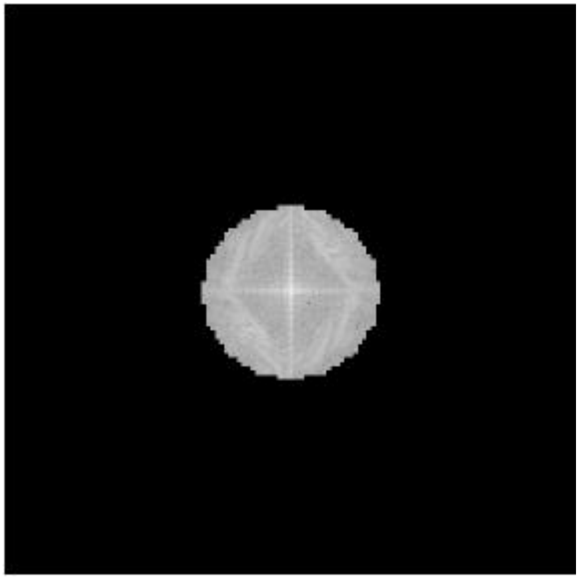


Frecuencias bajas y medias sobre una orientación especifica.

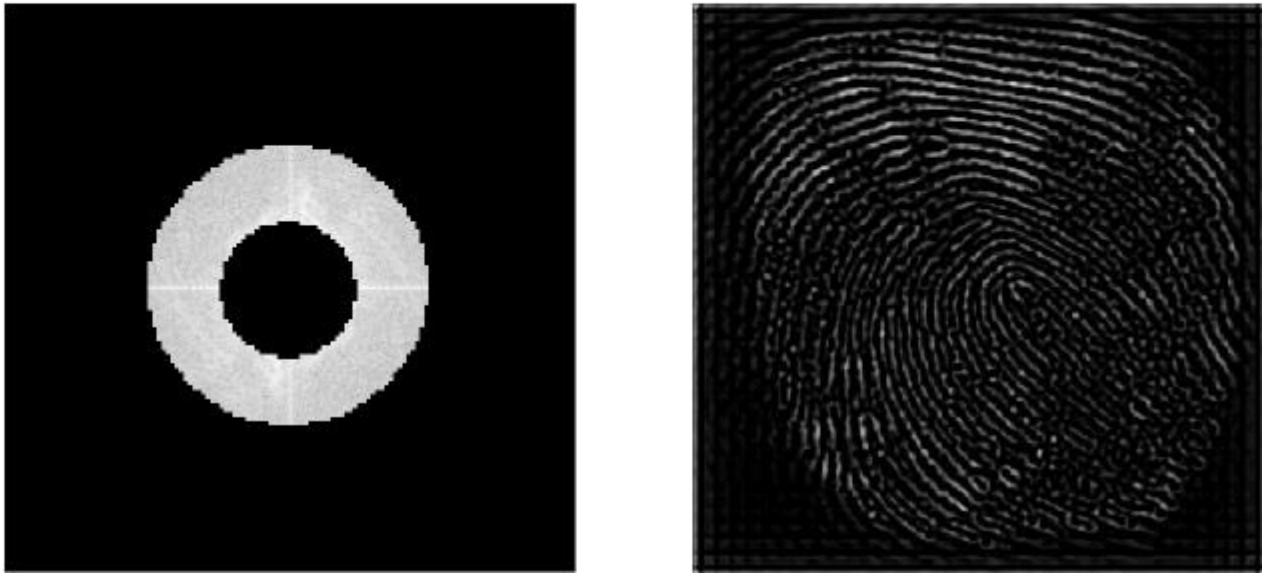


Frecuencias medias sobre todas las orientaciones.

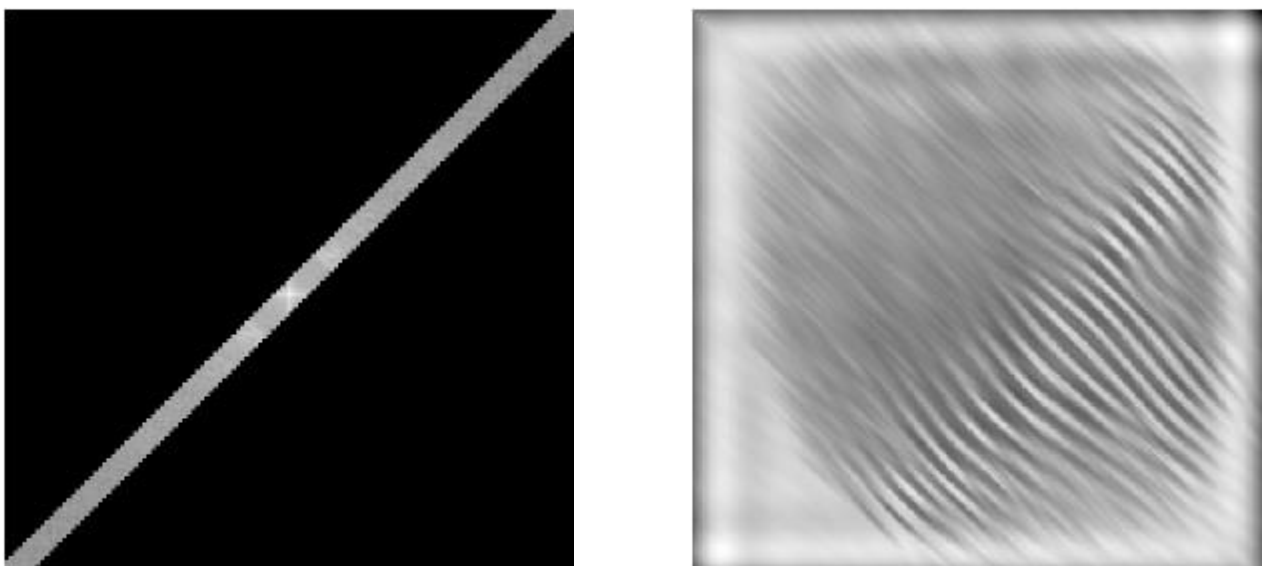
Si se eliminan las frecuencias altas de la imagen



Si se dejan sólo las componentes en una banda de frecuencia



Si se dejan sólo las frecuencias orientadas en 135°



Filtrado lineal 2D

Un sistema bidimensional L es un proceso que transforma una imagen $f(m,n)$ en una imagen $g(m,n)$, de acuerdo

$$f(m,n) \longrightarrow \boxed{L} \longrightarrow g(m,n)$$

Si el sistema L es lineal e invariante con el desplazamiento, i.e.,

- $L\{ax(m,n)+by(m,n)\} = aL\{x(m,n)\}+bL\{y(m,n)\}$

- Si $L\{x(m,n)\} = y(m,n)$ entonces
  $$L\{x(m-m_0,n-n_0)\} = y(m-m_0,n-n_0)$$

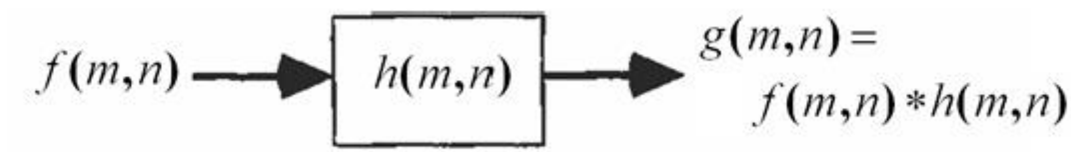
entonces se conoce como un sistema **LSI 2D** y se caracteriza completamente por su **respuesta impulso** $h(m,n)$, dada por

$$\delta(m,n) \longrightarrow \boxed{L} \longrightarrow h(m,n)$$

donde $\delta(m,n)$ es la función impulso bidimensional, definida como

$$\delta(m,n)=\begin{cases}1 & m=n=0\\ 0 & -\end{cases}$$

La salida del sistema a una imagen de entrada es encontrada como



La convolución entre 2 imágenes viene dada como

$$g(m,n)=f(m,n)*h(m,n)=\sum_{p=-\infty}^{\infty}\sum_{q=-\infty}^{\infty}f(p,q)h(m-p,n-q)$$

La respuesta impulso $h(m,n)$ es comúnmente es conocida como **máscara** o ***kernel* de convolución**.

Ejemplo. La imagen $f(m,n)$ es aplicada al sistema con respuesta impulso (máscara) $h(m,n)$.

$f(m,n)$

10	11	10	0	0	1
9	10	11	1	0	1
10	9	10	0	2	1
11	10	9	10	9	11
9	10	11	9	99	11
10	9	9	11	10	10

$g(m,n) = f(m,n) * h(m,n)$

X	X	X	X	X	X
X	10				X
X					X
X					X
X					X
X	X	X	X	X	X

$h(m,n)$

1/9

1	1	1
1	1	1
1	1	1

1/9.(10x1 + 11x1 + 10x1 + 9x1 + 10x1 + 11x1 + 10x1 + 9x1 + 10x1) = 1/9.(90) = 10

Propiedad de convolución lineal

La convolución entre 2 imágenes se define como

$$g(m,n) = f(m,n) * h(m,n) = \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} f(p,q)h(m-p,n-q)$$

Se puede demostrar que

$$\Im\{f(m,n) * h(m,n)\}_{2D} = \Im\{f(m,n)\}_{2D}\,\Im\{h(m,n)\}_{2D}$$

o lo que es equivalente

$$G(U,V) = F(U,V)H(U,V)$$

donde $F(U,V) = \Im\{f(m,n)\}_{2D}$ y $H(U,V) = \Im\{h(m,n)\}_{2D}$.

Por tal motivo,

$$g(m,n) = f(m,n) * h(m,n) = \Im^{-1}\{F(U,V)H(U,V)\}_{2D}$$

Ejemplo. La respuesta de un sistema a un exponencial complejo de frecuencia (U_0,V_0) está dada como

$$L\left\{e^{j2\pi(U_0m+V_0n)}\right\}=e^{j2\pi(U_0m+V_0n)}*h(m,n)=\sum_{p=-\infty}^{\infty}\sum_{q=-\infty}^{\infty}e^{j2\pi[U_0(m-p)+V_0(n-q)]}h(p,q)$$

$$=e^{j2\pi(U_0m+V_0n)}\sum_{p=-\infty}^{\infty}\sum_{q=-\infty}^{\infty}h(p,q)e^{-j2\pi(U_0m+V_0n)}=e^{j2\pi(U_0m+V_0n)}H(U_0,V_0)$$

Obs. $H(U_0,V_0)$ indica cómo responde el sistema a la componente de frecuencia (U_0,V_0), i.e., como responde el sistema a la

frecuencia radial $\Omega_0=\sqrt{U_0^2+V_0^2}$ en la **orientación** $\theta_0=\tan^{-1}\left(\dfrac{V_0}{U_0}\right)$

Respuesta en frecuencia

La transformada de Fourier en 2D de la imagen de salida puede encontrarse como

$$G(U,V)=F(U,V)H(U,V) \quad : \quad H(U,V)=\Im\{h(m,n)\}_{2D}$$

La función $H(U,V)$ se conoce como **respuesta en frecuencia** del sistema, pues indica como responde el sistema a la frecuencia (U,V).

Propiedad de convolución circular

Sean $f(m,n)$ y $h(m,n)$ imágenes de $M \times N$.

Observación1. La transformada de Fourier de $f(m,n)$ y $h(m,n)$ **no puede** ser encontrada utilizando un computador.

Observación2. La transformada **discreta** de Fourier de $f(m,n)$ y $h(m,n)$ **si puede** ser encontrada utilizando un computador.

Sean $f_p(m,n)$ y $h_p(m,n)$ extensiones periódicas de $f(m,n)$ y $h(m,n)$, respectivamente, de periodo M en filas y N en columnas, entonces

$$\mathbf{DFT}_{2D}\left\{f_p(m,n) \otimes h_p(m,n)\right\} = F(u,v)H(u,v)$$

donde $F(u,v) = \mathbf{DFT}_{2D}\left\{f(m,n)\right\}$ y $H(u,v) = \mathbf{DFT}_{2D}\left\{h(m,n)\right\}$.

Por tal motivo,

$$f_p(m,n) \otimes h_p(m,n) = \mathbf{DFT}_{2D}^{-1}\left\{F(u,v)H(u,v)\right\}$$

Convolución lineal vía DFT 2D

La convolución entre las imágenes $f(m,n)$ de $M_1 \times N_1$ y $h(m,n)$ de $M_2 \times N_2$ genera la imagen $g(m,n)$ de $(M_1 + M_2 - 1) \times (N_1 + N_2 - 1)$.

Idea. La convolución circular debe igualar a la convolución lineal.

Para realizar la convolución lineal vía DFT se sigue

1. Se rellenan con ceros las imágenes $f(m,n)$ y $h(m,n)$ hasta obtener imágenes de $(M_1 + M_2 - 1) \times (N_1 + N_2 - 1)$.

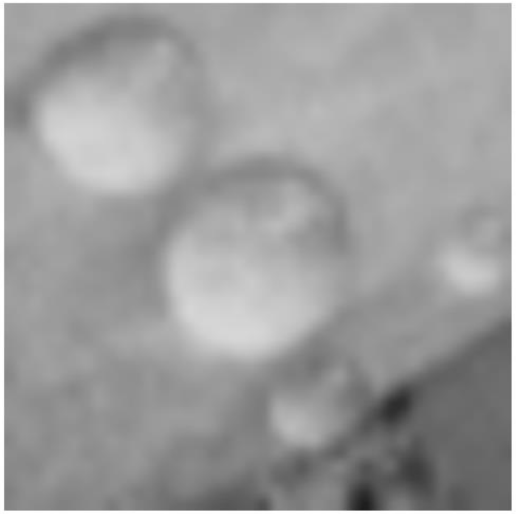
2. Se obtienen las DFT's $F(u,v)$ y $H(u,v)$.

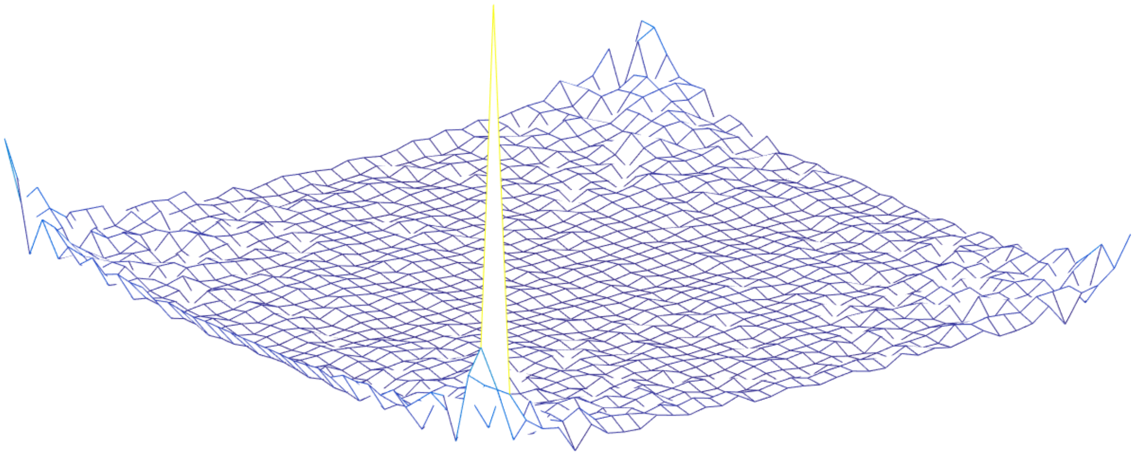
3. La convolución lineal es encontrada como

$$g(m,n) = \mathbf{DFT}_{2D}^{-1}\left\{ F(u,v)H(u,v) \right\}$$

Ejemplo

Imagen





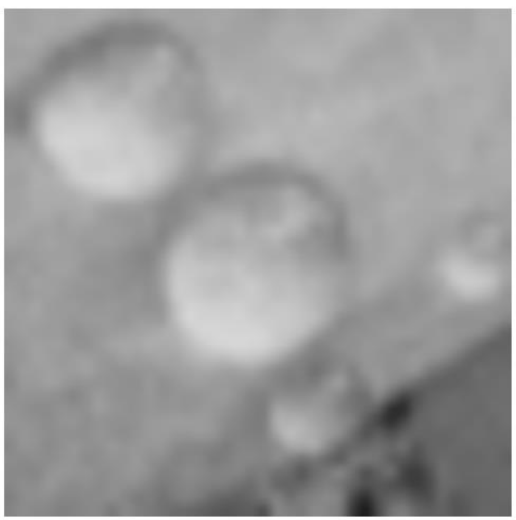
Transformada Discreta de Fourier
(original)

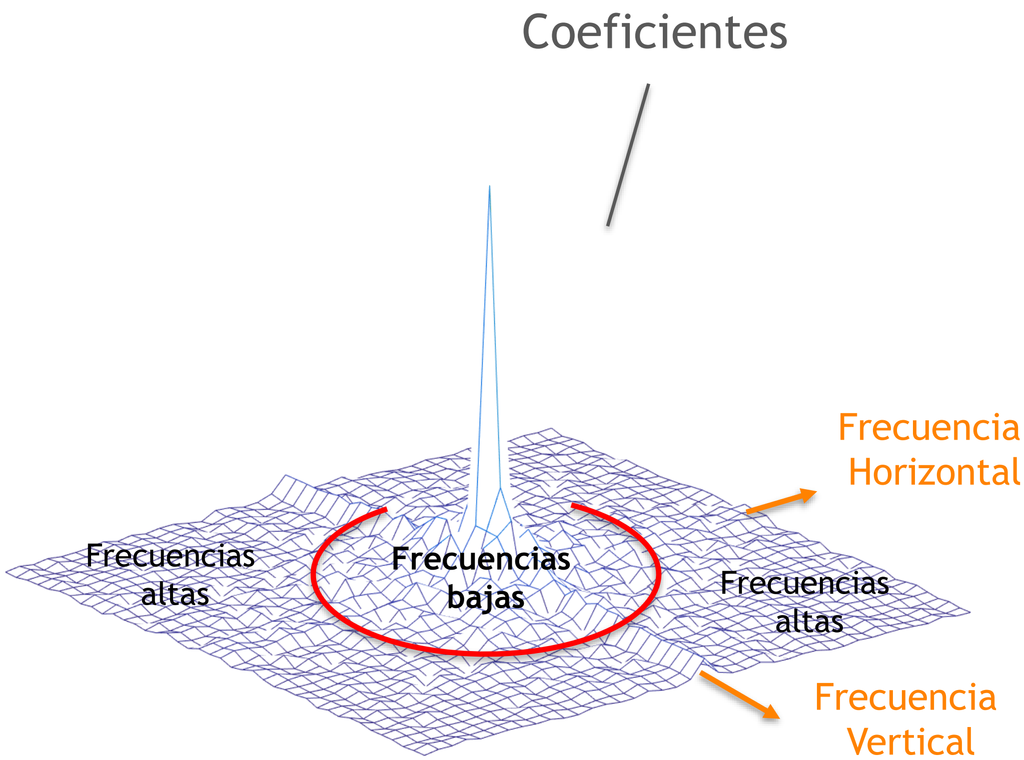
Ejemplo

¿Por qué es fftshiftnecesario?
¿Dónde está el centro de la imagen? Parece una pregunta tonta, ¿no? Sin embargo, cuando representas una imagen como una matriz en Python, el píxel con índices (0, 0) es el que está en la parte superior derecha. Este es el centro efectivo de la imagen desde un punto de vista matemático.

Ahora considere las imágenes de las transformadas de Fourier que ves abajo, donde el centro de la imagen tiene importancia y la distancia de los puntos desde este centro contiene información importante sobre la frecuencia de las rejillas sinusoidales.

Imagen

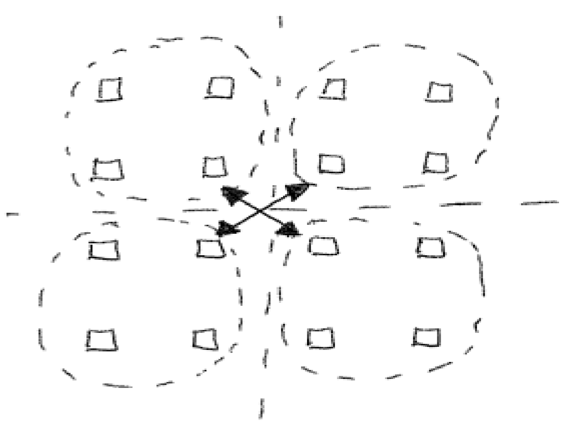




Transformada Discreta de Fourier (con FFTSHIFT)

```
>>> import numpy as np
>>> small_array = np.array([[1, 2, 3, 4],
...                         [5, 6, 7, 8],
...                         [9, 10, 11, 12],
...                         [13, 14, 15, 16]])
...
>>> small_array
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12],
       [13, 14, 15, 16]])

>>> np.fft.fftshift(small_array)
array([[11, 12,  9, 10],
       [15, 16, 13, 14],
       [ 3,  4,  1,  2],
       [ 7,  8,  5,  6]])
```

Los cuatro cuadrantes se intercambian de modo que el cuadrante superior izquierdo sea ahora el inferior derecho, y así sucesivamente. Esto se puede representar gráficamente con el siguiente diagrama:



¿Qué sucede con la matriz si tiene un número impar de filas y columnas? ¿En qué cuadrante se deben considerar la fila y la columna centrales? Ésta es la razón por la que existen dos funciones, un desplazamiento FFT y su inversa:

- >>> small_array = np.array([[1, 2, 3, 4, 5],
- ...   [6, 7, 8, 9, 10],
- ...   [11, 12, 13, 14, 15],
- ...   [16, 17, 18, 19, 20],
- ...   [21, 22, 23, 24, 25]])
- ...
- >>> small_array
- array([[1, 2, 3, 4, 5],
-       [6, 7, 8, 9, 10],
-       [11, 12, 13, 14, 15],
-       [16, 17, 18, 19, 20],
-       [21, 22, 23, 24, 25]])

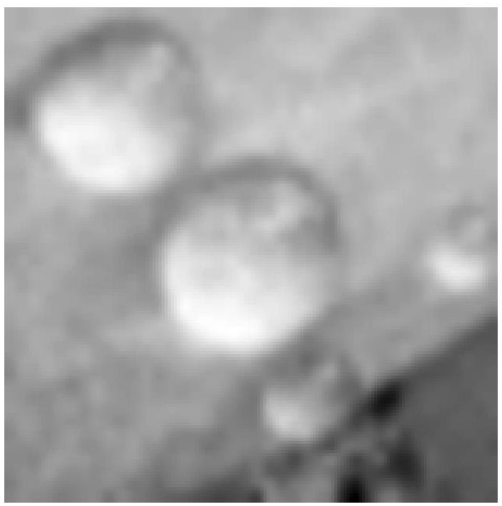
- >>> np.fft.fftshift(small_array)
- array([[19, 20, 16, 17, 18],
-       [24, 25, 21, 22, 23],
-       [4, 5, 1, 2, 3],
-       [9, 10, 6, 7, 8],
-       [14, 15, 11, 12, 13]])

- >>> np.fft.ifftshift(small_array) # Note the i in ifftshift
- array([[13, 14, 15, 11, 12],
-       [18, 19, 20, 16, 17],
-       [23, 24, 25, 21, 22],
-       [3, 4, 5, 1, 2],
-       [8, 9, 10, 6, 7]])

Mire el importante píxel central, que tiene el valor de 13 en la matriz original. Una de las funciones lo coloca al final de la matriz (abajo a la derecha), mientras que la otra lo coloca en la primera posición (arriba a la izquierda).

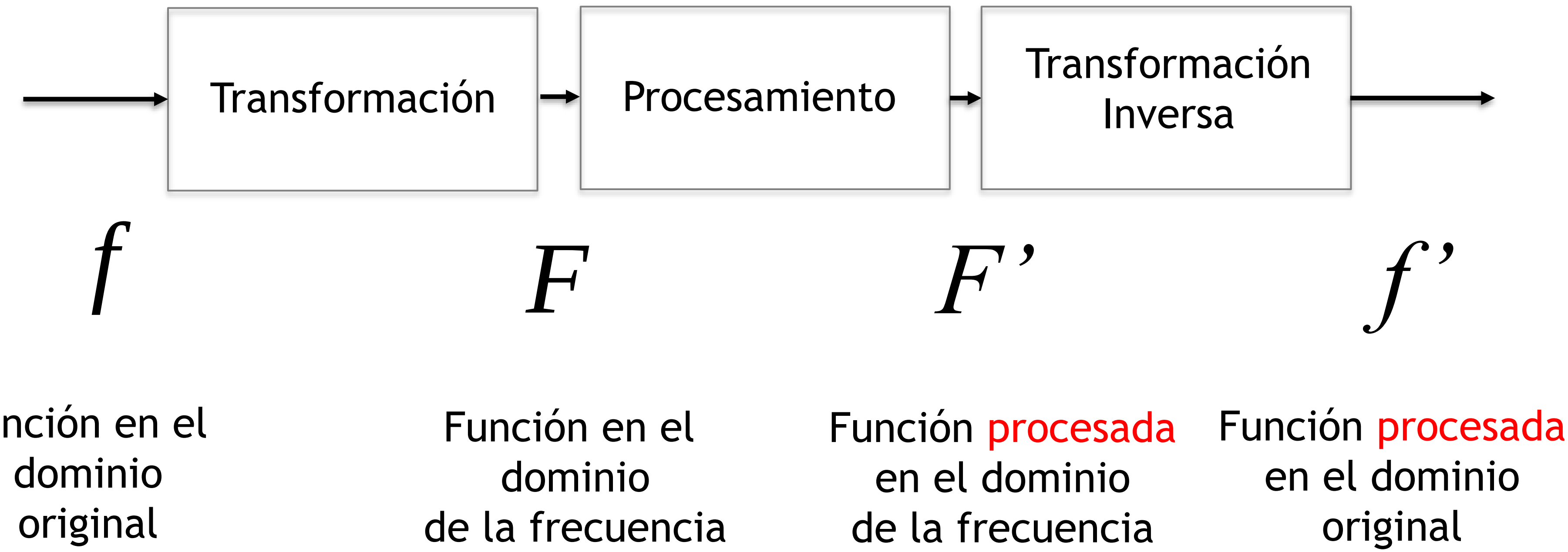
Ejemplo

Imagen



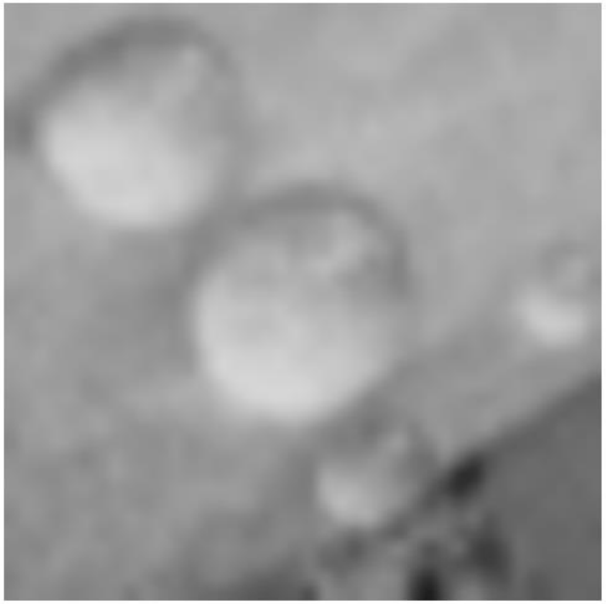
Reconstrucción

Procesamiento en el Dominio de la Frecuencia



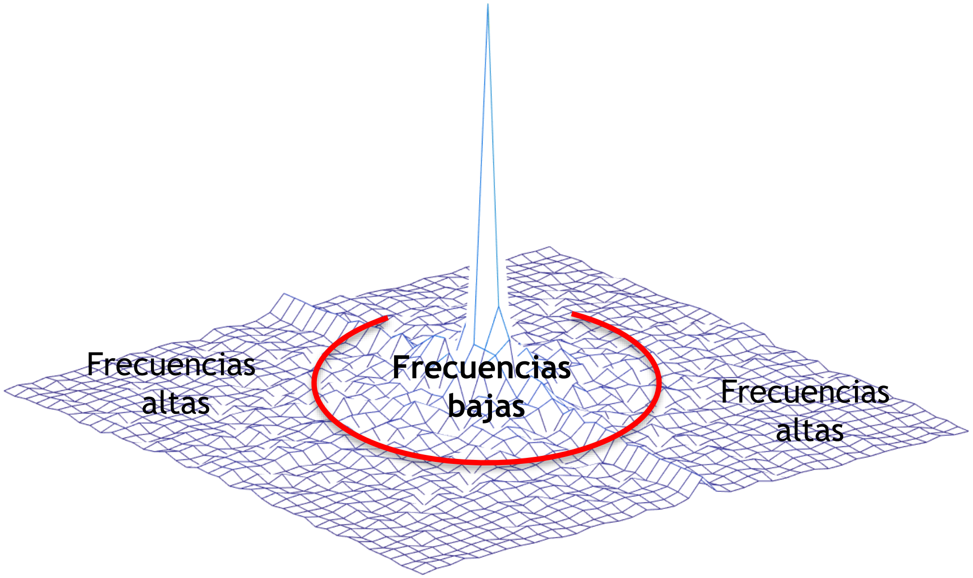
Ejemplo

f



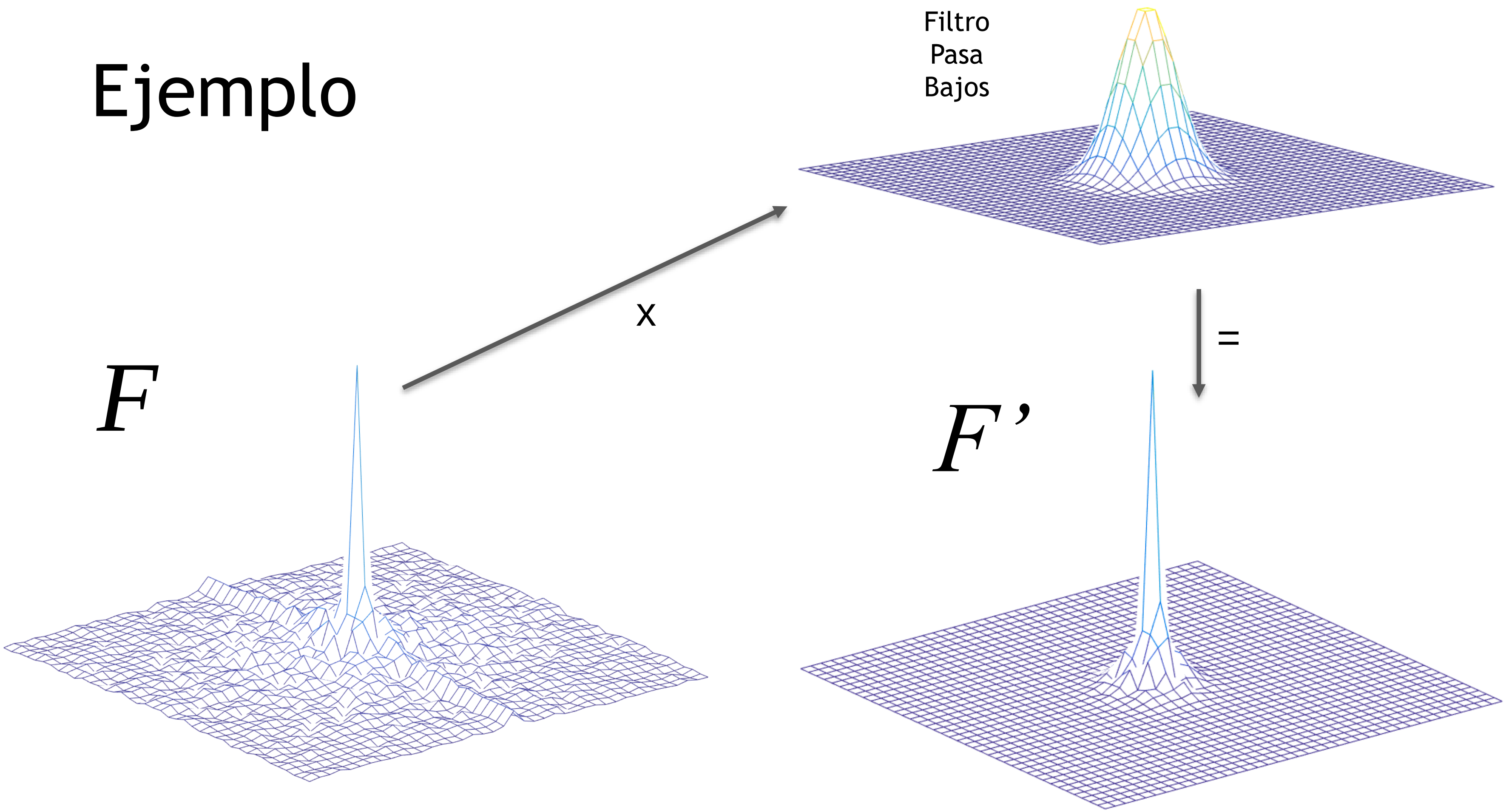
Función en el
dominio
original

F



Función en el
dominio
de la frecuencia

Ejemplo

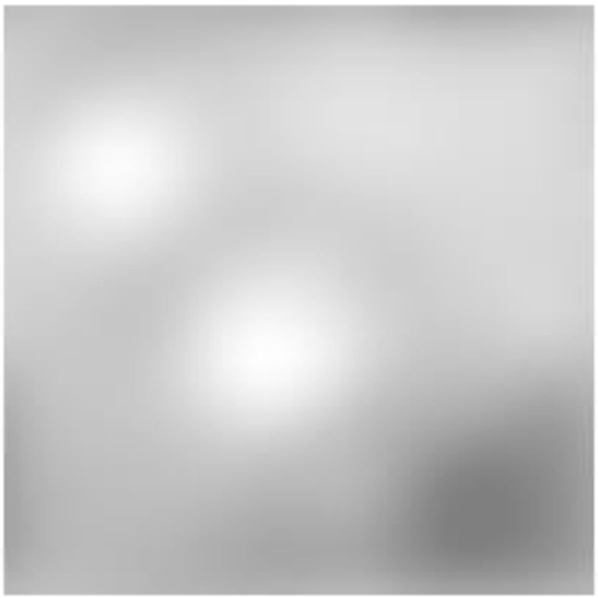


Función en el dominio de la frecuencia

Función filtrada en el dominio de la frecuencia

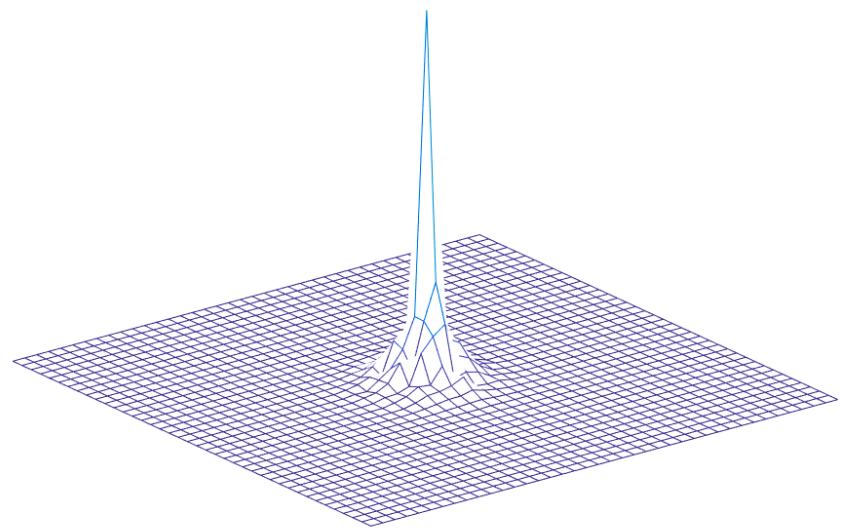
Ejemplo: Filtro Pasa Bajos

f'



Función filtrada
en el dominio
original

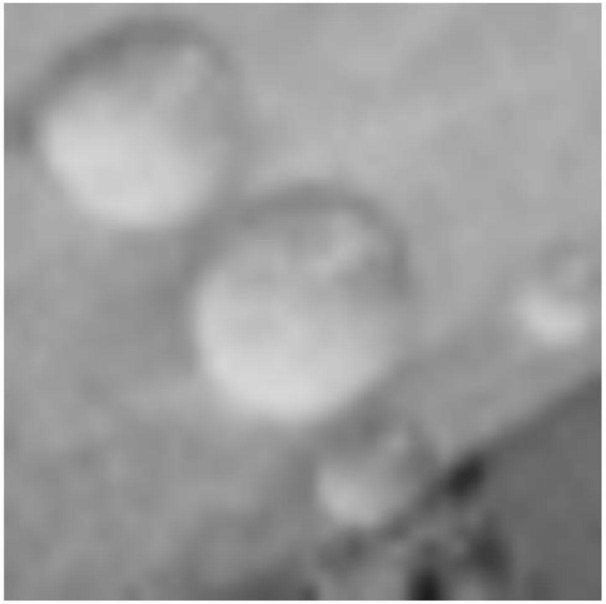
F'



Función filtrada
en el dominio
de la frecuencia

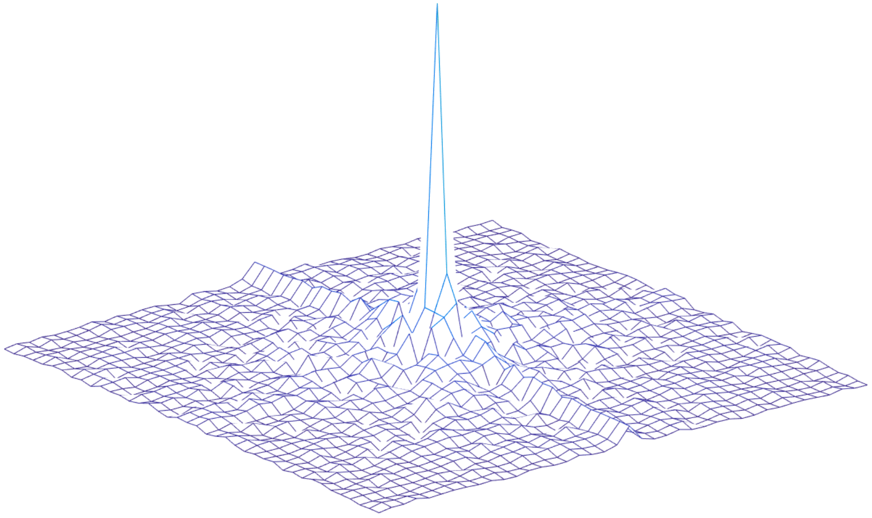
Ejemplo

f



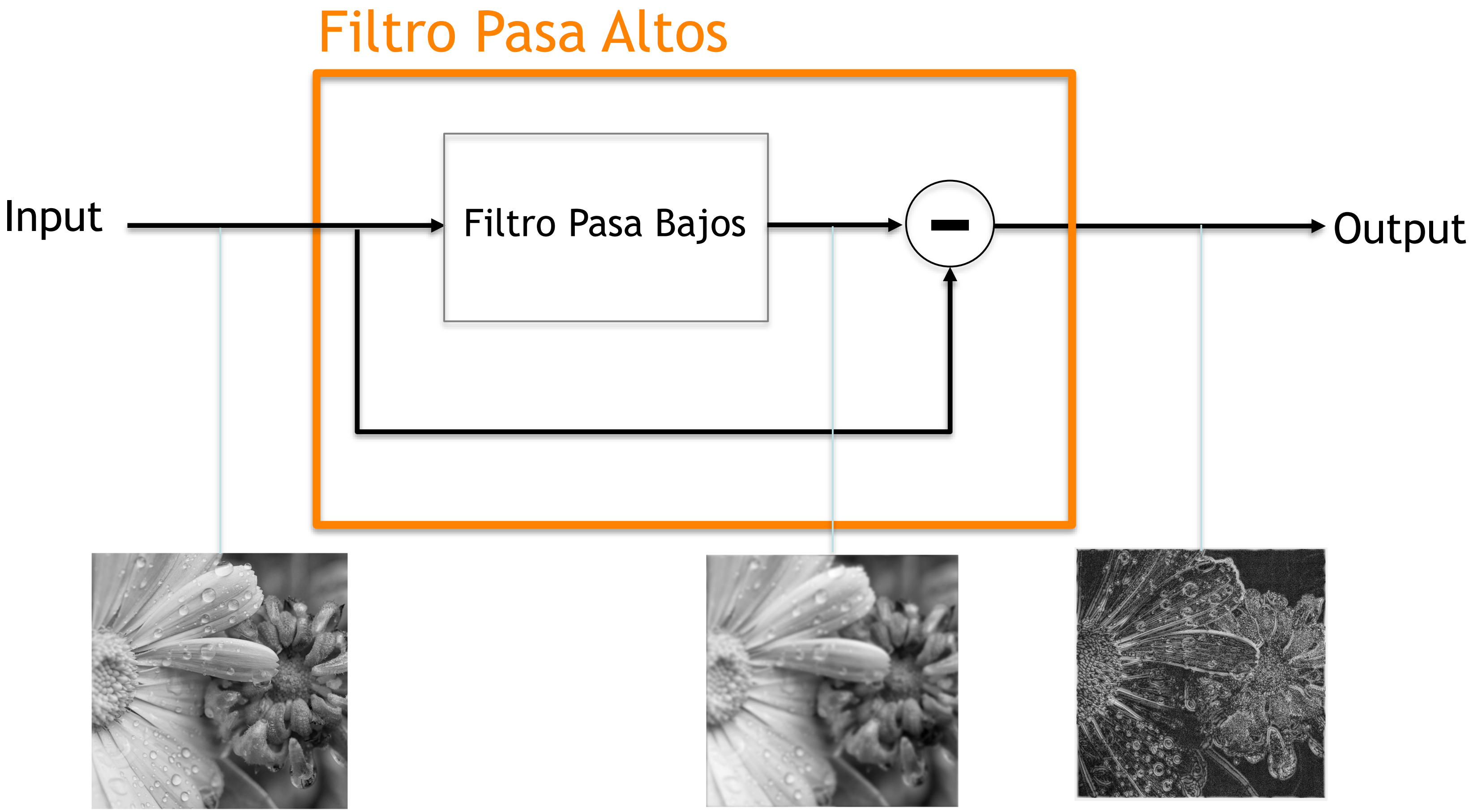
Función
en el dominio
original

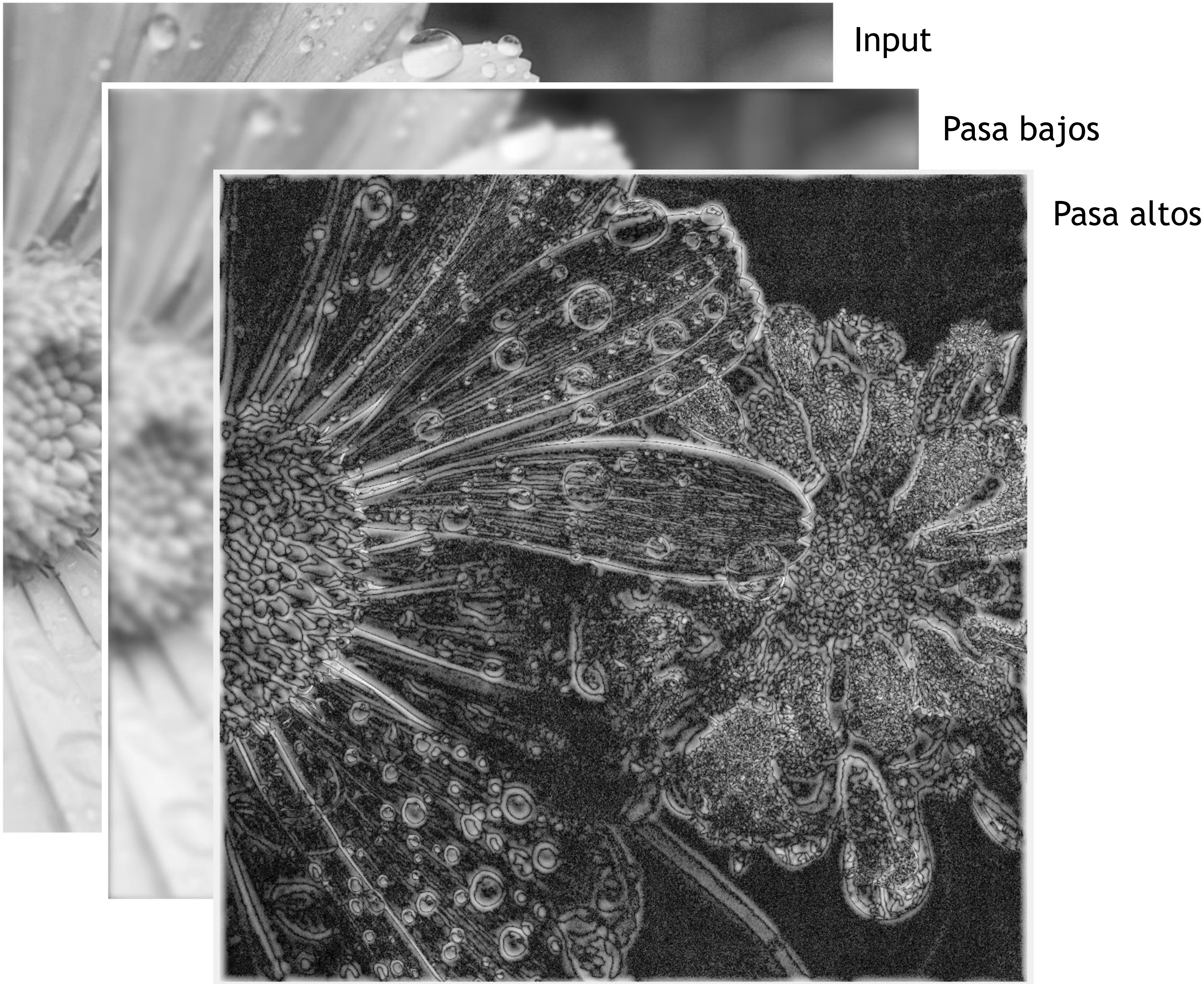
F



Función
en el dominio
de la frecuencia

Ejemplo: Filtro pasa altos




Input
Pasa bajos
Pasa altos