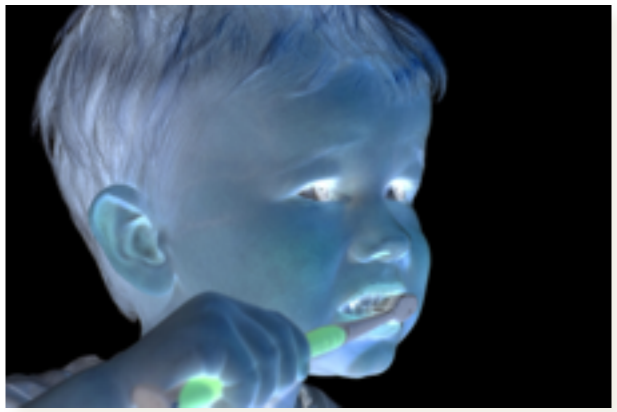


Image filtering

Frequency domain

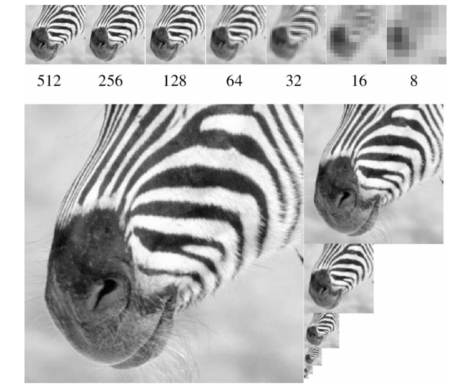


image pyramids

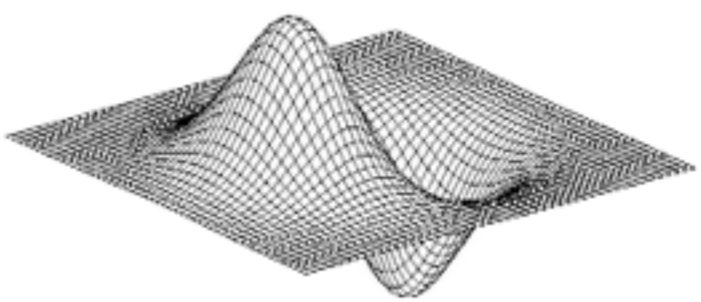
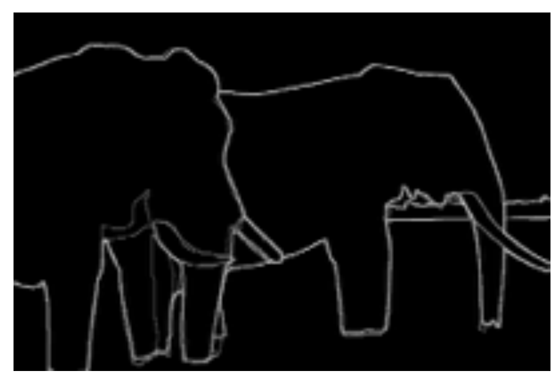
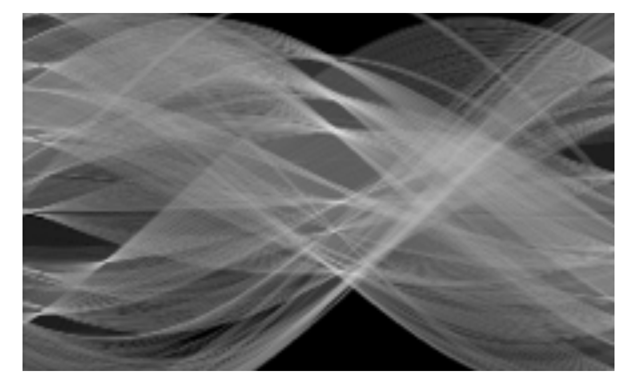


Image gradients



Boundaries



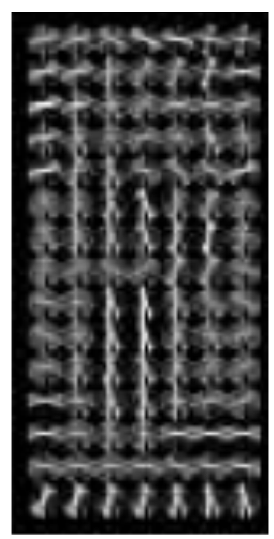
Hough Transform

Image Manipulation

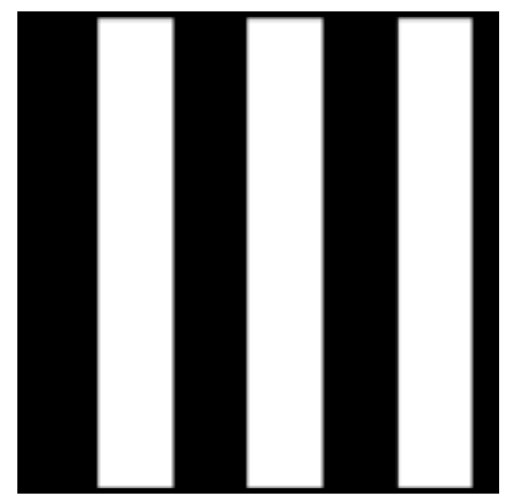
Corner detection

Multi-scale detection

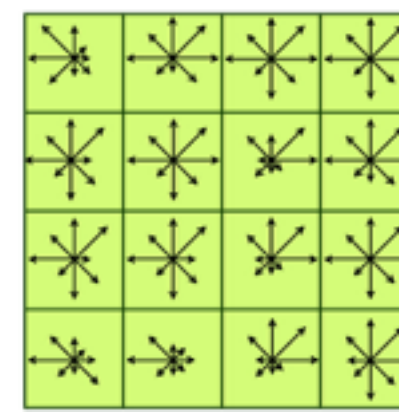
Haar-like



HOG

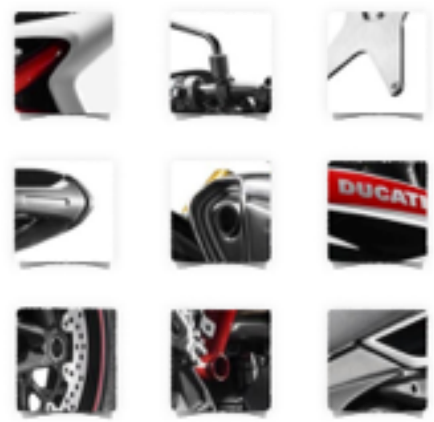


SURF

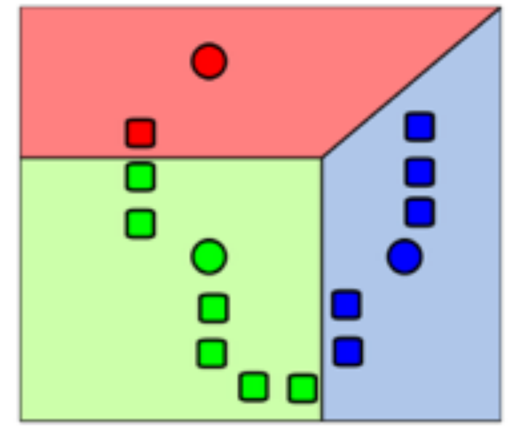


SIFT

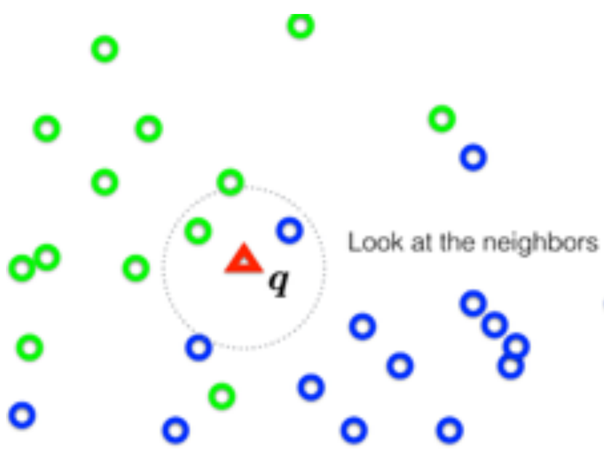
Image Features



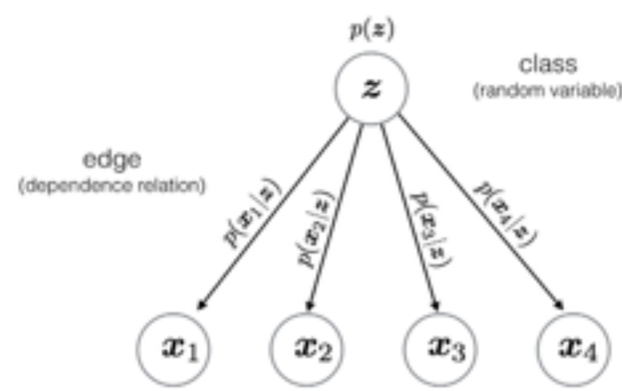
Bag-of-words



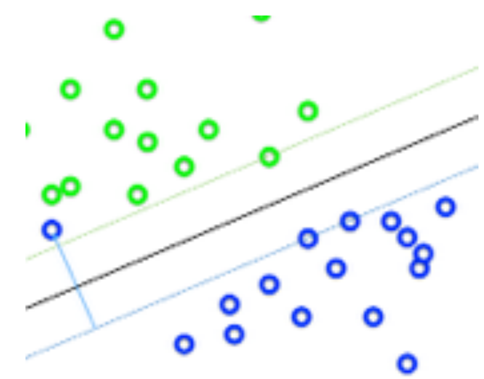
K-means



Nearest Neighbor

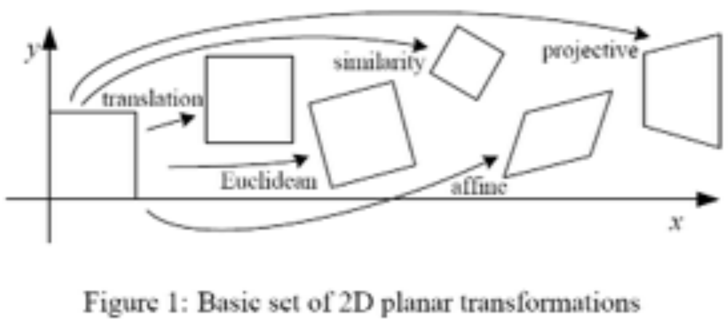


Naive Bayes

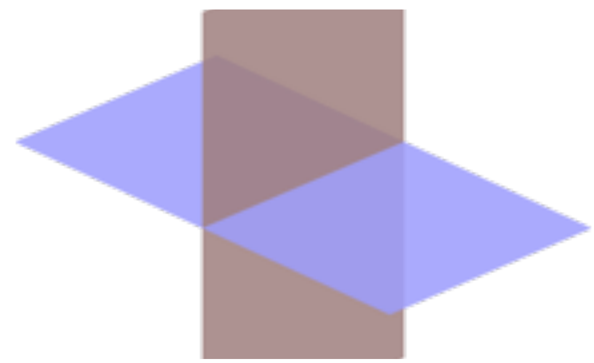


SVM

Object Recognition



2D Transforms



DLT

RANSAC

2D Alignment

$\mathbf{x} = \mathbf{P}\mathbf{X}$

camera matrix

$\mathbf{P}$

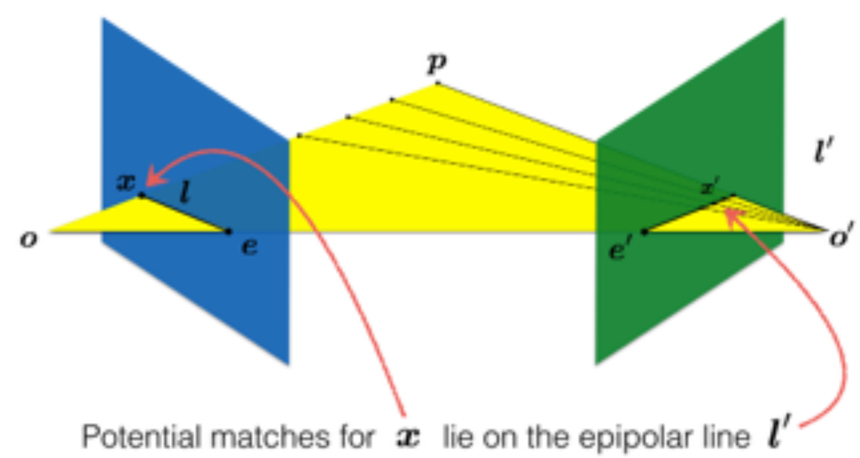
pose estimation

$\mathbf{X}$

triangulation

$\mathbf{F}$

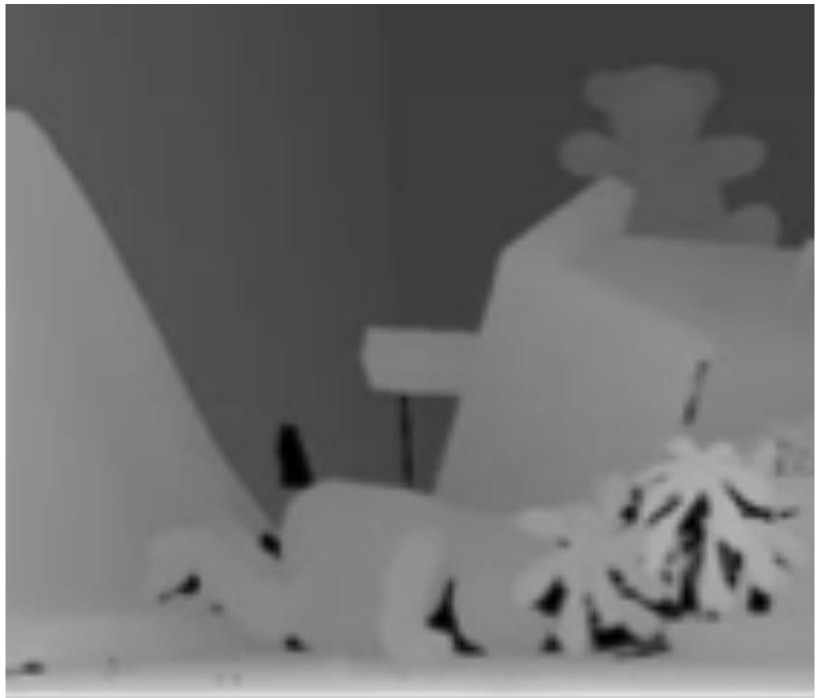
fundamental matrix



epipolar geometry

Reconstruction

2 view geometry



Block matching

Energy minimization

Stereo

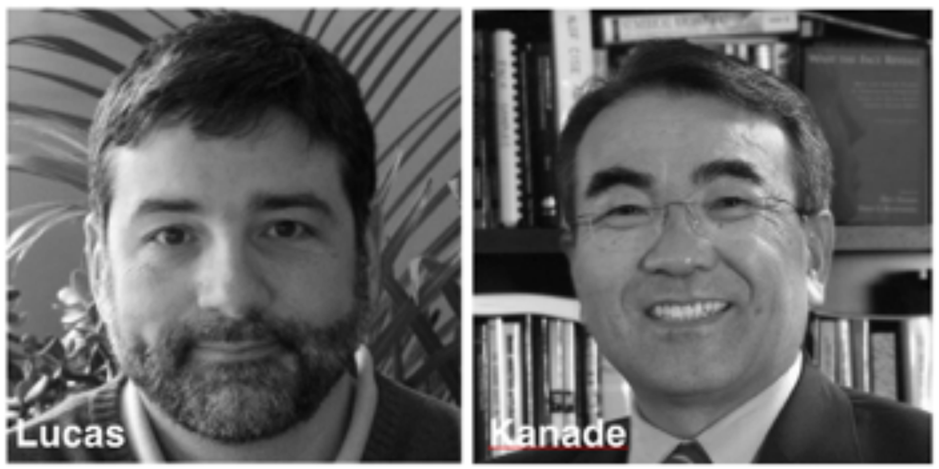
$$\begin{bmatrix} I_x(\boldsymbol{p}_1) & I_y(\boldsymbol{p}_1) \\ I_x(\boldsymbol{p}_2) & I_y(\boldsymbol{p}_2) \\ \vdots & \vdots \\ I_x(\boldsymbol{p}_{25}) & I_y(\boldsymbol{p}_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\boldsymbol{p}_1) \\ I_t(\boldsymbol{p}_2) \\ \vdots \\ I_t(\boldsymbol{p}_{25}) \end{bmatrix}$$

Constant Flow

$$\min_{\boldsymbol{u},\boldsymbol{v}} \sum_{ij} \left\{ E_d(i,j) + \lambda E_s(i,j) \right\}$$

Horn Schunck

Optical Flow

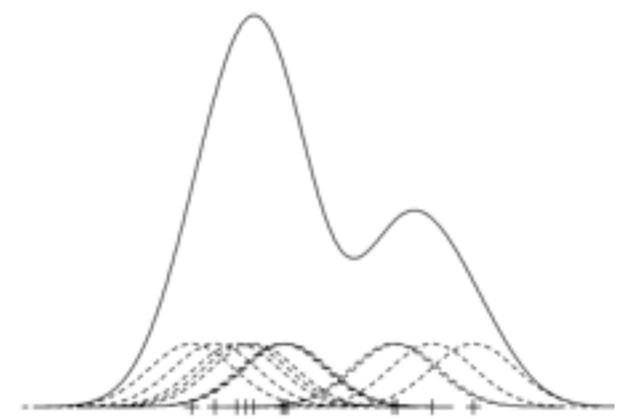


Lucas Kanade
(Forward additive)

Baker Matthews
(Inverse Compositional)

Image Alignment

KLT



Mean shift

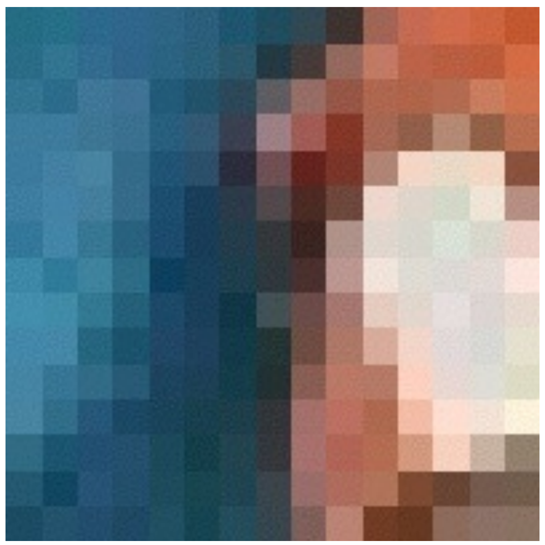
Tracking

Image Filtering

16-385 Computer Vision

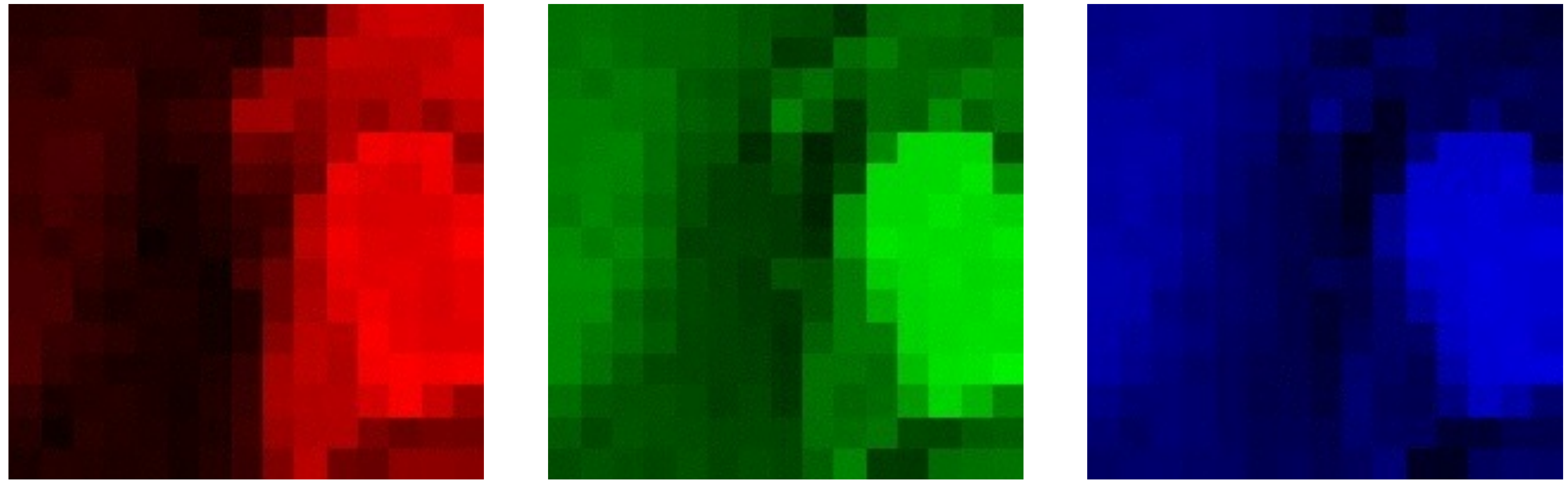

What is an image?

An image is an array of numbers

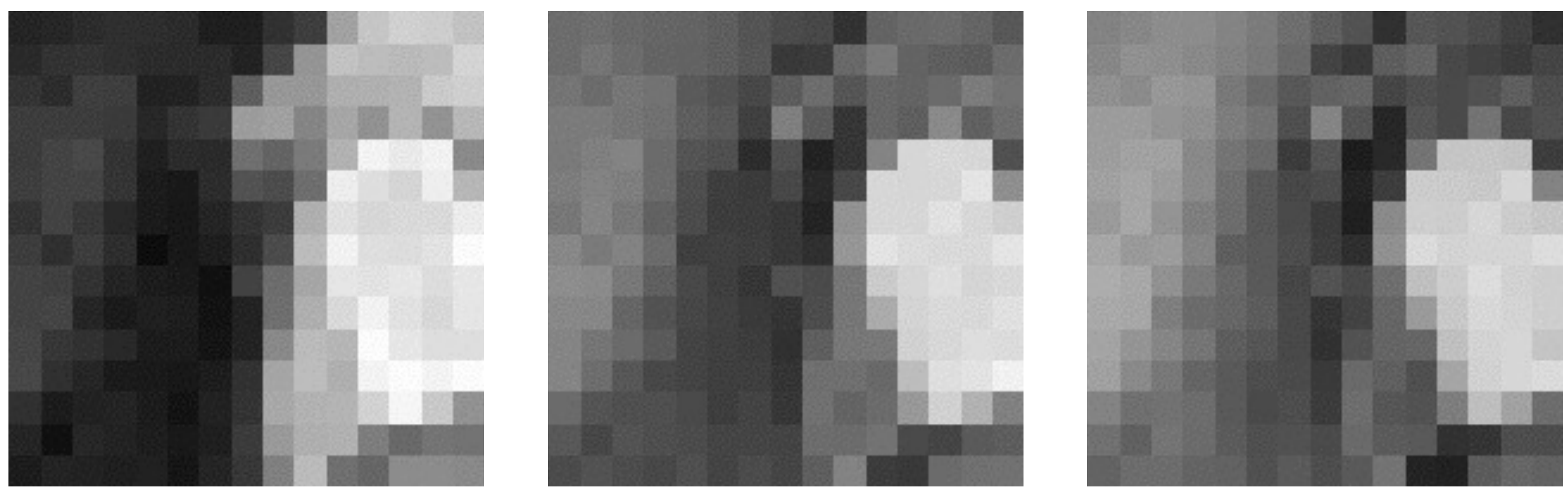


color image patch

Red | Green | Blue



colorized for visualization

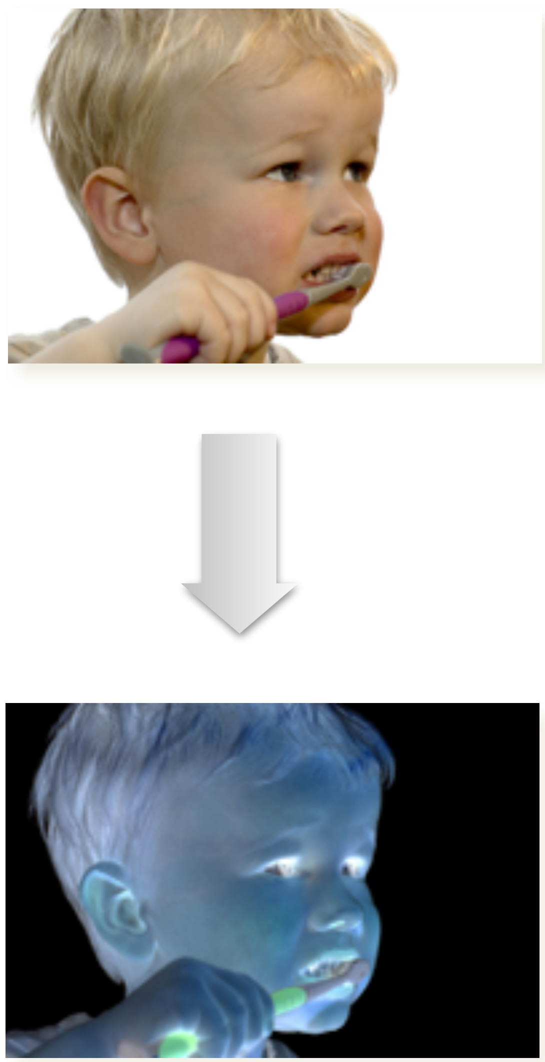


actual intensity values per channel (quantized to 256 values)

how many bits?

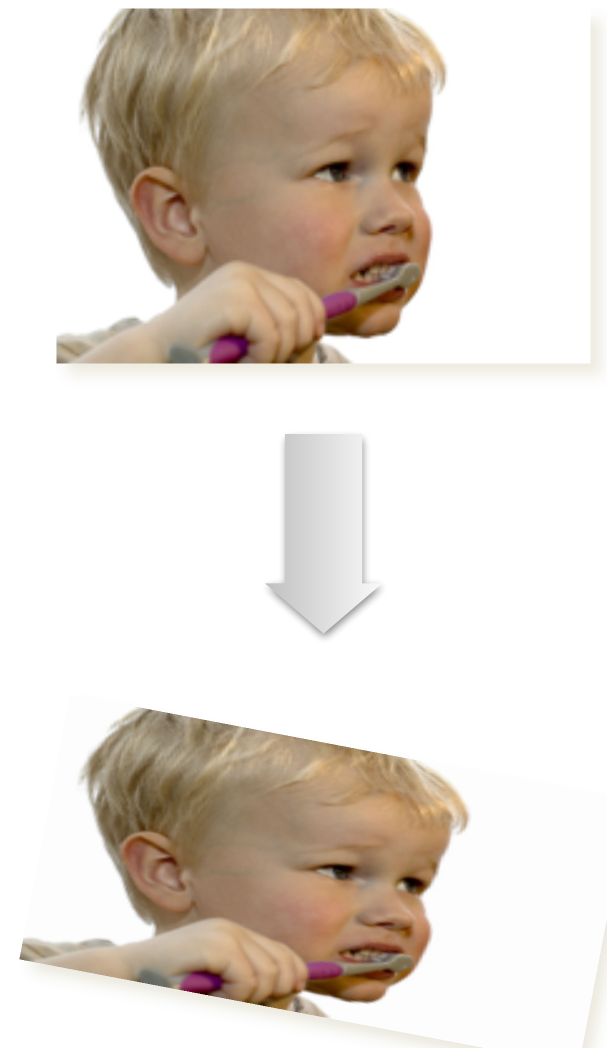
What kind of image transformations can we perform?

Filtering



changes the pixel values

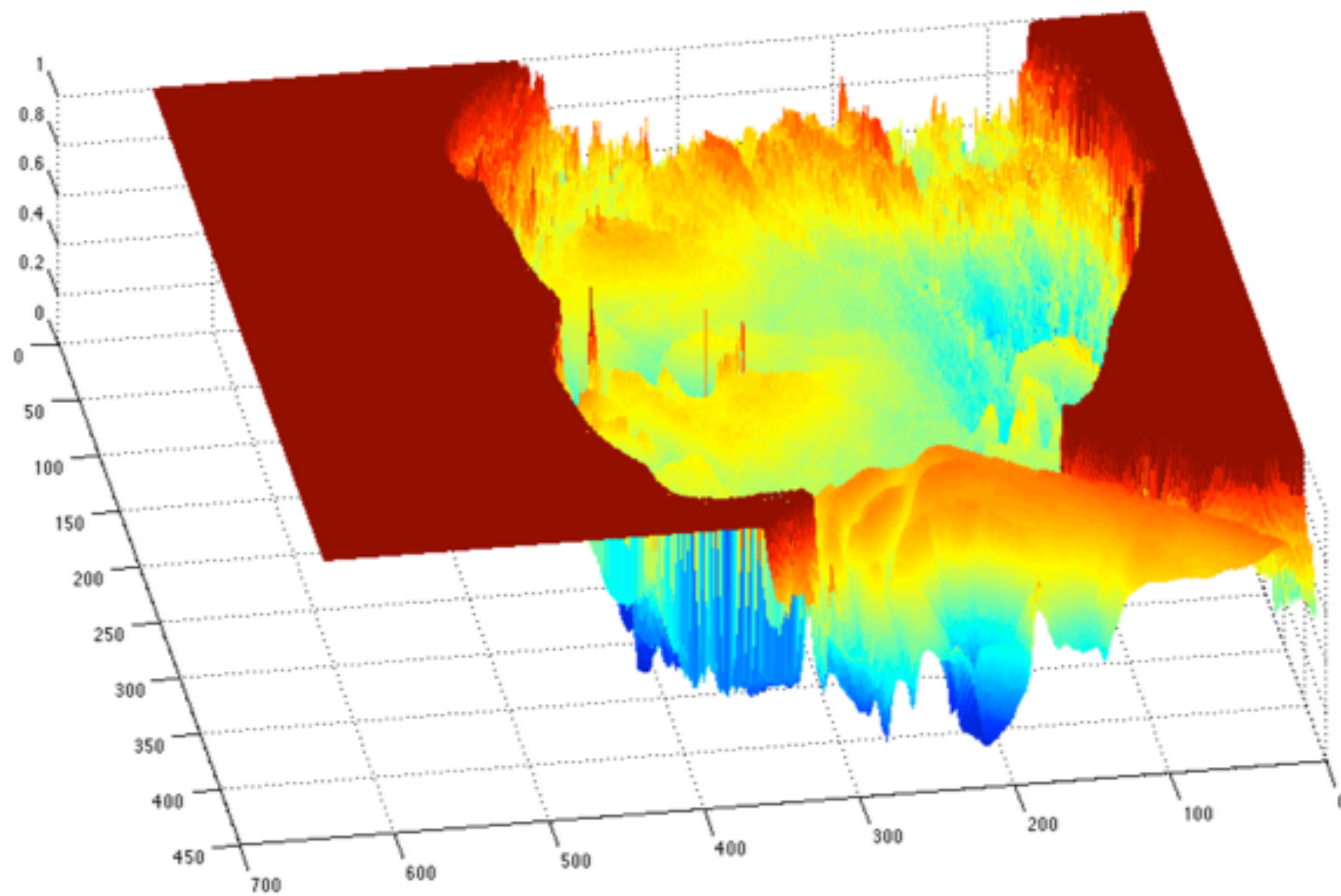
Warping



changes the pixel location

An image as a 2D function

$$f(\boldsymbol{x}) \qquad \boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

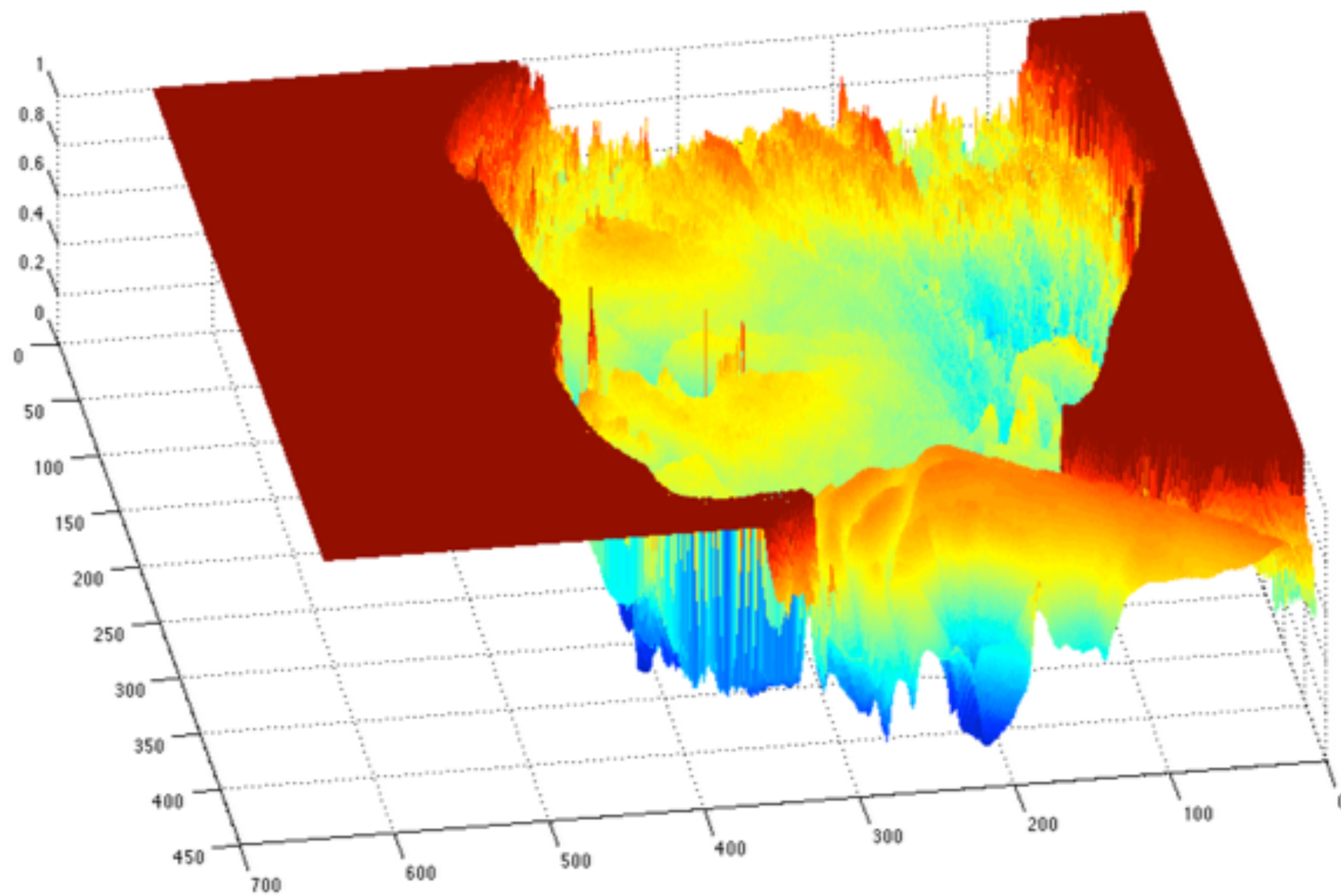


An image as a 2D function

$$f(\boldsymbol{x})$$

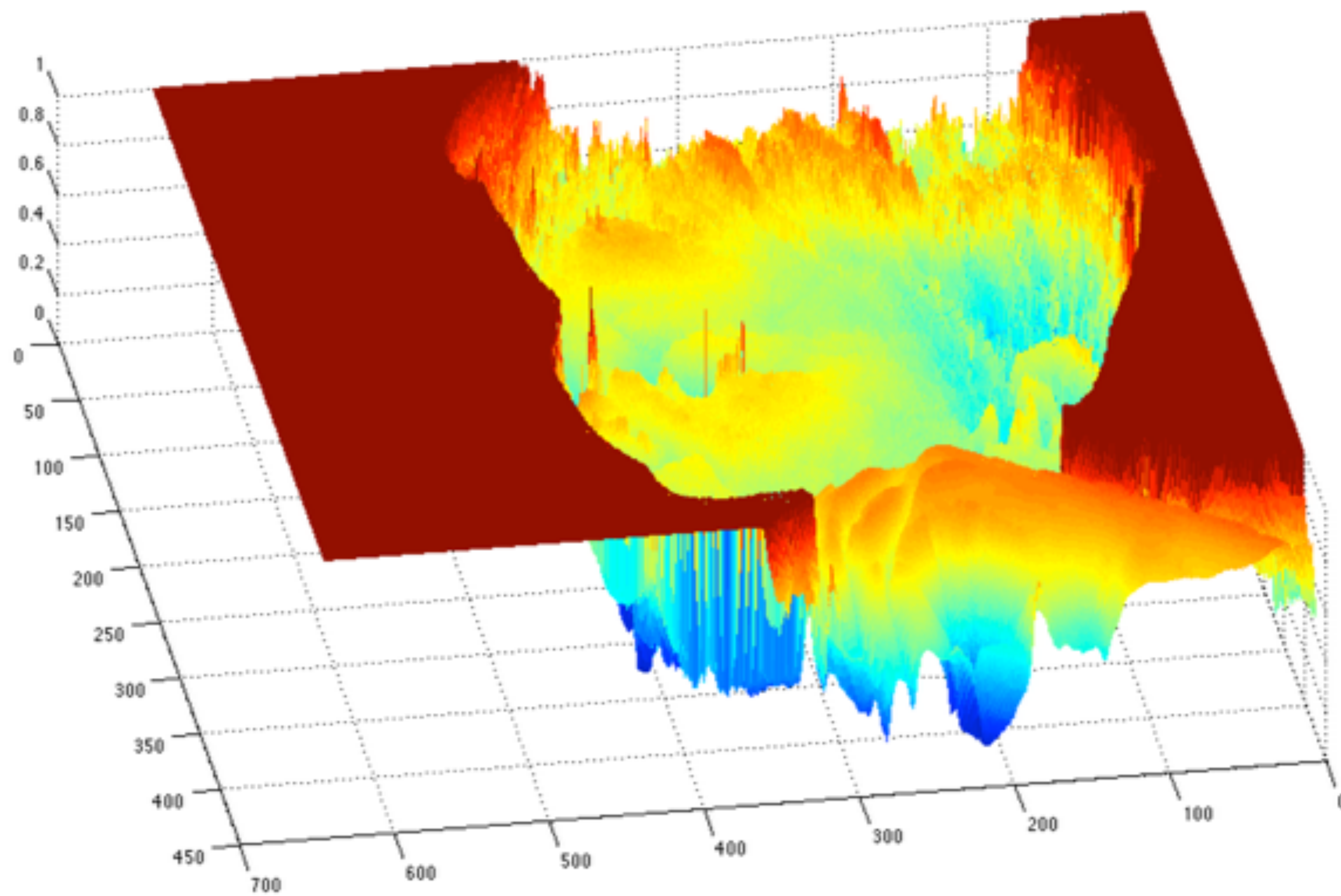
$$\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

What is the range of $f(\boldsymbol{x})$?



An image as a 2D function

$$f(\boldsymbol{x}) \qquad \boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$



What is the range of $f(\boldsymbol{x})$?

8-bit image: 256 values

What kind of image transformations can we perform?

Filtering

F

$$G(\boldsymbol{x}) = h\{F(\boldsymbol{x})\}$$

G

changes the **range** of image

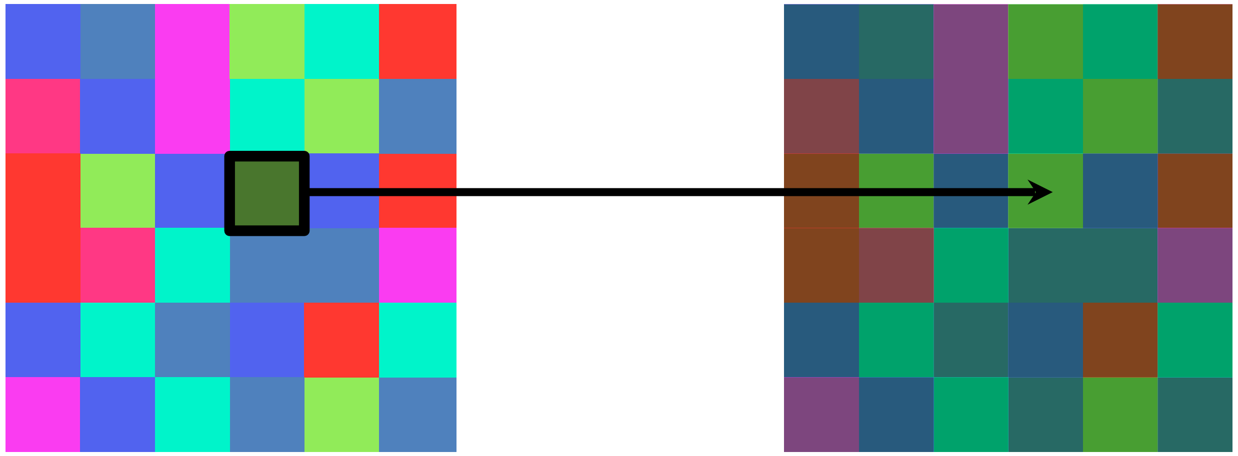
Warping

$$G(\boldsymbol{x}) = F(h\{\boldsymbol{x}\})$$

changes the **domain** of image

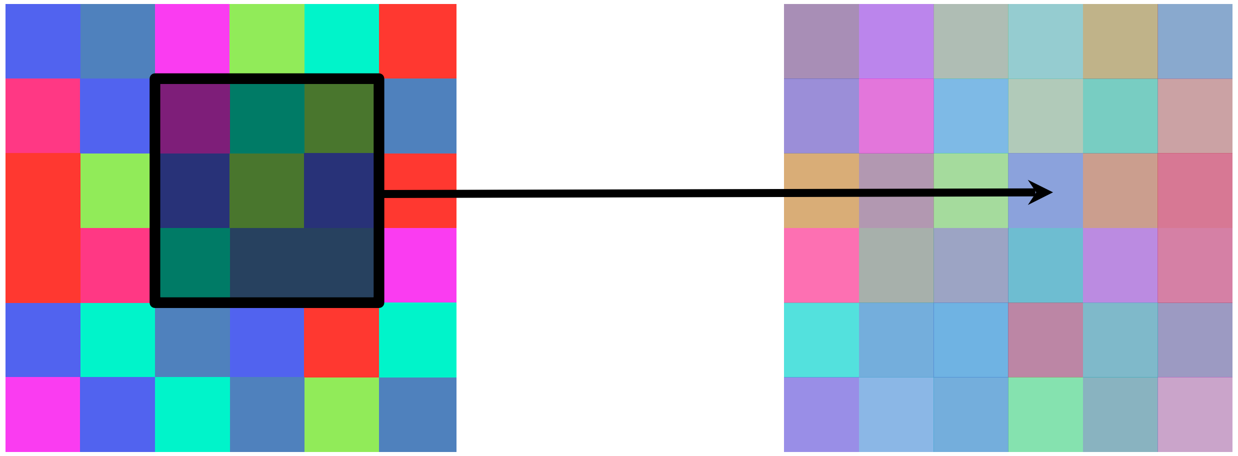
What kind of image filtering can we perform?

Point Operation



point processing

Neighborhood Operation

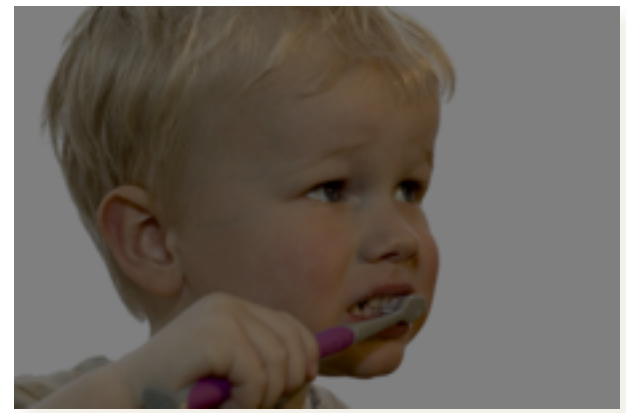


filtering

Examples of Point Processing


Original


Darken


Lower Contrast


Nonlinear Lower Contrast


Invert


Lighten


Raise Contrast


Nonlinear Raise Contrast

Examples of filtering

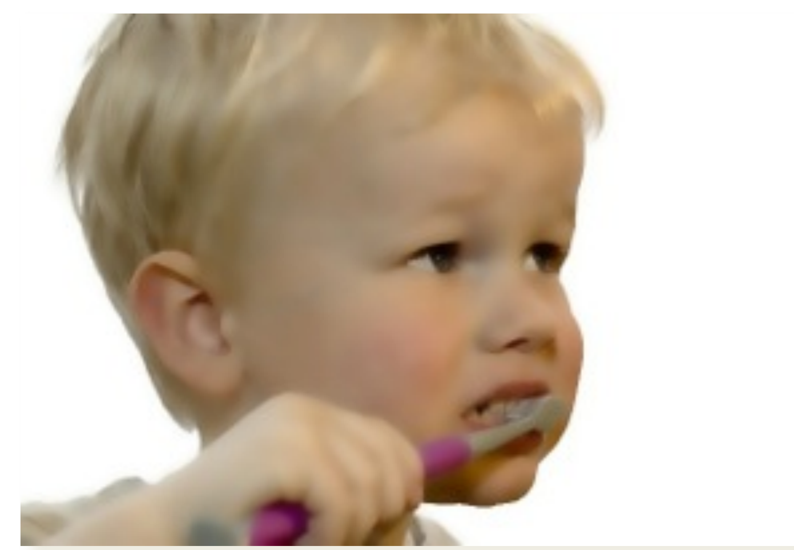
Original

Gradient Magnitude

Gaussian Blur

Median

Adaptive Thresholding



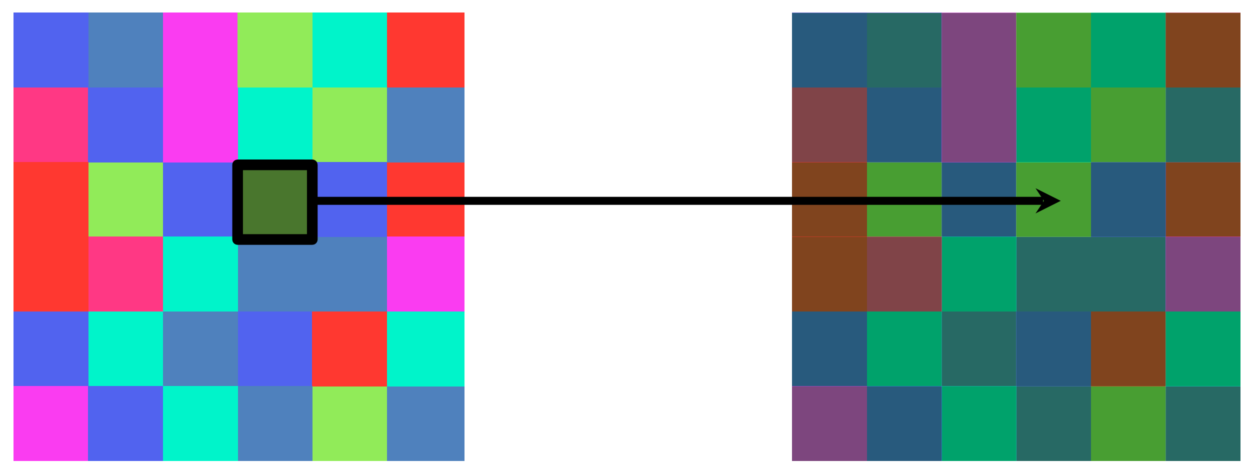
Bilateral

Point Processing

16-385 Computer Vision

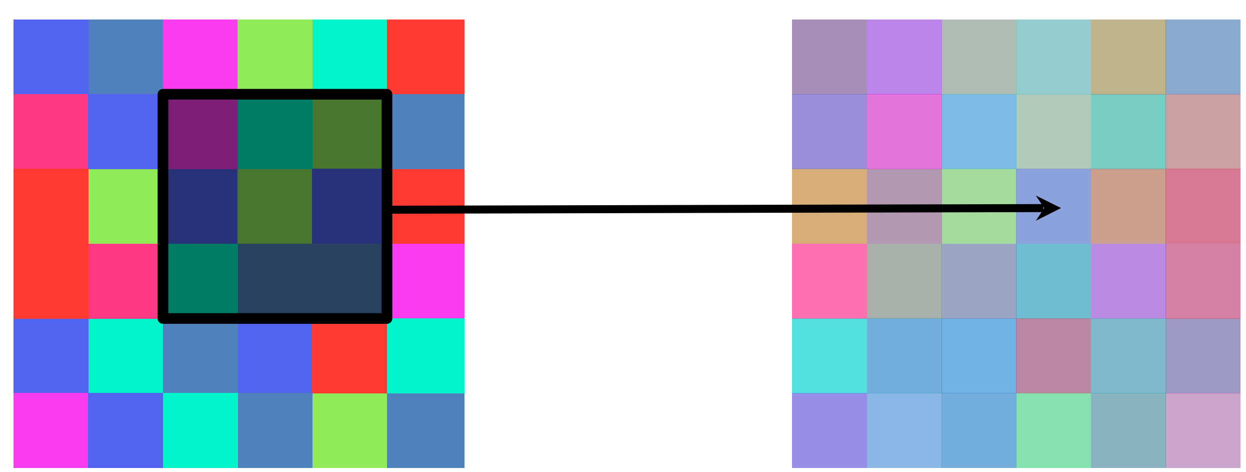
What kind of image filtering can we perform?

Point Operation



point processing

Neighborhood Operation

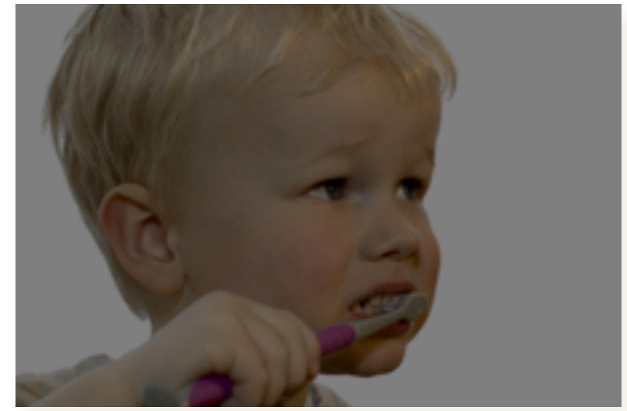


filtering

Original

Darken

Lower Contrast



Nonlinear Lower Contrast

x

pixel value

Invert

Lighten

Raise Contrast

Nonlinear Raise Contrast

Original

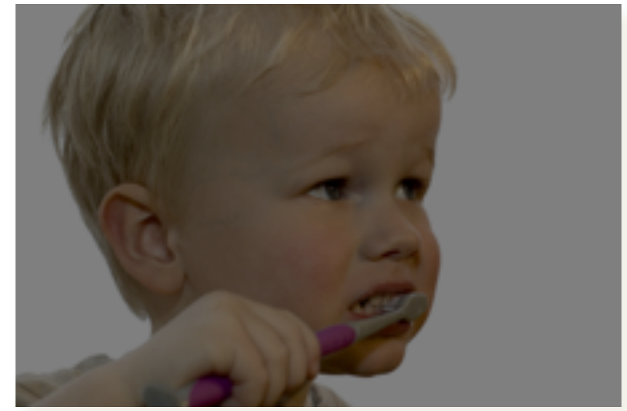
x

Darken

$x - 128$

how would you code this?

Lower Contrast



Nonlinear Lower Contrast

Invert

Lighten

Raise Contrast

Nonlinear Raise Contrast

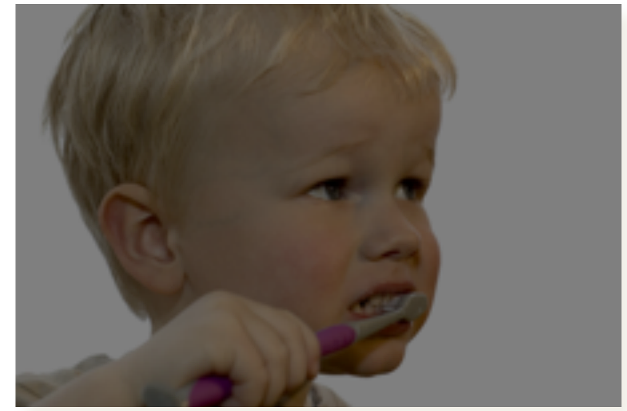
Original

$$x$$

Darken

$$x - 128$$

Lower Contrast



$$\frac{x}{2}$$

Nonlinear Lower Contrast

Invert

Lighten

Raise Contrast

Nonlinear Raise Contrast

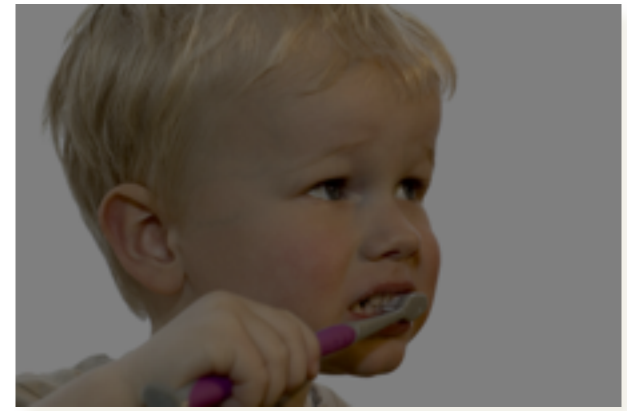
Original

$$x$$

Darken

$$x - 128$$

Lower Contrast



$$\frac{x}{2}$$

Nonlinear Lower Contrast

$$\left(\frac{x}{255}\right)^{1/3} \times 255$$

Invert

Lighten

Raise Contrast

Nonlinear Raise Contrast

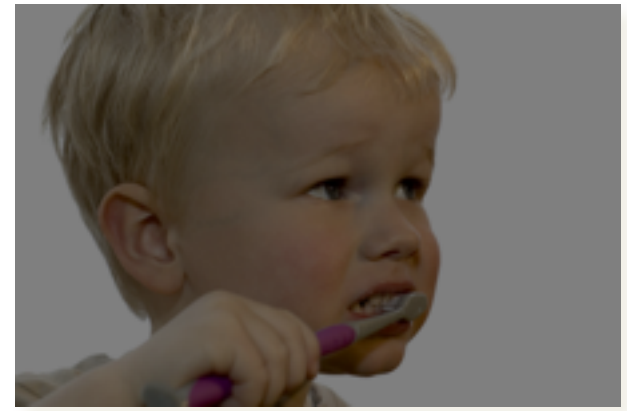
Original

$$x$$

Darken

$$x - 128$$

Lower Contrast



$$\frac{x}{2}$$

Nonlinear Lower Contrast

$$\left(\frac{x}{255}\right)^{1/3} \times 255$$

Invert

$$255 - x$$

Lighten

Raise Contrast

Nonlinear Raise Contrast

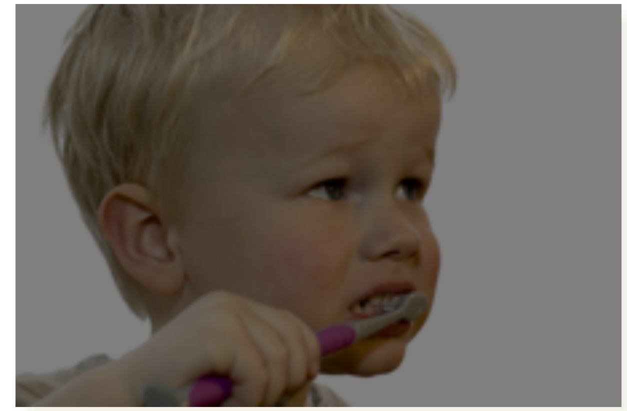
Original

$$x$$

Darken

$$x - 128$$

Lower Contrast



$$\frac{x}{2}$$

Nonlinear Lower Contrast

$$\left(\frac{x}{255}\right)^{1/3} \times 255$$

Invert

$$255 - x$$

Lighten

$$x + 128$$

Raise Contrast

Nonlinear Raise Contrast

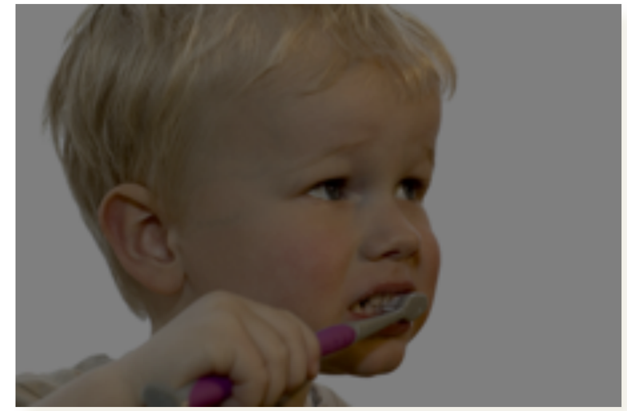
Original

$$x$$

Darken

$$x - 128$$

Lower Contrast



$$\frac{x}{2}$$

Nonlinear Lower Contrast

$$\left(\frac{x}{255}\right)^{1/3} \times 255$$

Invert

$$255 - x$$

Lighten

$$x + 128$$

Raise Contrast

$$x \times 2$$

Nonlinear Raise Contrast

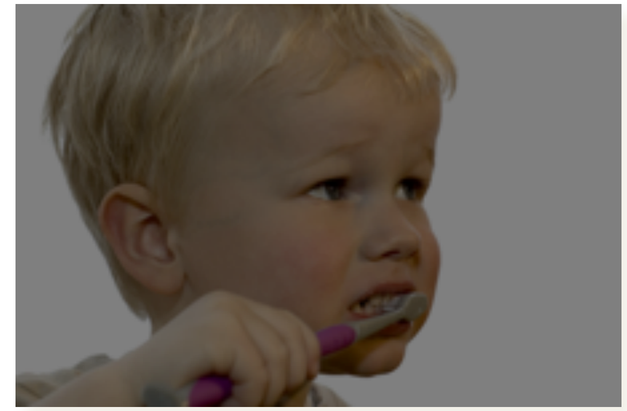
Original

$$x$$

Darken

$$x - 128$$

Lower Contrast



$$\frac{x}{2}$$

Nonlinear Lower Contrast

$$\left(\frac{x}{255}\right)^{1/3} \times 255$$

Invert

$$255 - x$$

Lighten

$$x + 128$$

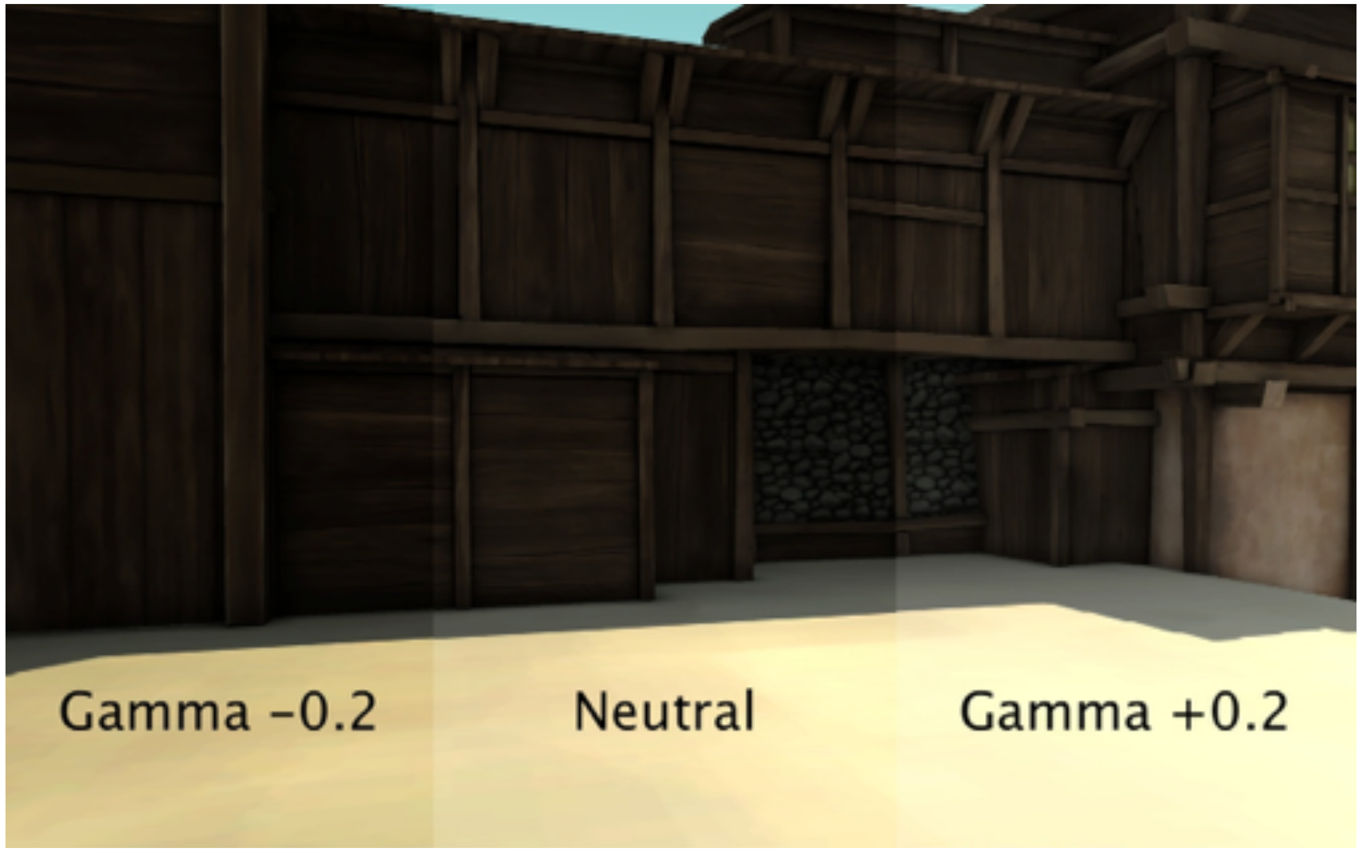
Raise Contrast

$$x \times 2$$

Nonlinear Raise Contrast

$$\left(\frac{x}{255}\right)^{2} \times 255$$

Other point processes



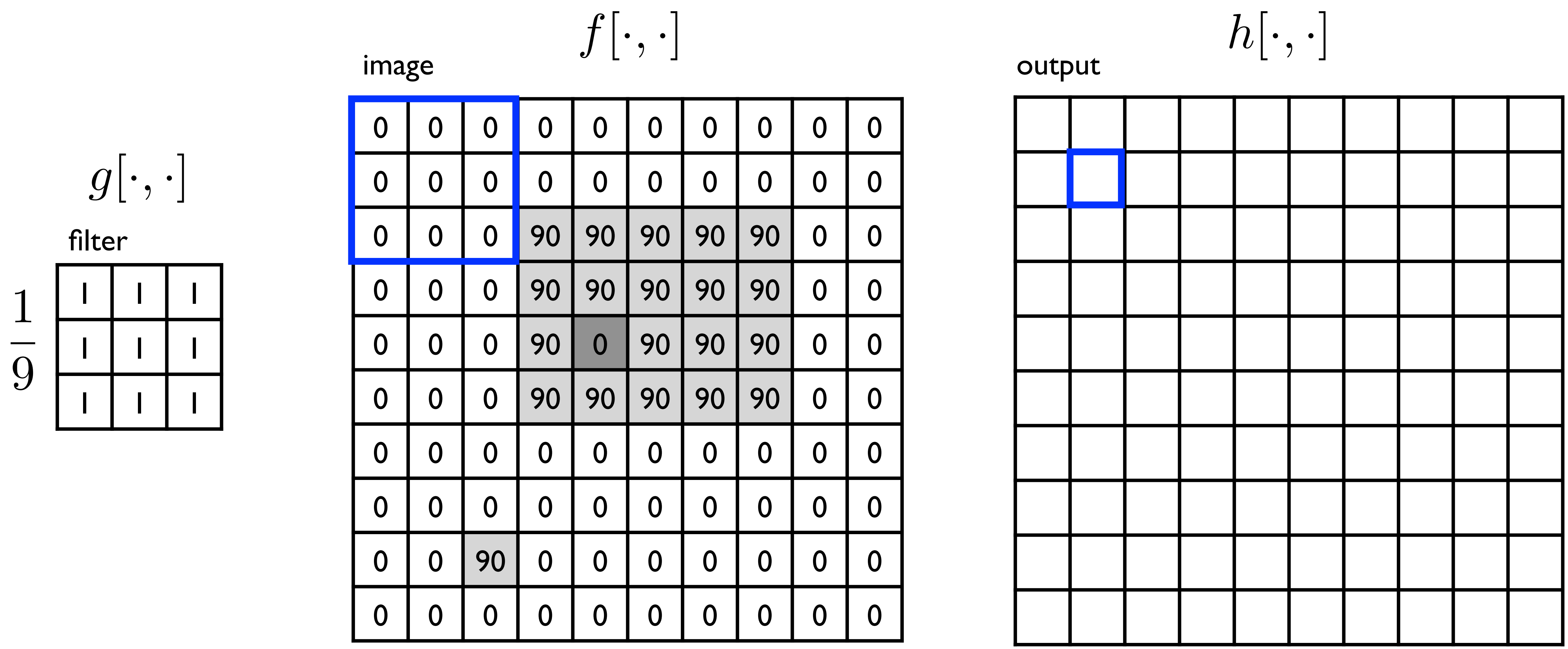
Box Filter

The ‘Box’ filter

$$g[\cdot, \cdot] = \frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

replaces pixel with local average

has a smoothing effect



$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[\cdot,\cdot]$

image

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[\cdot,\cdot]$

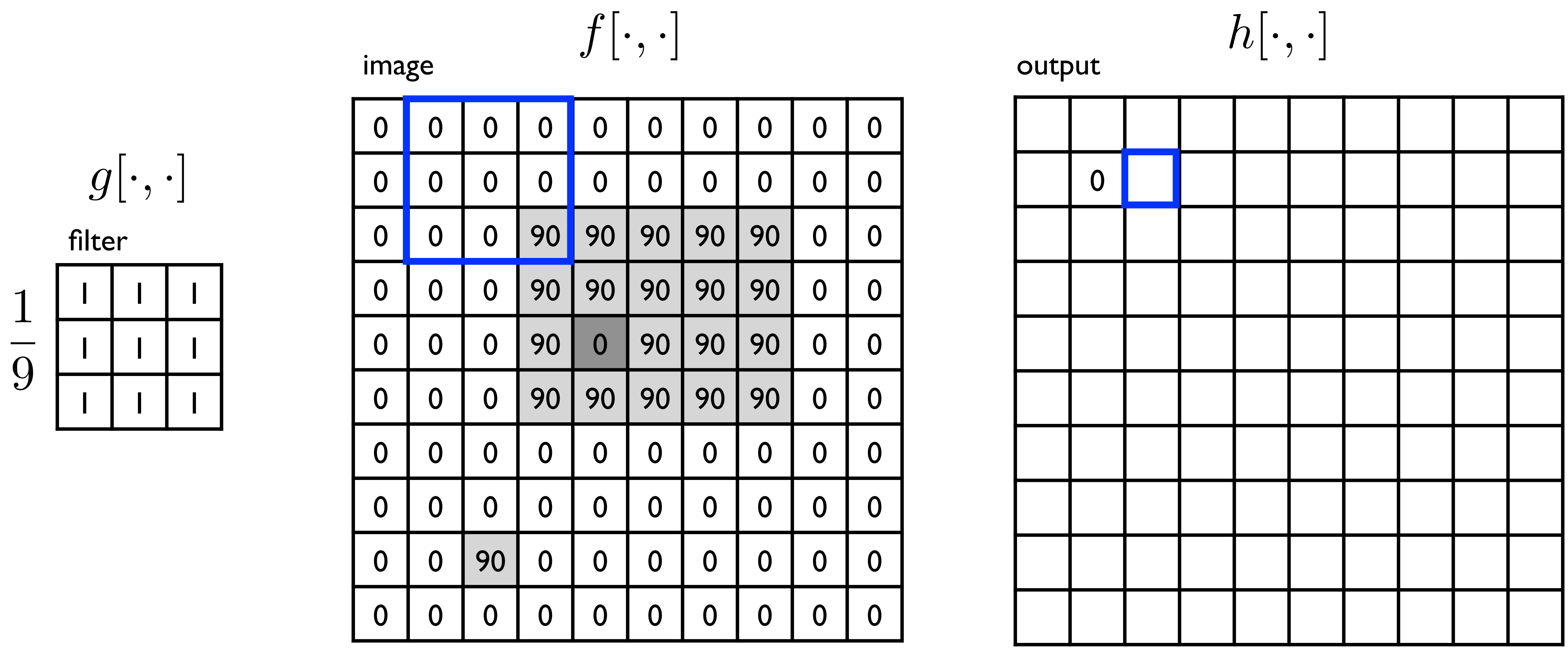
output

	0								

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

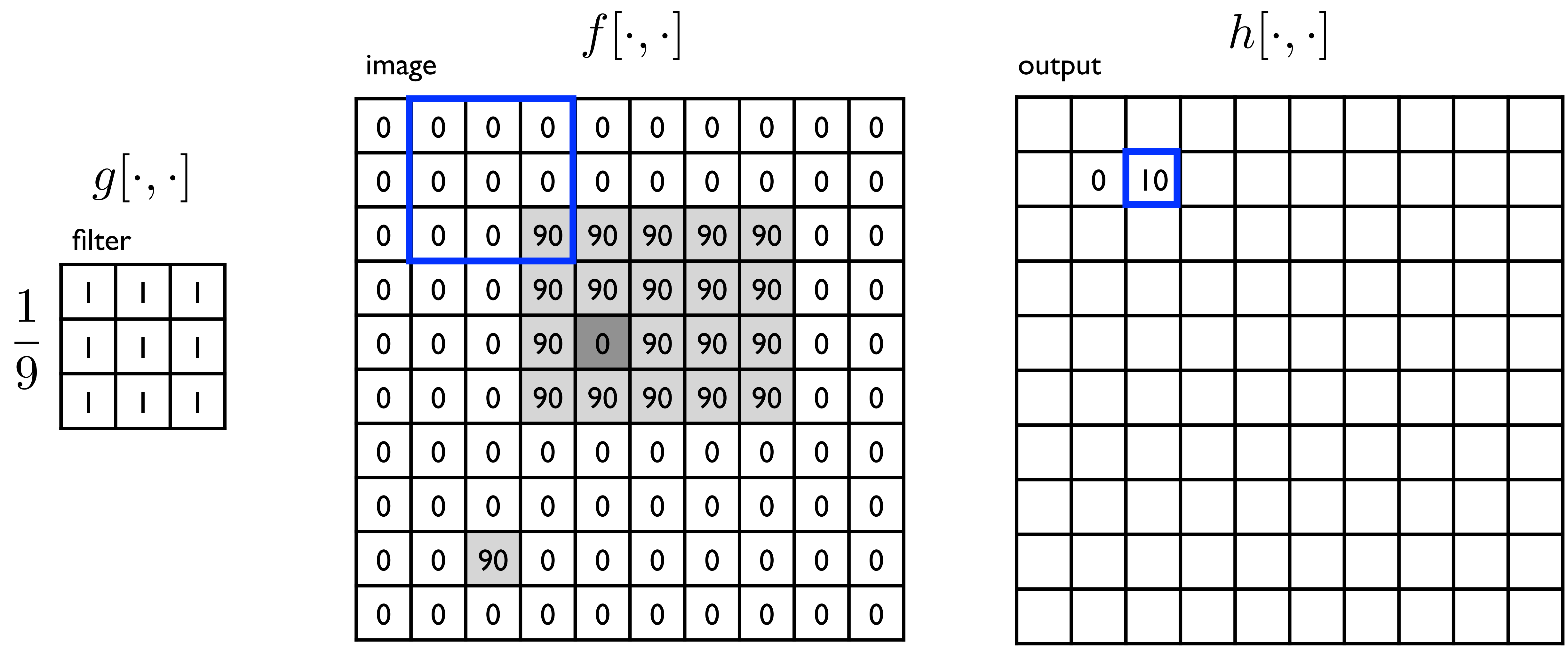
* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l]\, f[m+k, n+l]$$

output — filter — image (signal)

* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

image $f[\cdot,\cdot]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

output $h[\cdot,\cdot]$

	0	10							

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output — filter — image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[\cdot,\cdot]$

image

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[\cdot,\cdot]$

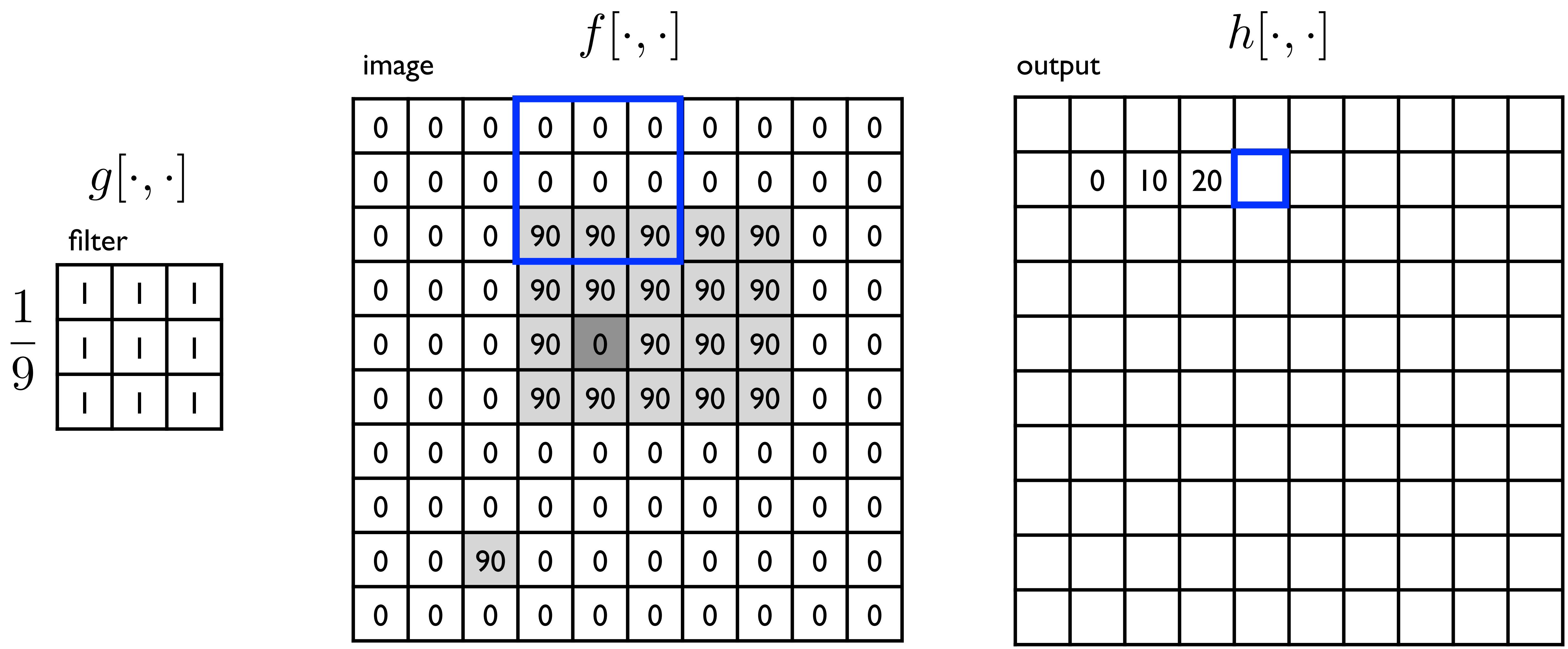
output

	0	10	20						

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

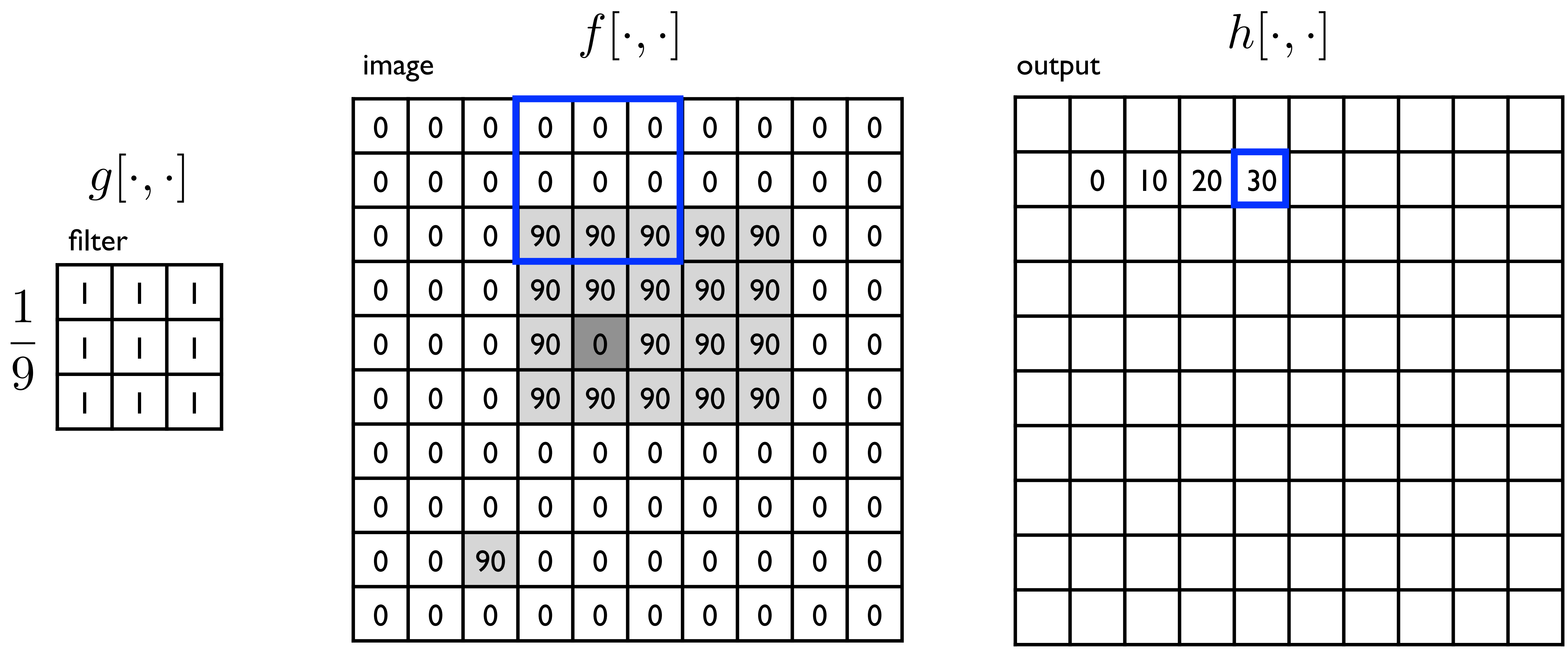
* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

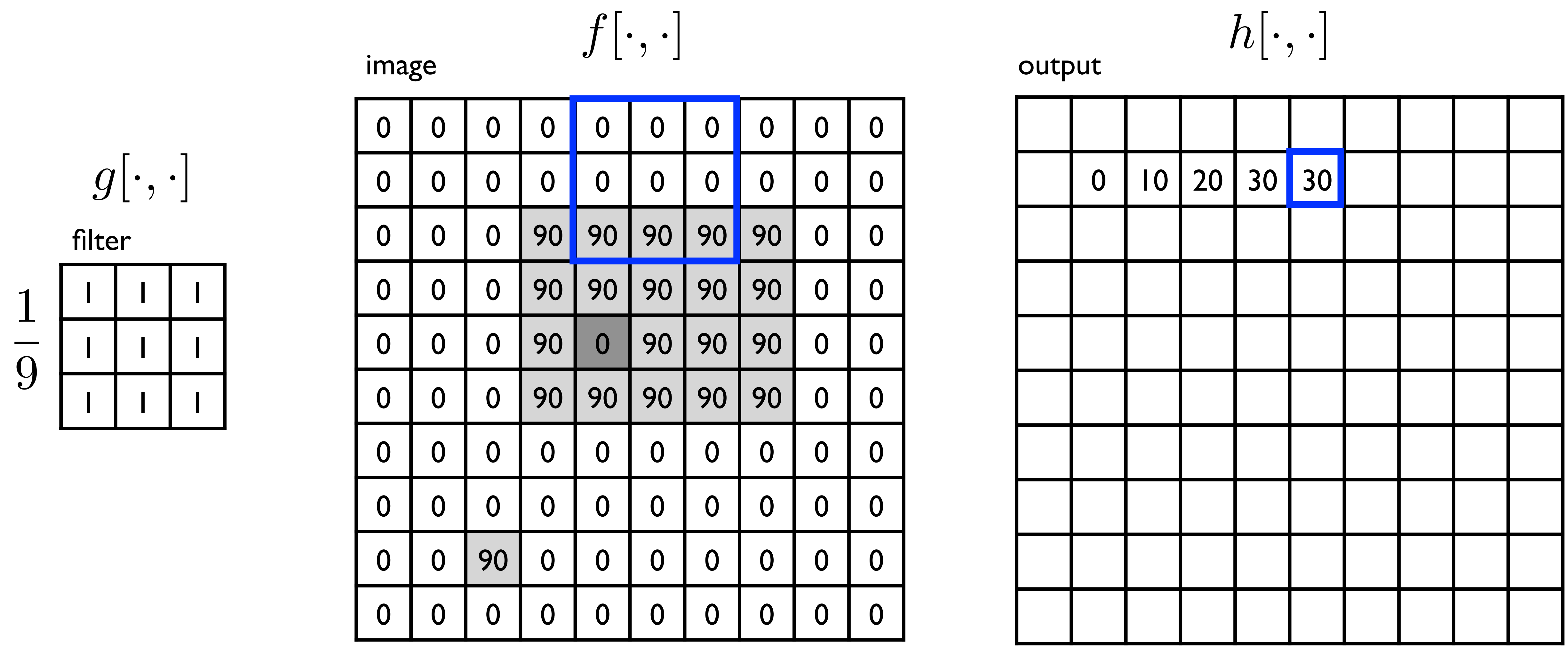
* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

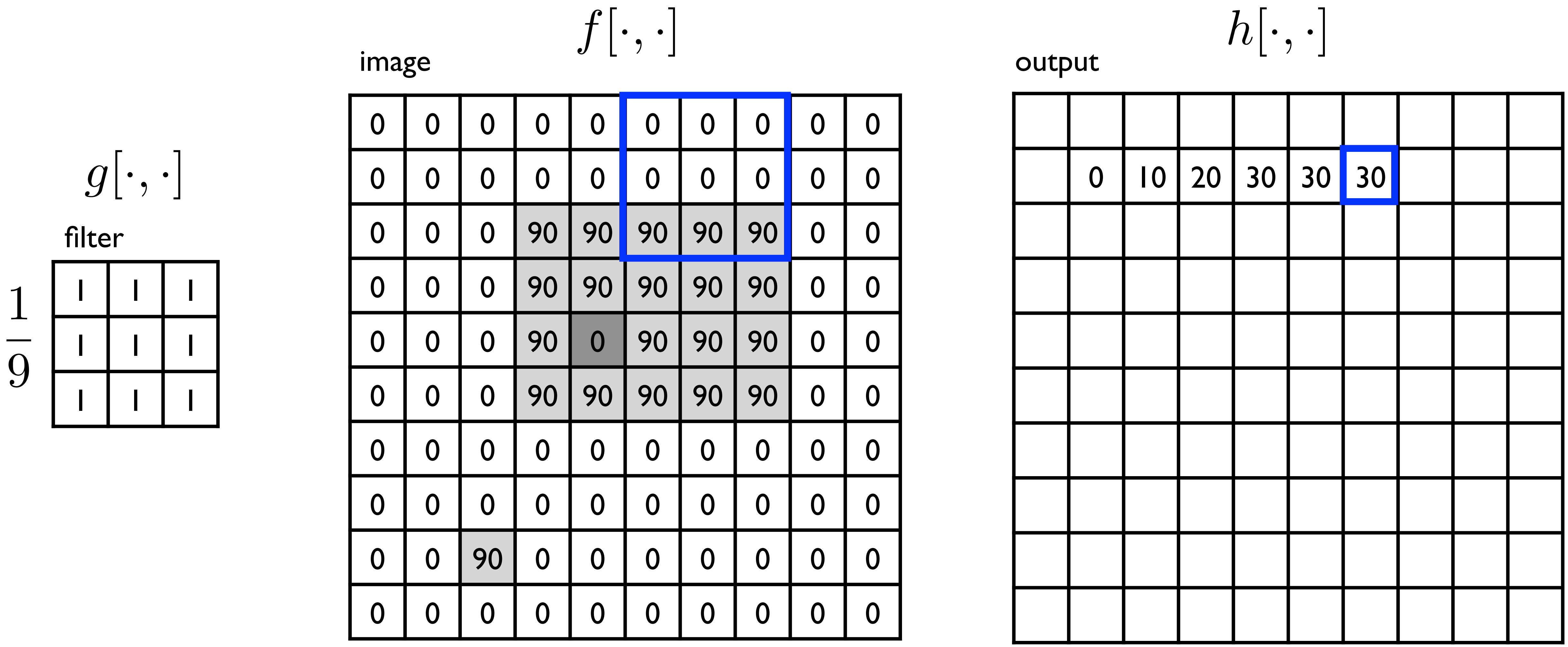
* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l]\, f[m+k, n+l]$$

output filter image (signal)

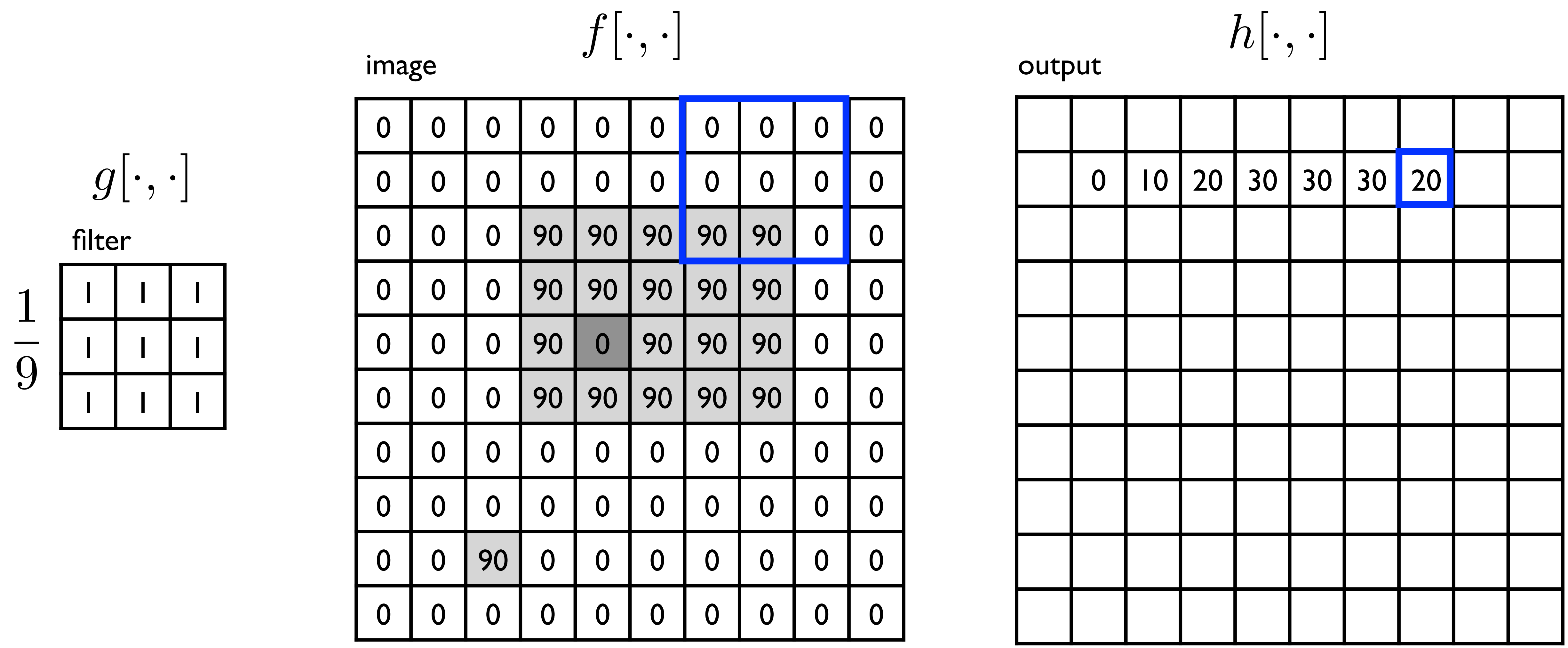
* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[\cdot,\cdot]$

image

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[\cdot,\cdot]$

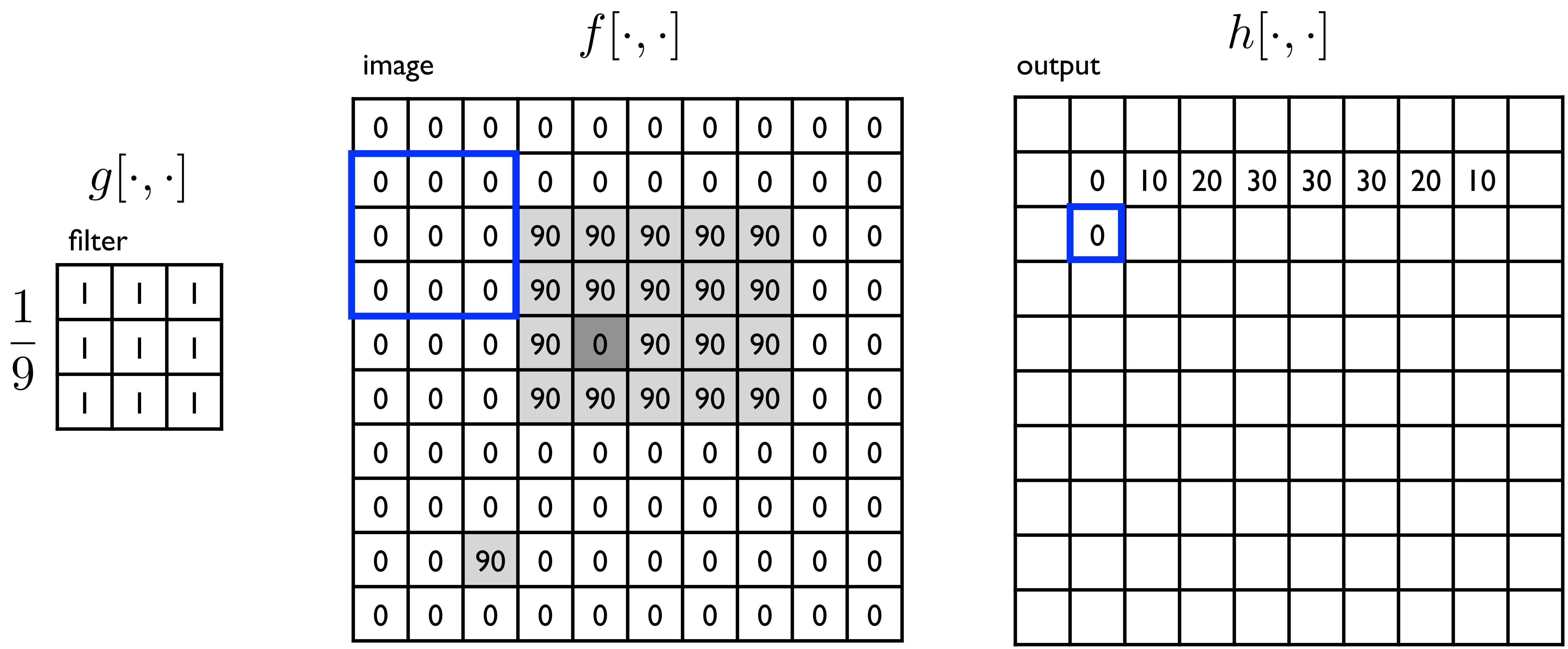
output

	0	10	20	30	30	30	20	10	

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output — filter — image (signal)

* some zero values are white for visualization but they should be black



$$h[m,n] = \sum_{k,l} g[k,l]\, f[m+k, n+l]$$

output filter image (signal)

* some zero values are white for visualization but they should be black

filter $g[\cdot,\cdot]$

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

image $f[\cdot,\cdot]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

output $h[\cdot,\cdot]$

	0	10	20	30	30	30	20	10	
	0	20							

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output — filter — image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

image $f[\cdot,\cdot]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

output $h[\cdot,\cdot]$

	0	10	20	30	30	30	20	10	
	0	20	40						

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

* some zero values are white for visualization but they should be black

filter $g[\cdot,\cdot]$

$$\frac{1}{9}$$

1	1	1
1	1	1
1	1	1

image $f[\cdot,\cdot]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

output $h[\cdot,\cdot]$

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0								

$$h[m,n] = \sum_{k,l} g[k,l]\, f[m+k, n+l]$$

output   filter   image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

image $f[\cdot,\cdot]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

output $h[\cdot,\cdot]$

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30							

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output — filter — image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

image $f[\cdot,\cdot]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

output $h[\cdot,\cdot]$

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	0	10	20	30	30	30	20	10	
	10	10	10	10	0	0	0	0	
	10								

$$h[m,n] = \sum_{k,l} g[k,l]\, f[m+k, n+l]$$

output — filter — image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

$f[\cdot,\cdot]$

image

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$h[\cdot,\cdot]$

output

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	0	10	20	30	30	30	20	10	
	10	10	10	10	0	0	0	0	
	10	10	10	10	0	0	0	0	

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

output filter image (signal)

* some zero values are white for visualization but they should be black

$g[\cdot,\cdot]$

filter

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

image $f[\cdot,\cdot]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

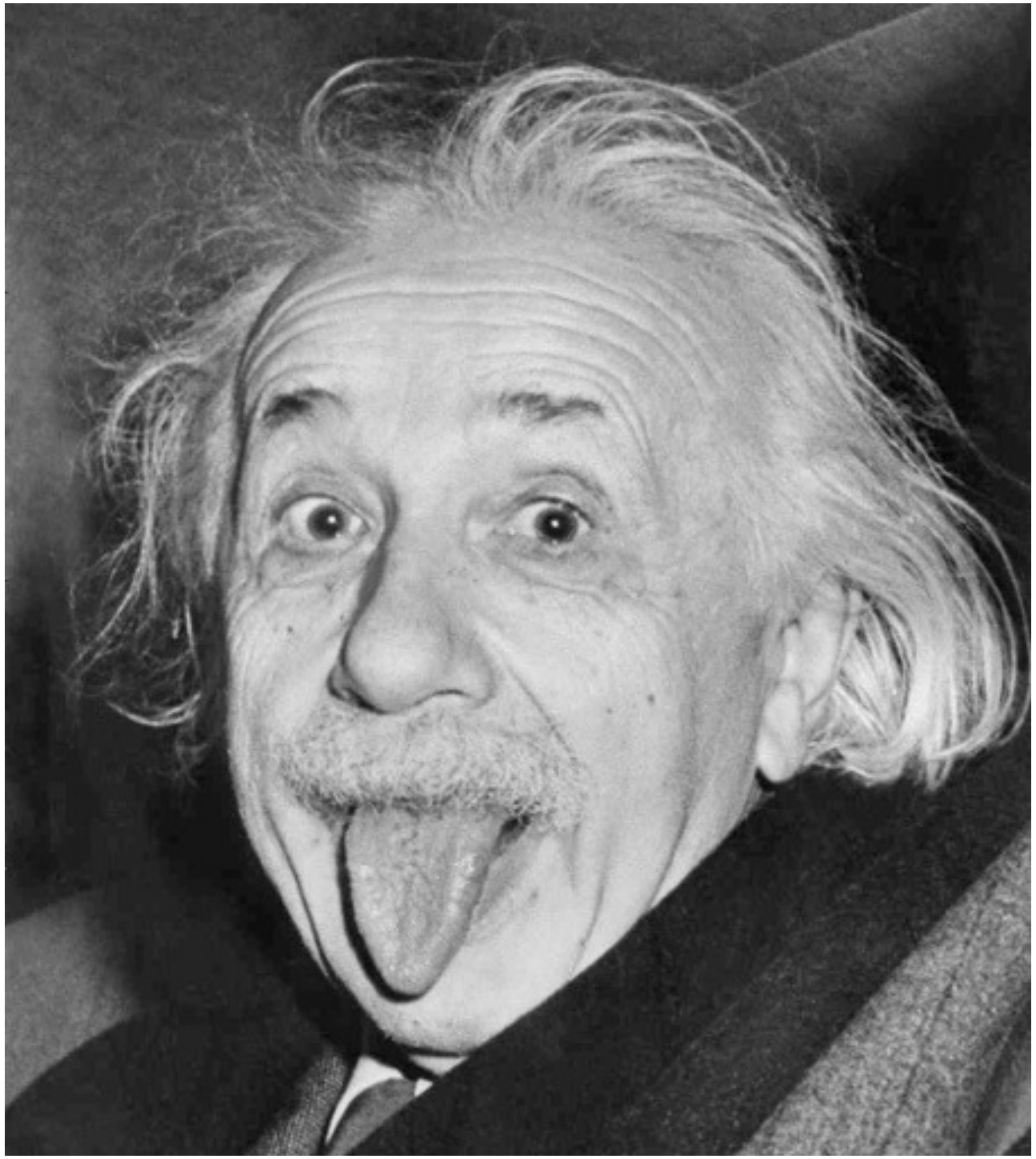
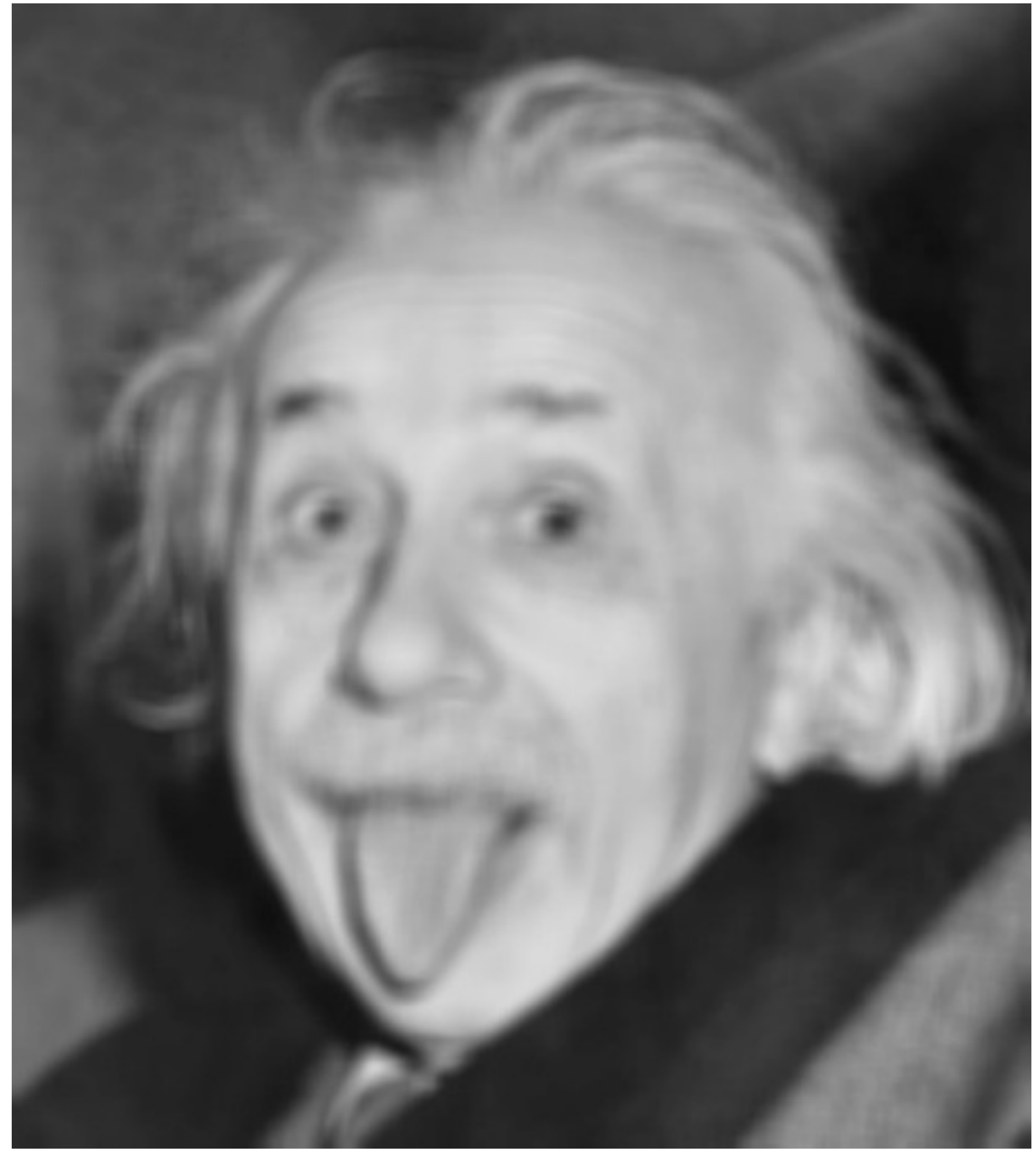
output $h[\cdot,\cdot]$

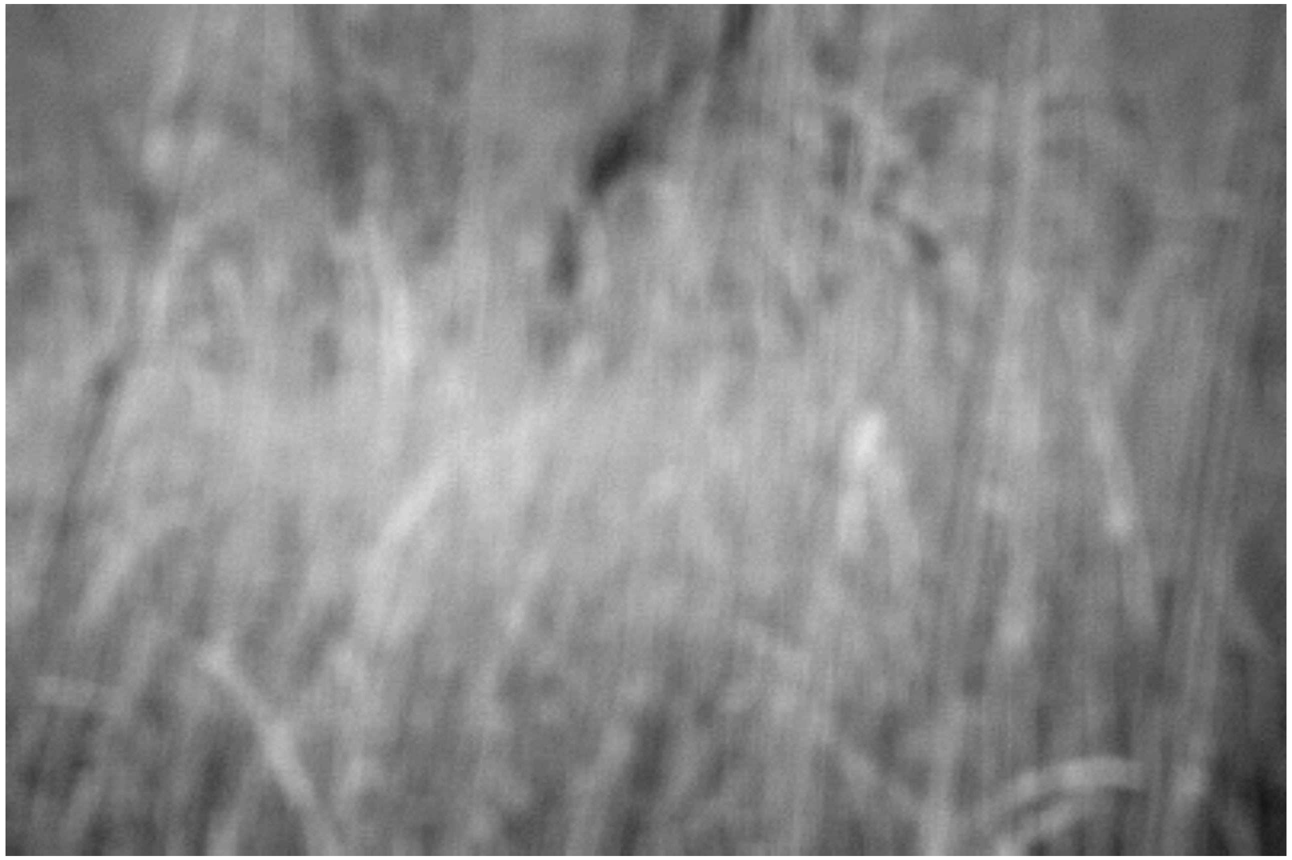
	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	0	10	20	30	30	30	20	10	
	10	10	10	10	0	0	0	0	
	10	10	10	10	0	0	0	0	

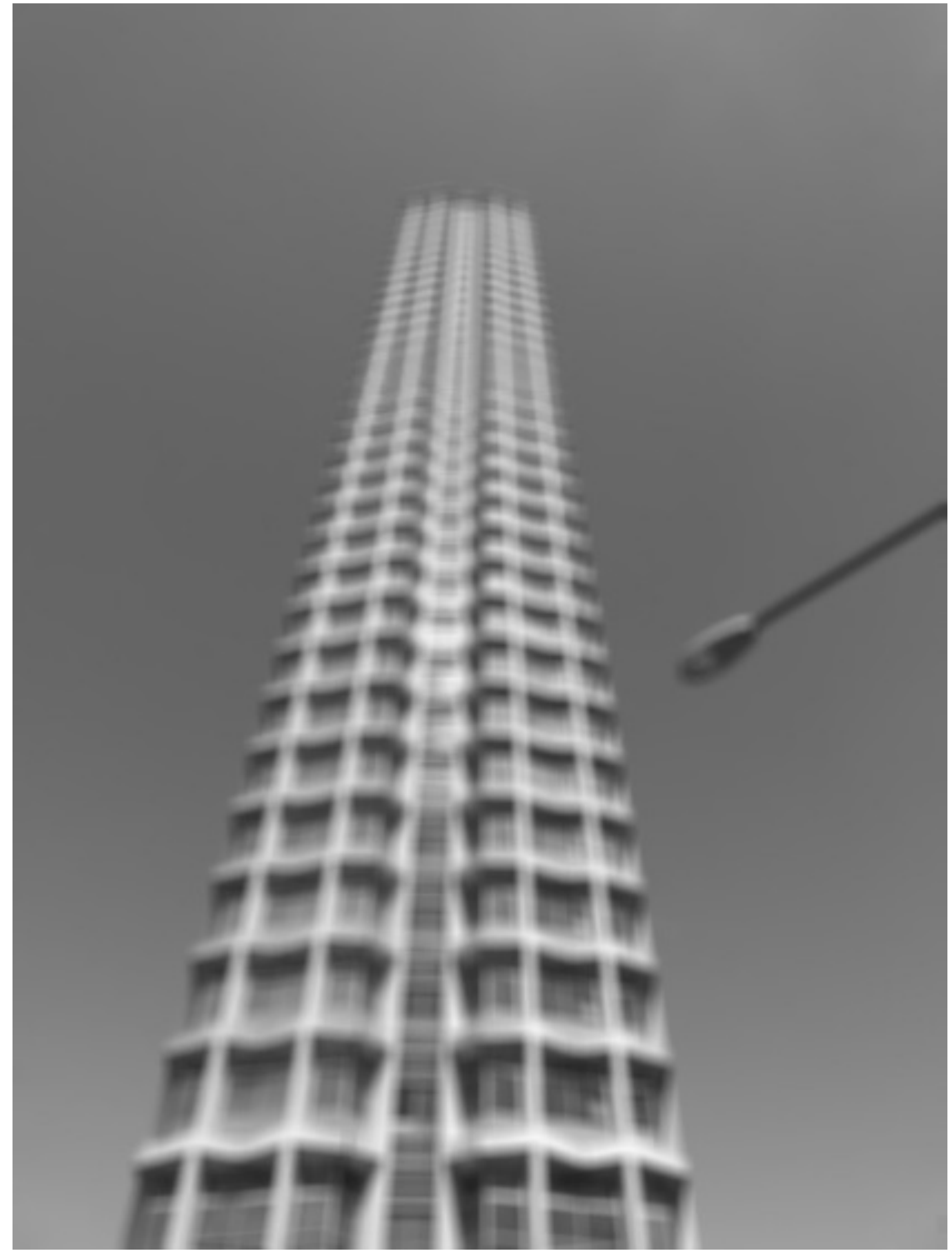
$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

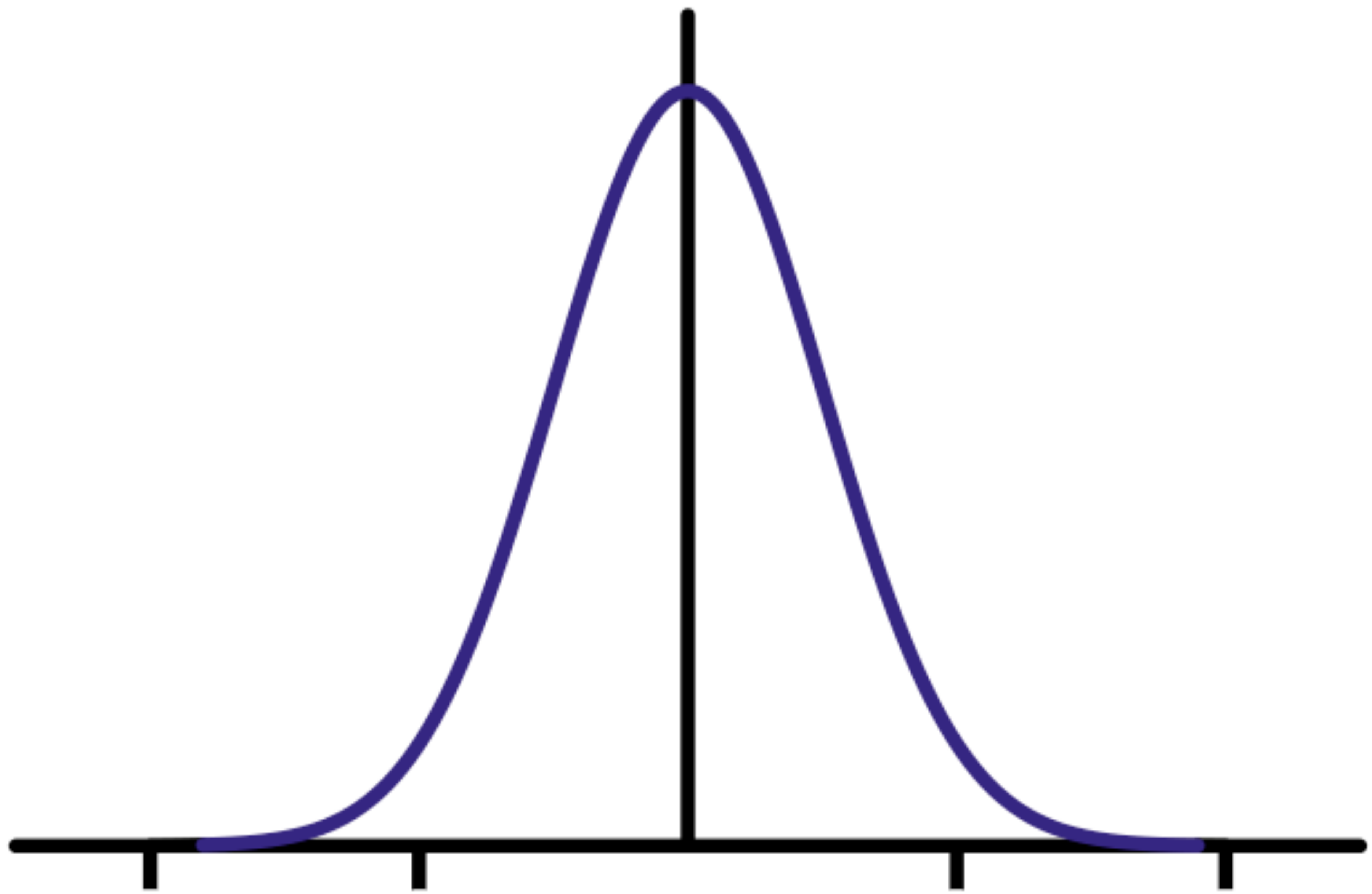
output filter image (signal)

* some zero values are white for visualization but they should be black







Gaussian Filter

The Gaussian filter

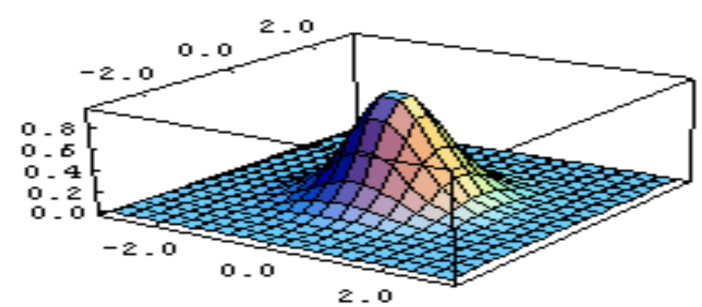
$\frac{1}{16}$

1	2	1
2	4	2
1	2	1

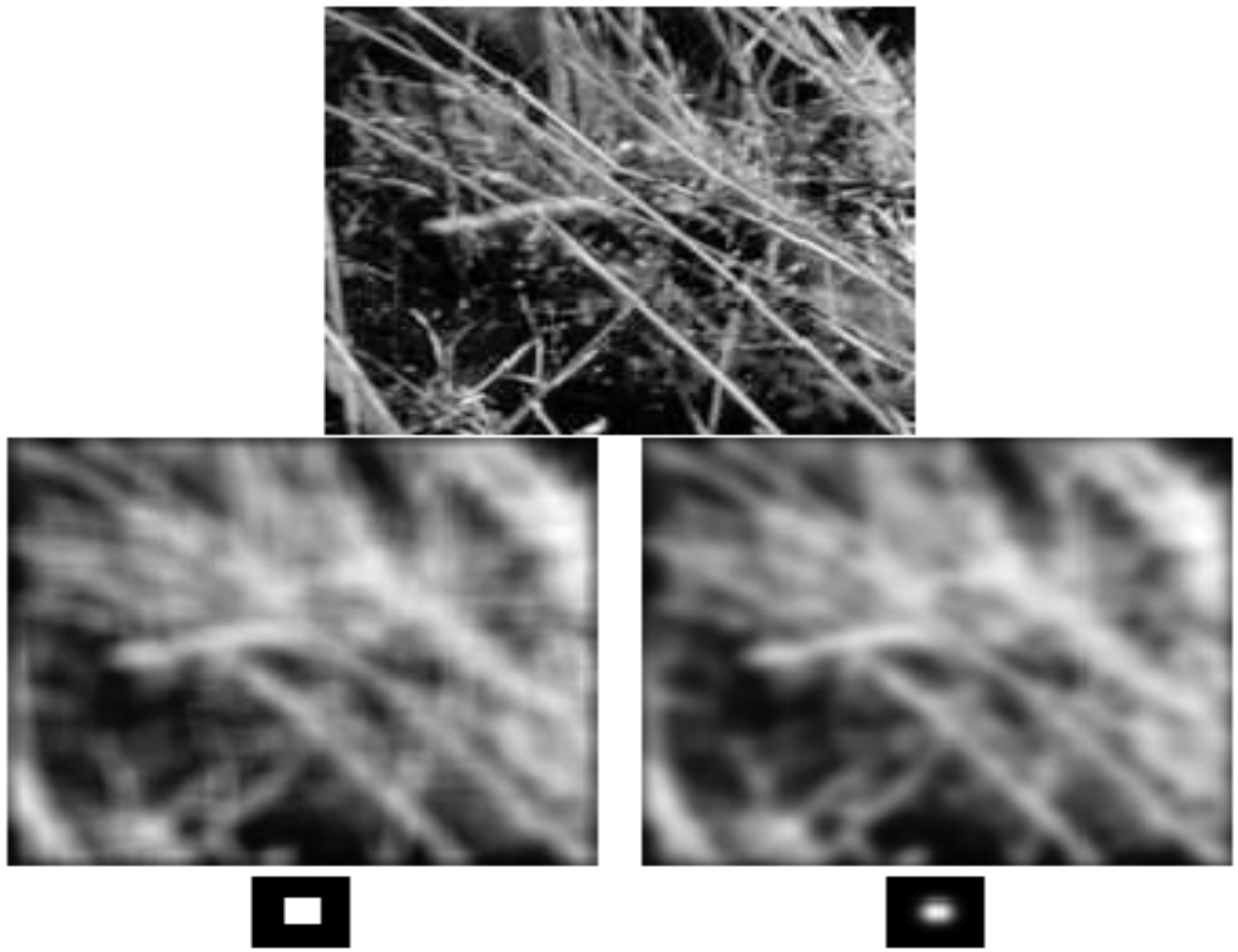
A Gaussian kernel gives less weight to pixels further from the center of the window

$$h(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{\sigma^2}}$$

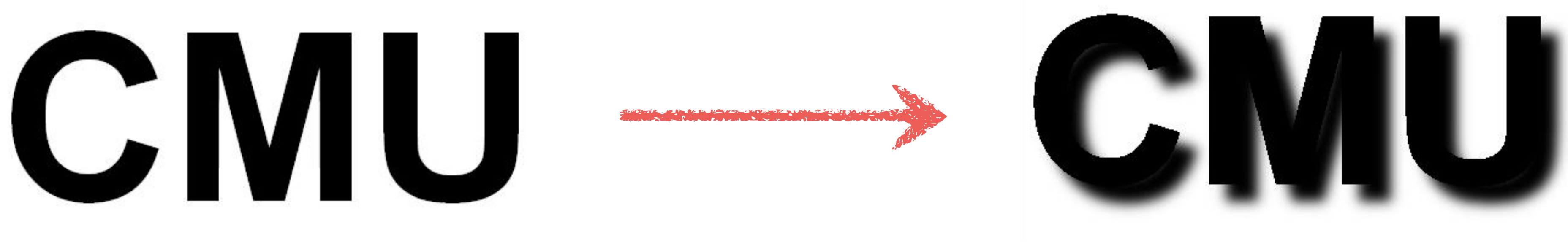
This kernel is an approximation of a Gaussian function



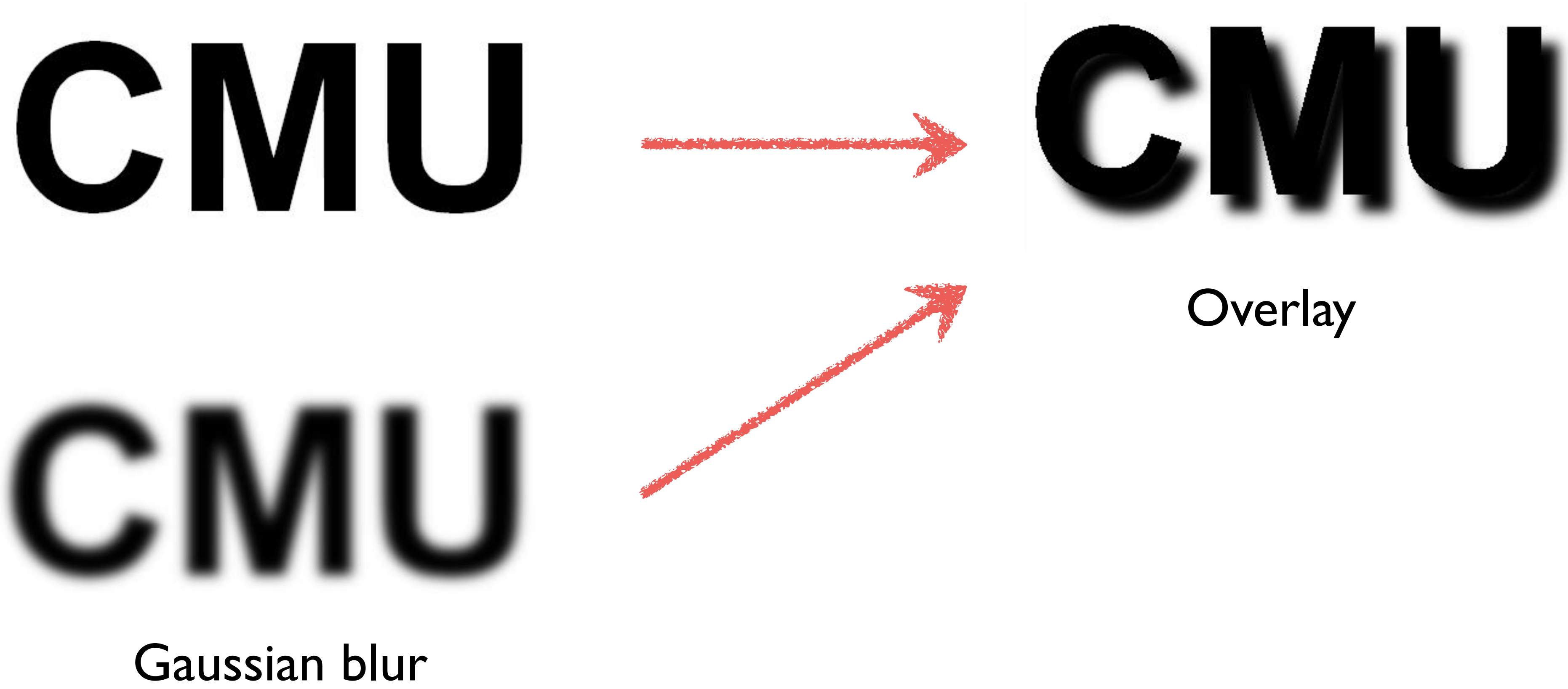
Gaussian filtering versus mean filtering



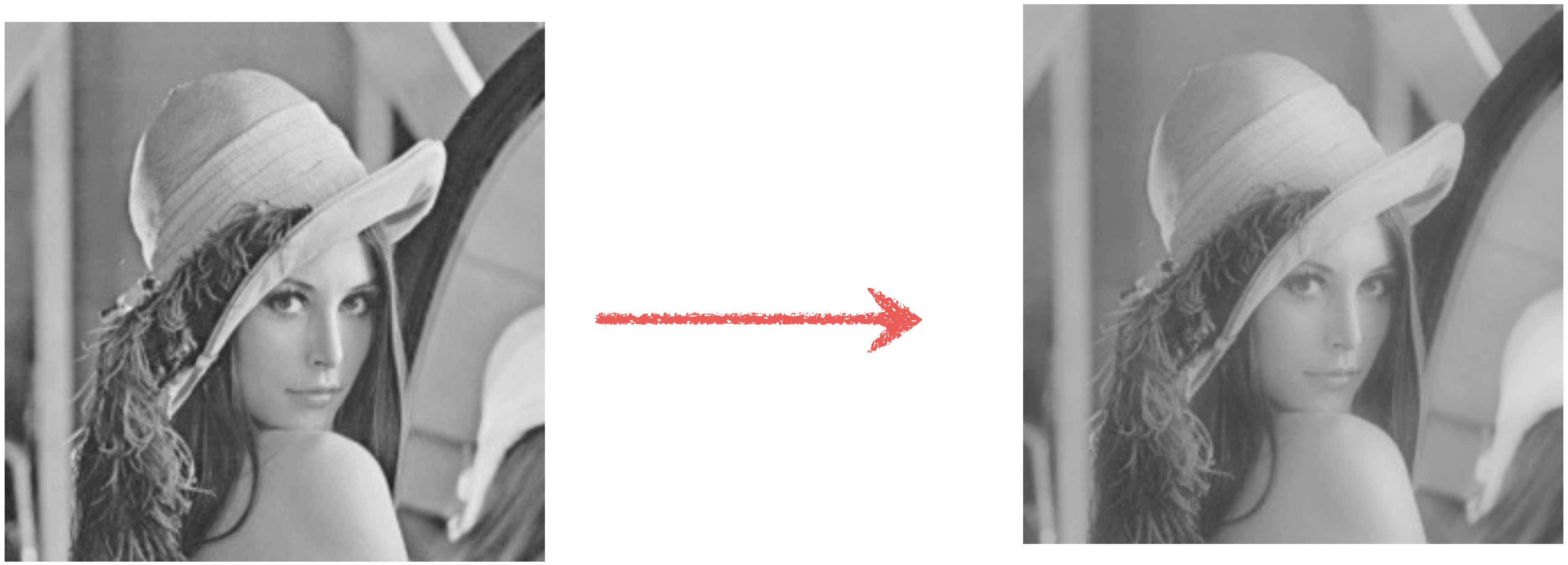
How would you create a shadow effect?



How would you create a shadow effect?



How would you create a soft focus effect?



How would you create a soft focus effect?

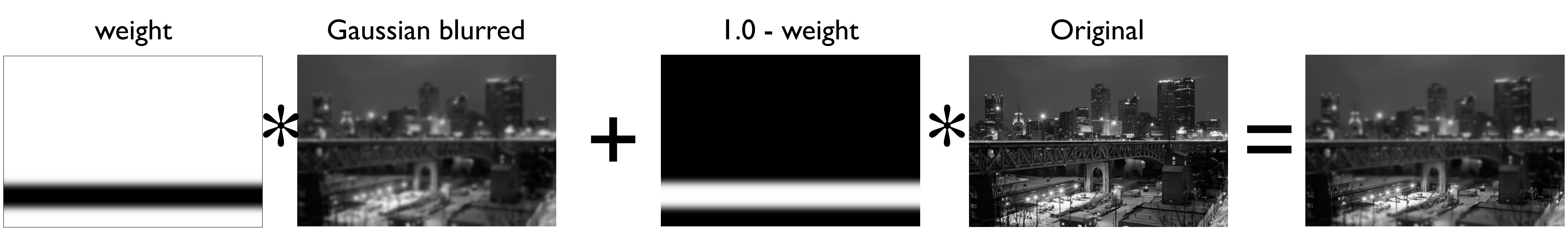
Tilt Shift Effect

http://www.flickr.com/photos/ender079/2704450659/

How would you create a (super low-budget) tilt-shift effect?

http://farm8.staticflickr.com/7061/6867631897_f8377709b9_z.jpg

How would you create a (super low-budget) tilt-shift effect?



Tell me everything wrong with this wannabe tilt-shift image

Fourier’s Claim

It all starts with this guy...

Who is this fellow?

Jean Baptiste Joseph Fourier (1768-1830)

Jean Baptiste Joseph Fourier (1768-1830)

What was his claim?

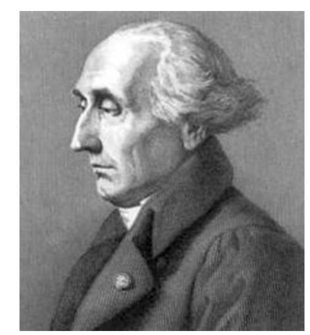
Jean Baptiste Joseph Fourier (1768-1830)

'Any univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies' (1807)

Jean Baptiste Joseph Fourier (1768-1830)

'Any univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies' (1807)

Laplace



Lagrange

Legendre

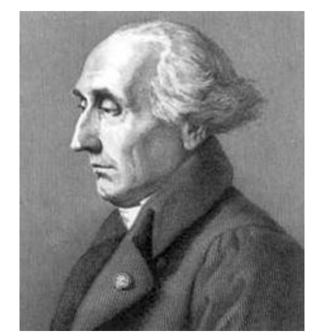
Poisson

What did these guys think of his claims?

Jean Baptiste Joseph Fourier (1768-1830)

'Any univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies'
(1807)

Laplace



Lagrange

Legendre

Poisson

...the manner in which the author arrives at these equations is not exempt of difficulties and...his analysis to integrate them still leaves something to be desired on the score of generality and even rigour.

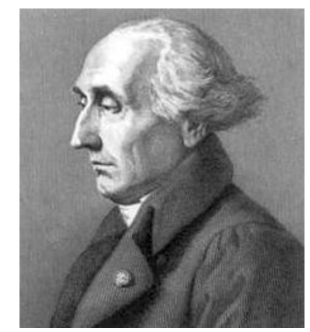
- Laplace

Not translated to English until 1878!

Jean Baptiste Joseph Fourier (1768-1830)

'Any univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies'
(1807)

Laplace



Lagrange

Legendre

Poisson

Why is he so angry?

1820 watercolor caricatures of French mathematicians Adrien-Marie Legendre (left) and Joseph Fourier (right) by French artist Julien-Leopold Boilly, watercolor portrait numbers 29 and 30 of Album de 73 Portraits-Charge Aquarelle's des Membres de l'Institute.

http://en.wikipedia.org/wiki/Joseph_Fourier

Basic building block

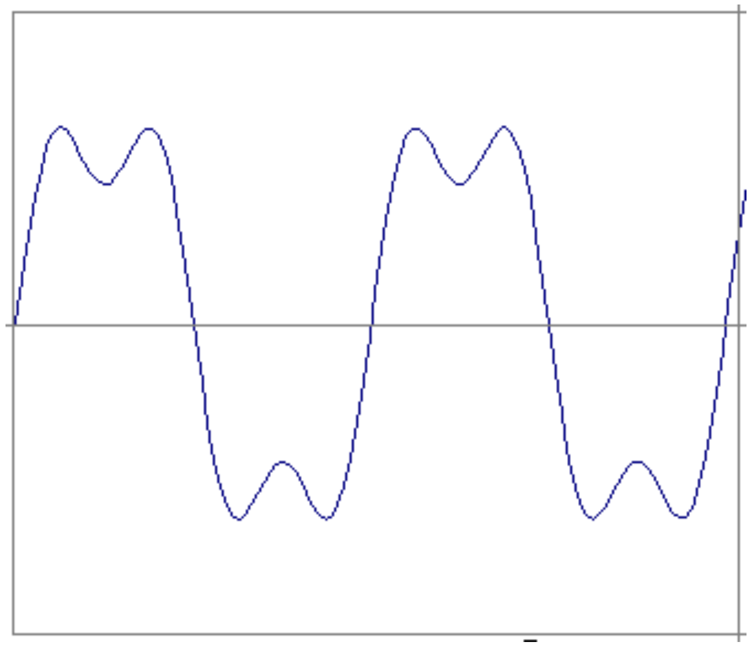
$$A \sin(\omega x + \phi)$$

amplitude — sinusoid — angular frequency — variable — phase

CLAIM:

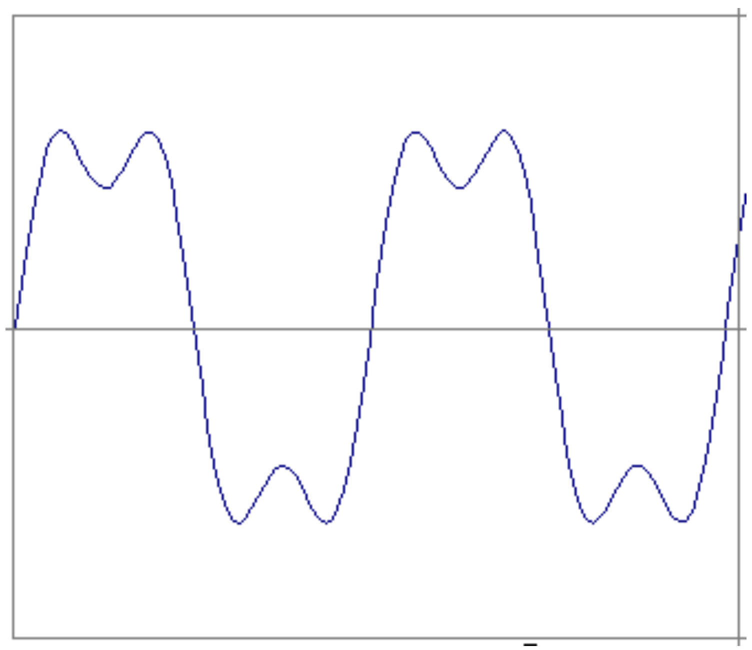
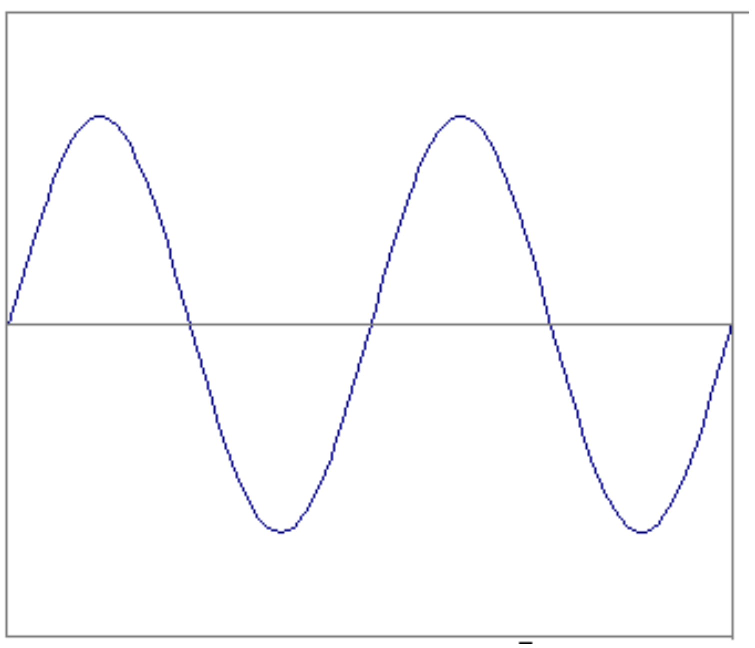
Add enough of them to get <u>any</u> signal you want!

How would you generate this function?

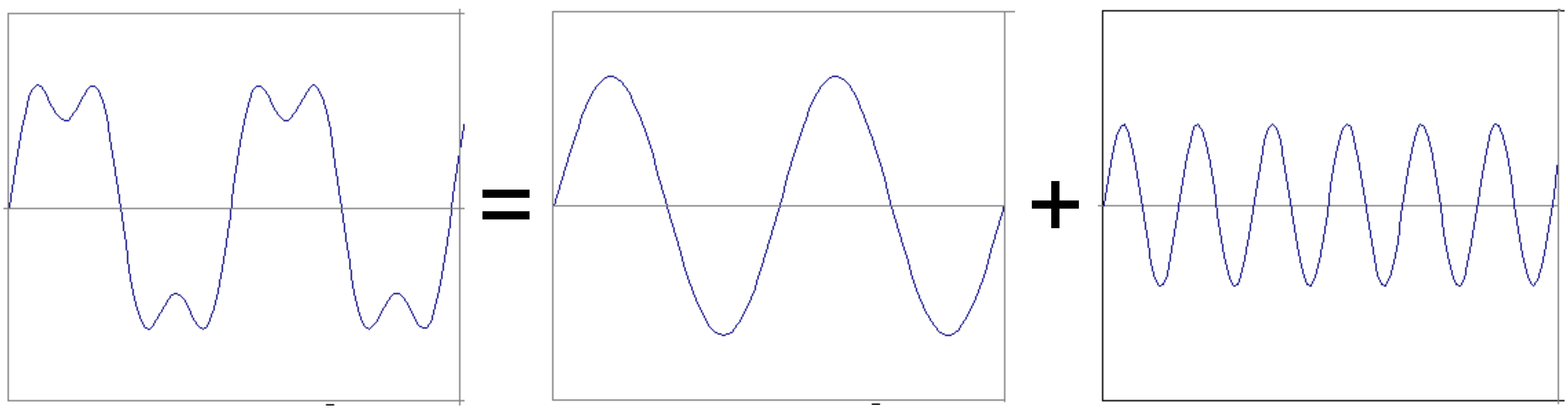


= ? + ?

How would you generate this function?

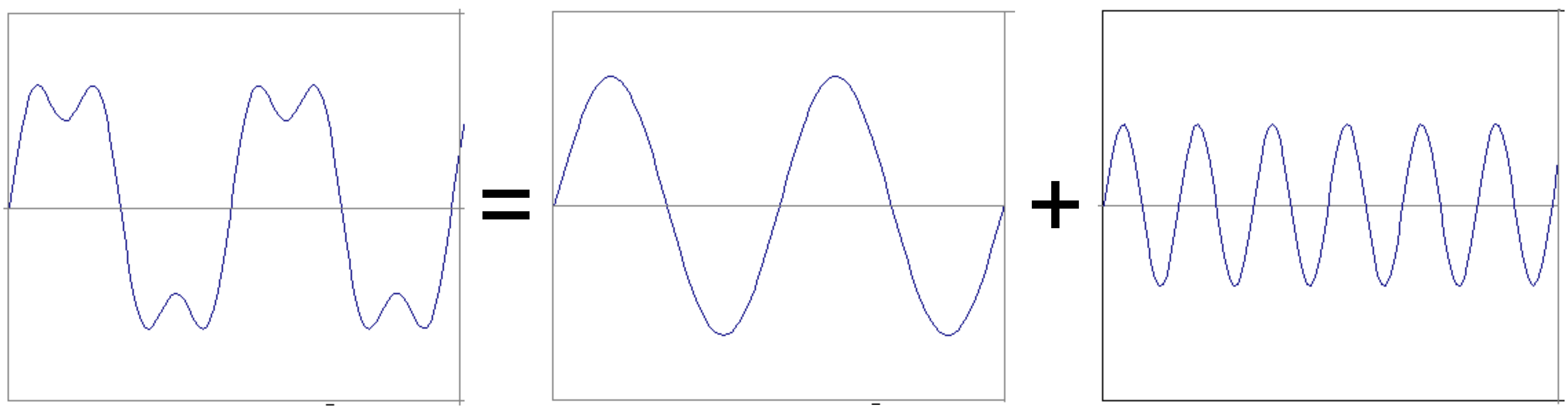
 =  + ?

$\sin(2\pi x)$



$$\sin(2\pi x) \qquad \frac{1}{3}\sin(2\pi 3x)$$

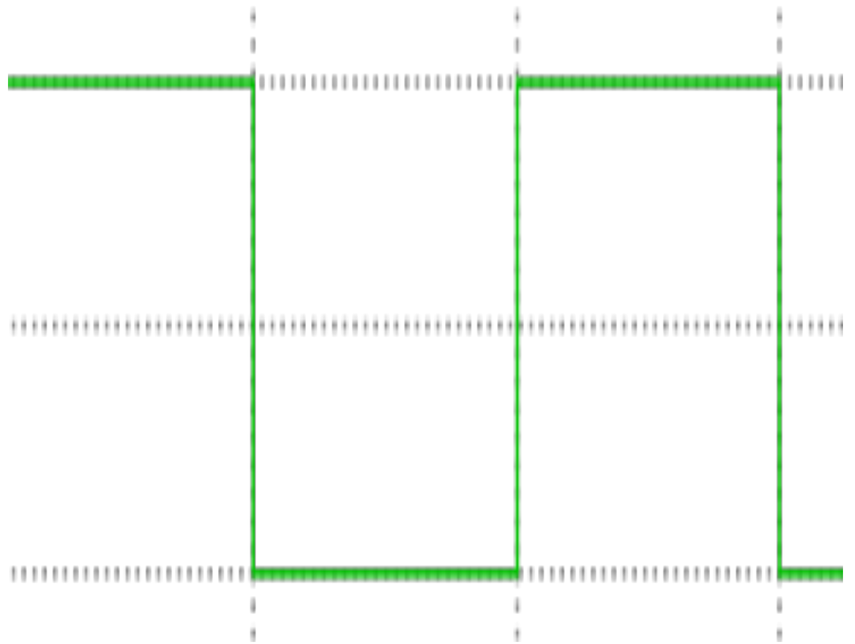
$$f(x) = \sin(2\pi x) + \frac{1}{3}\sin(2\pi 3x)$$



$\sin(2\pi x)$

$\frac{1}{3}\sin(2\pi 3x)$

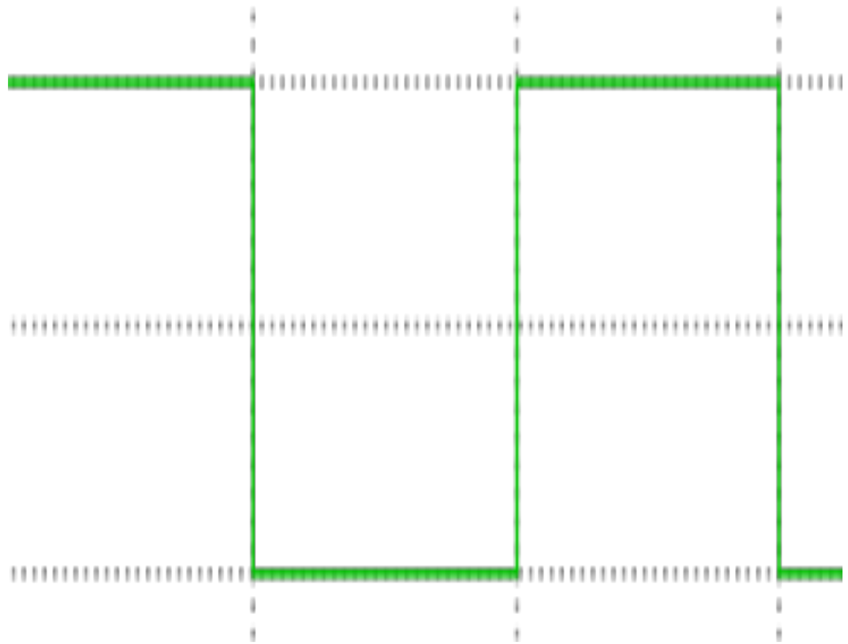
How would you generate this function?



= ? + ?

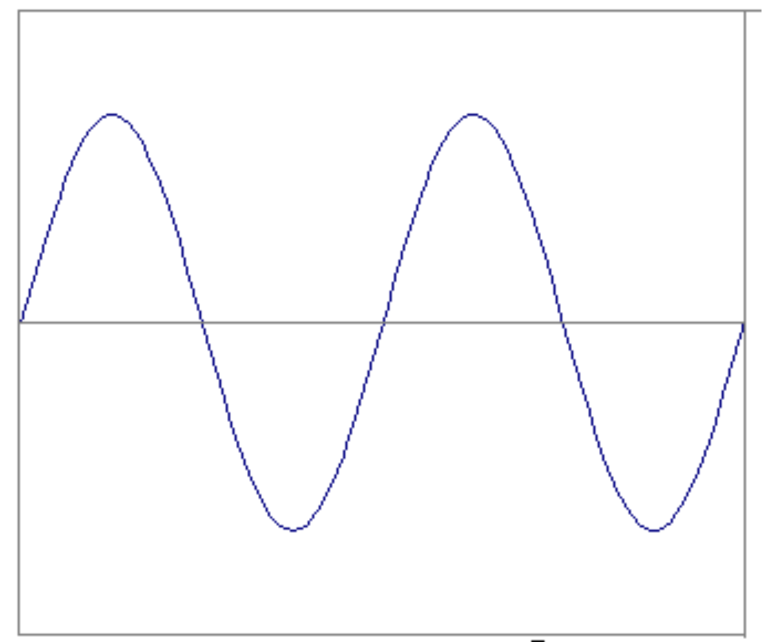
Square wave

How would you generate this function?

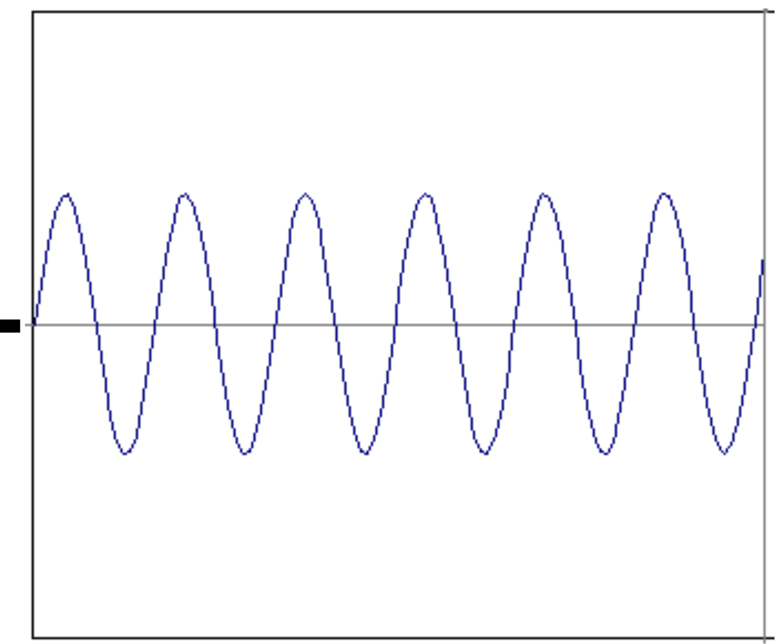


Square wave

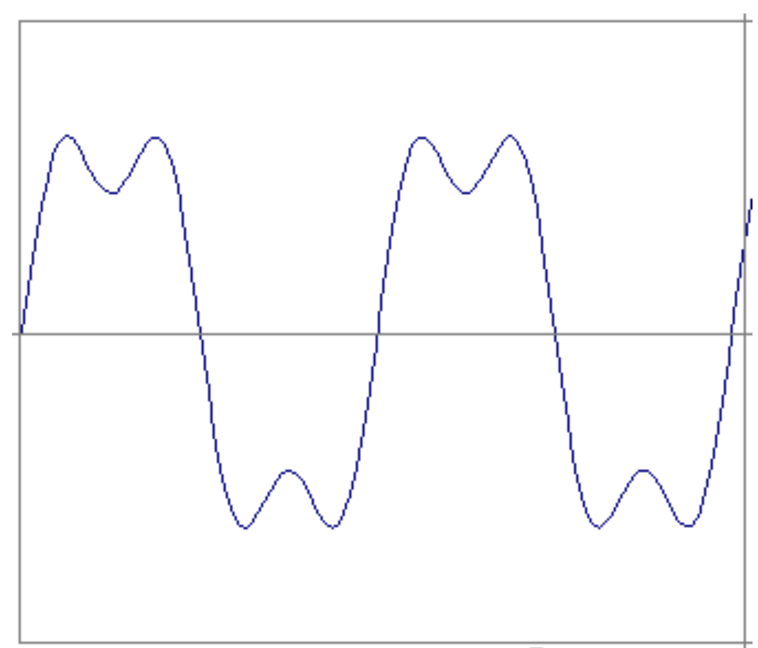
≈

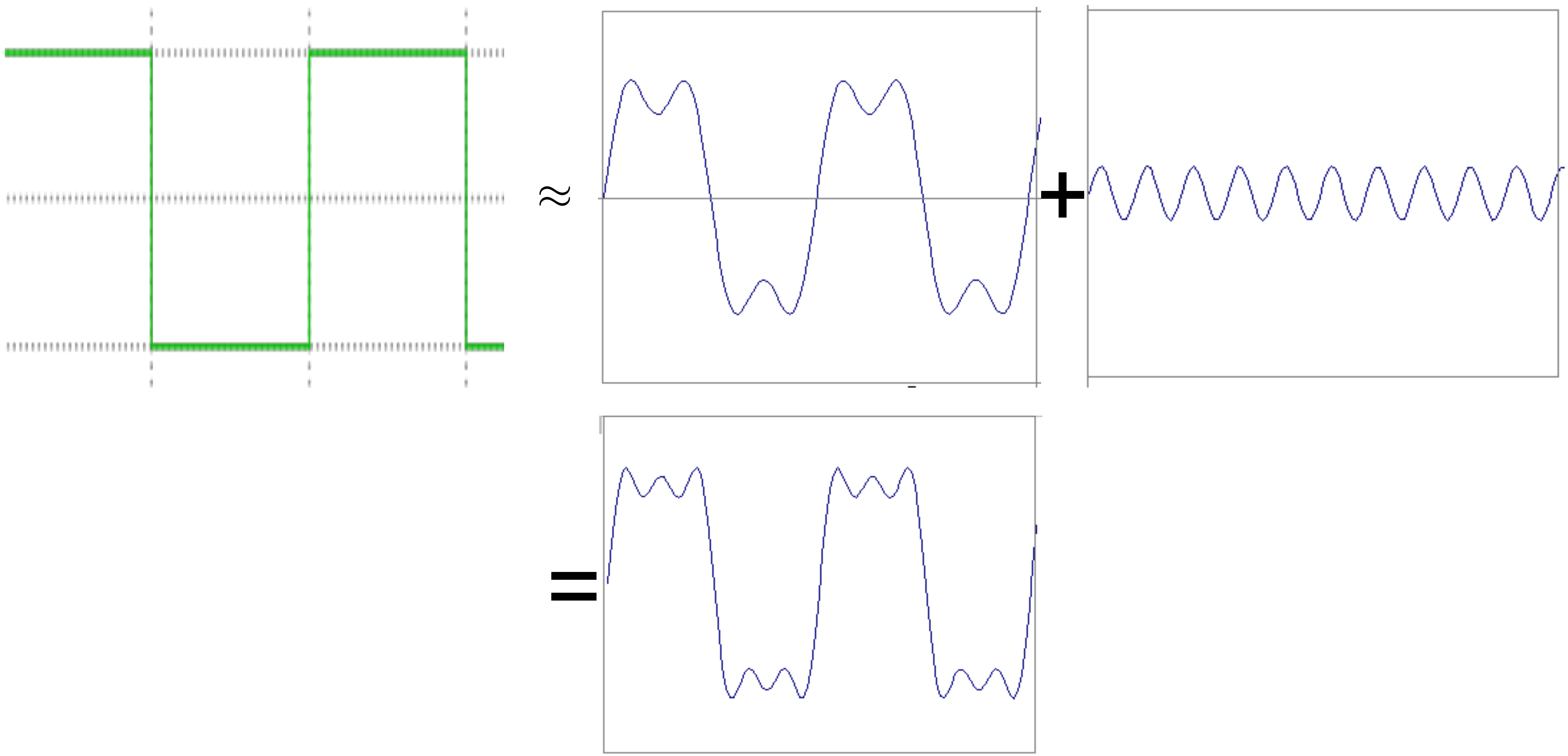


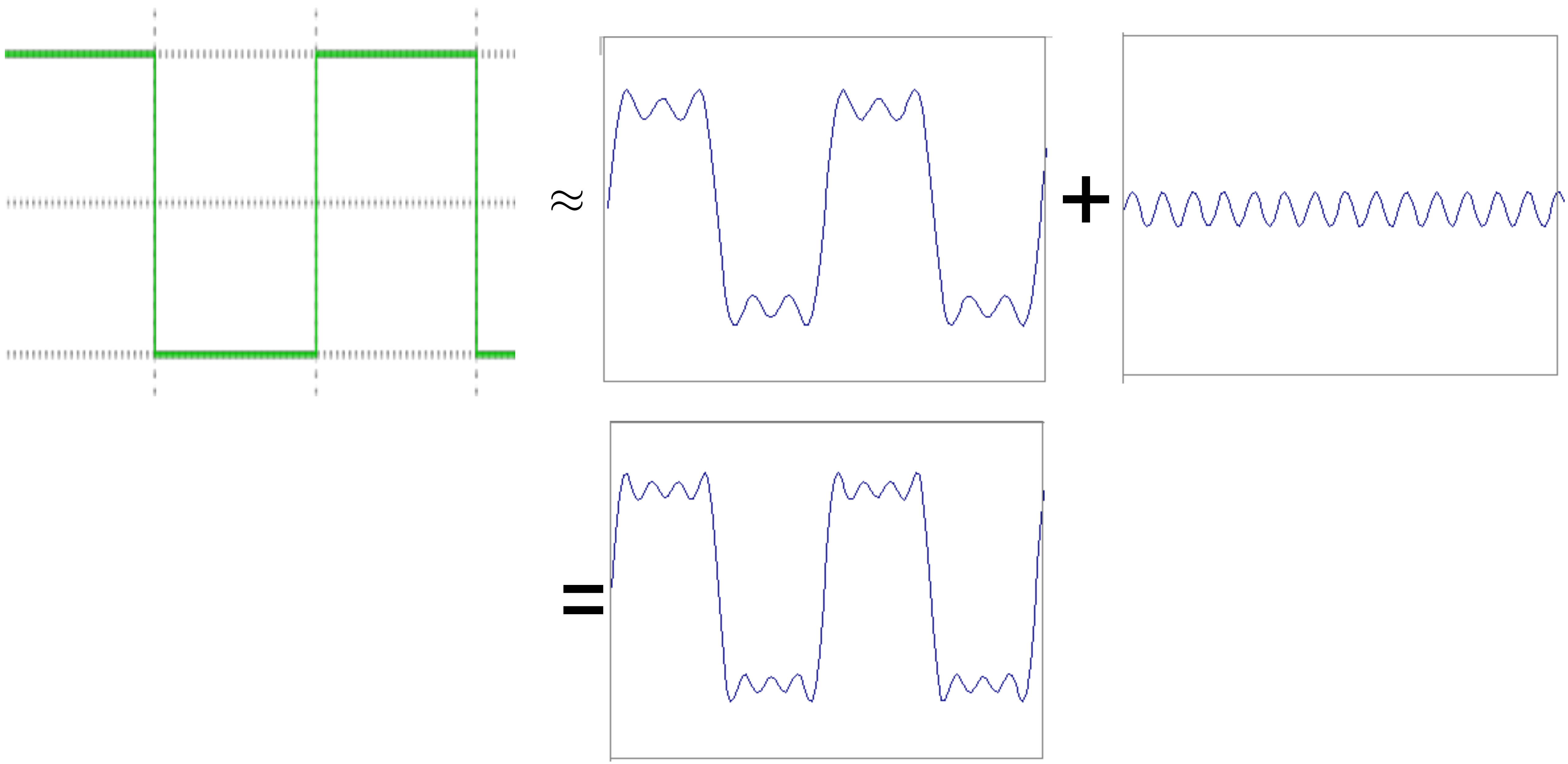
\+

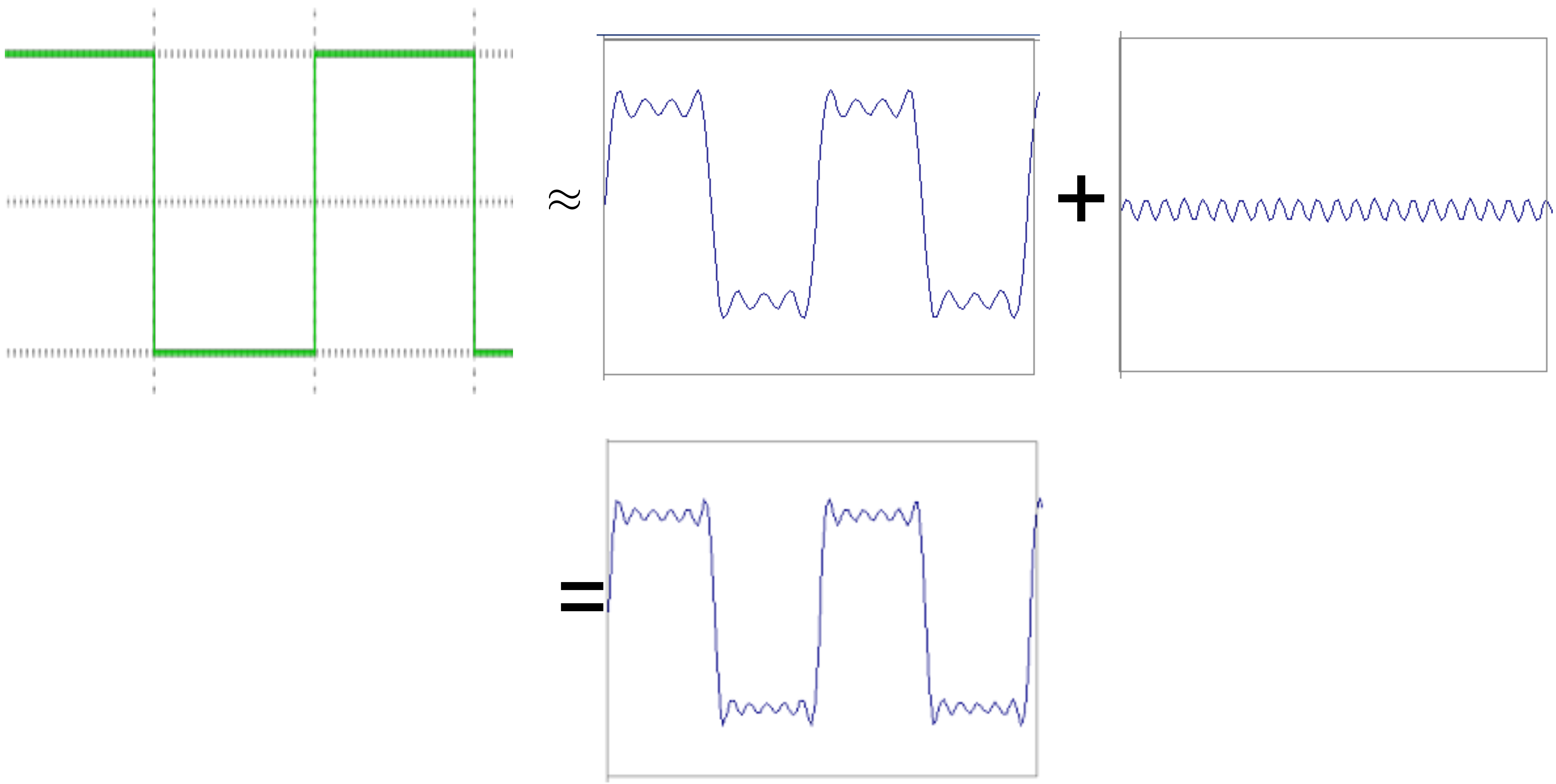


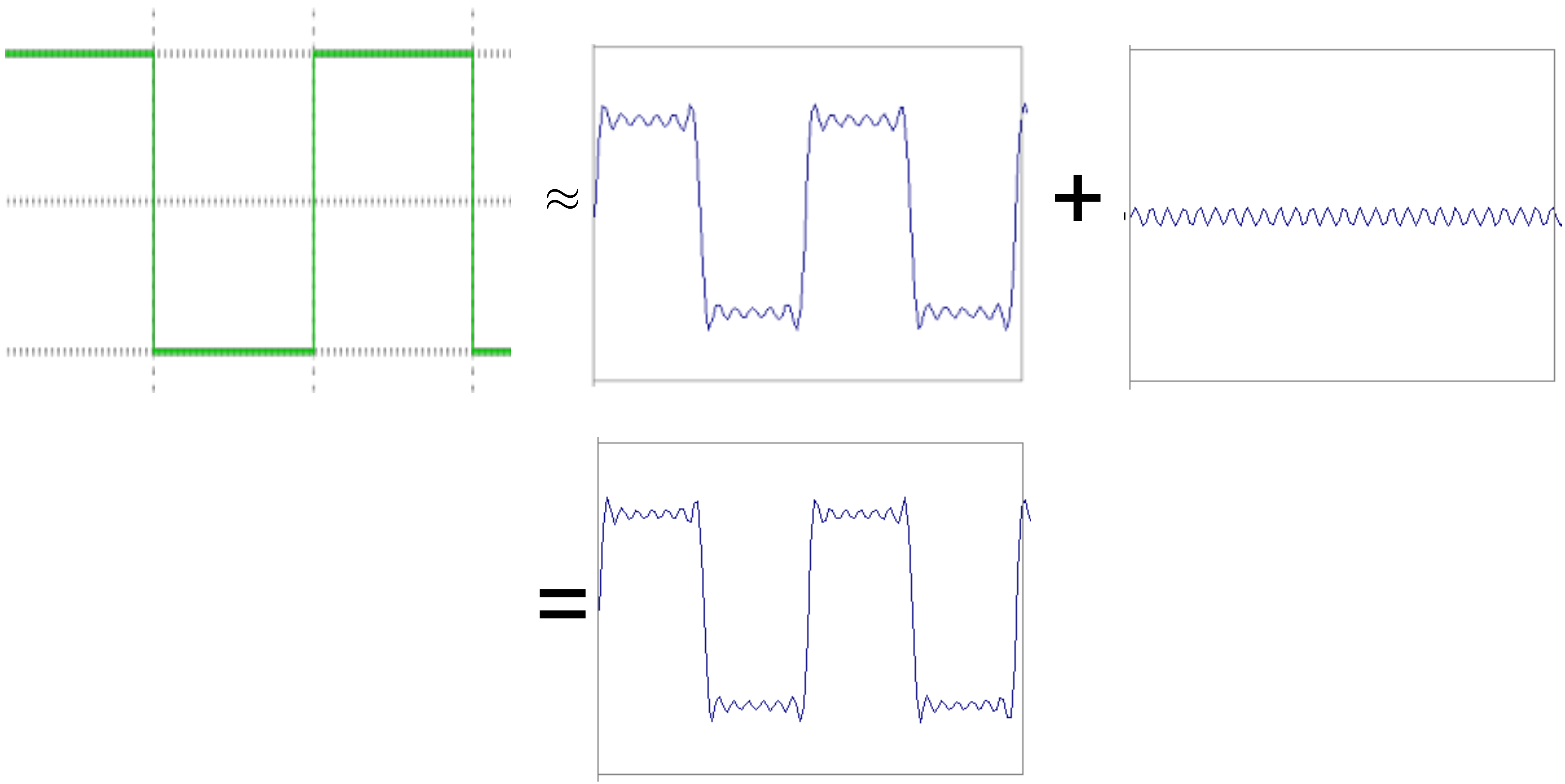
=



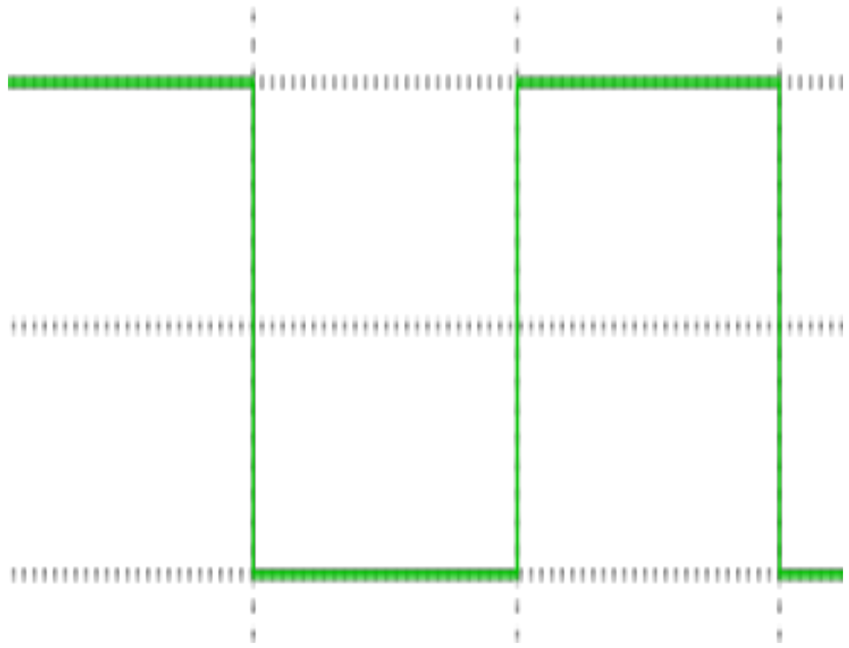




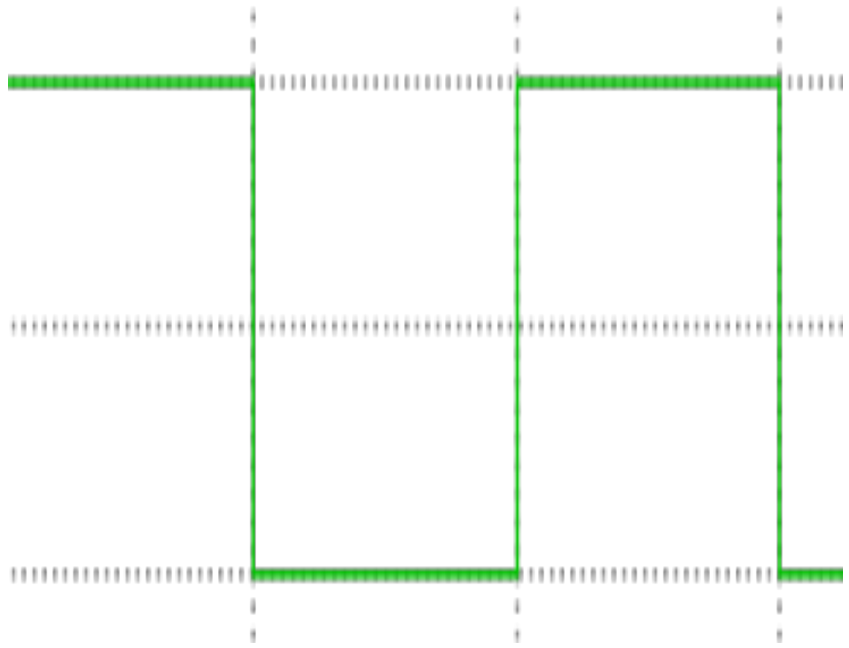




How would you express this mathematically?



$$= \sum_{k=1}^{\infty}$$

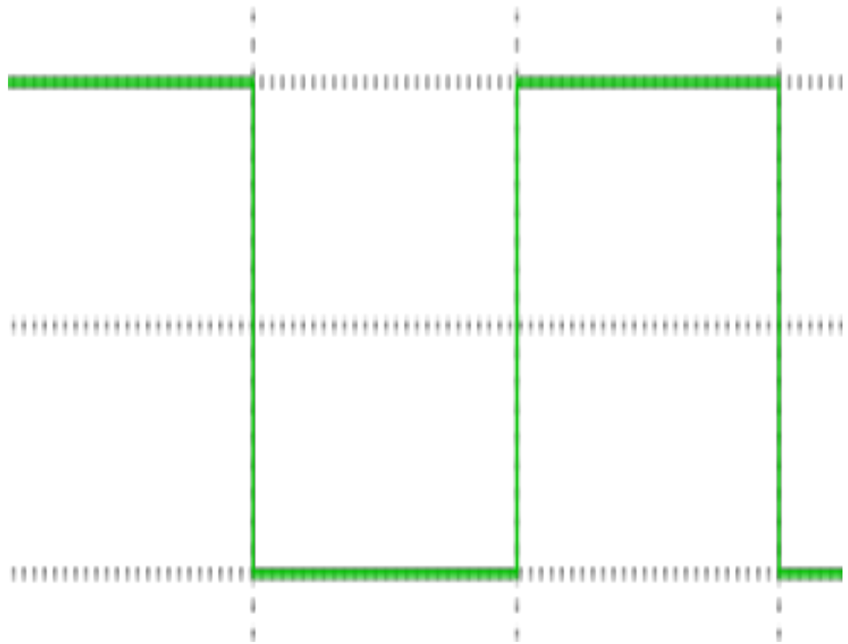


=

$$A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kx)$$

A square wave is an infinite sum of sine waves

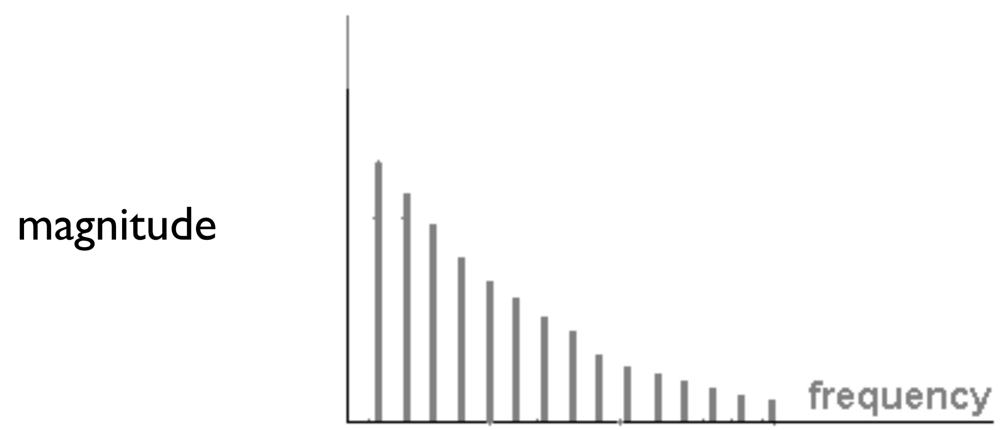
How would could you visualize this in the frequency domain?



=

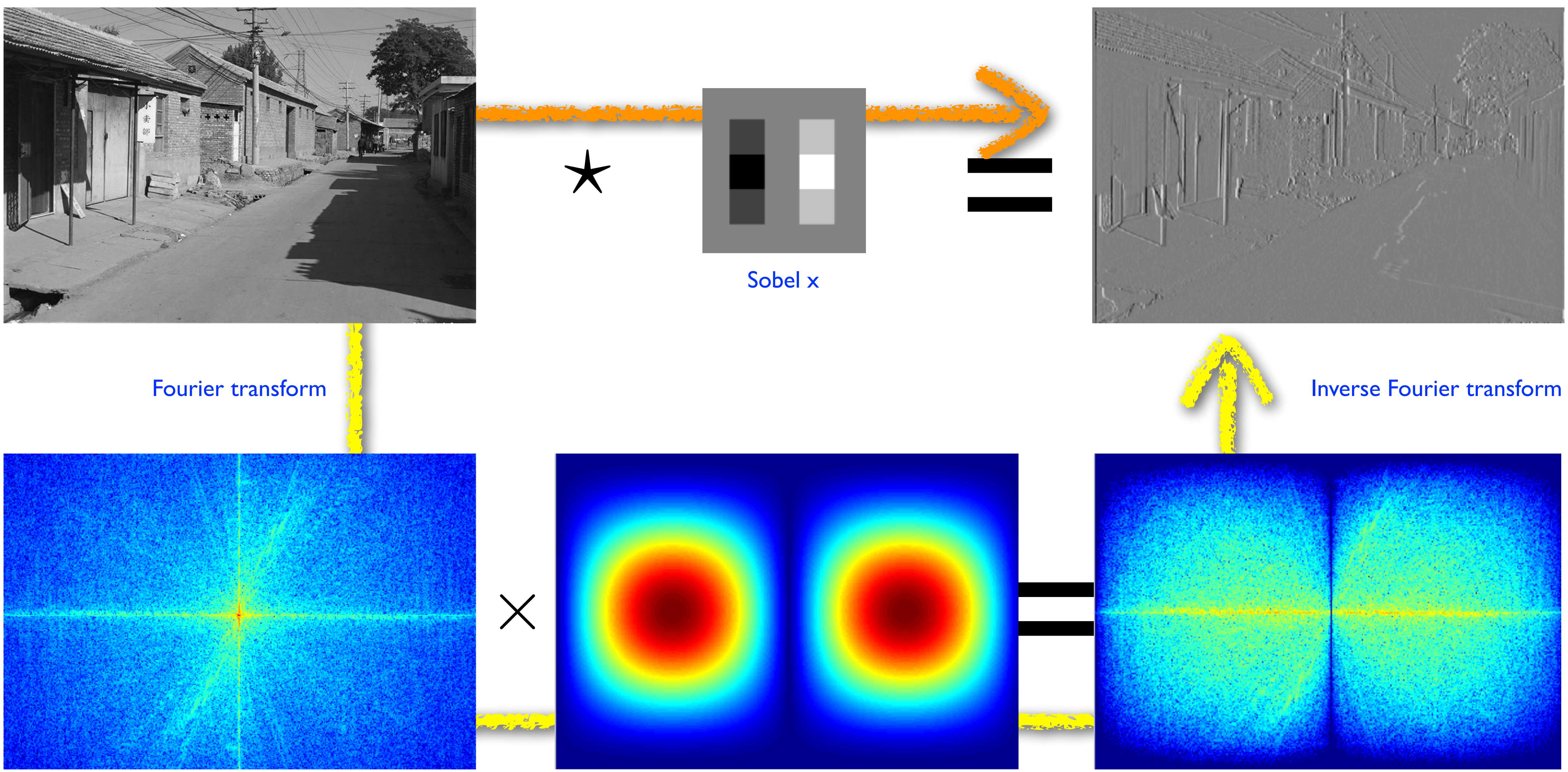
$$A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kx)$$

A square wave is an infinite sum of sine waves

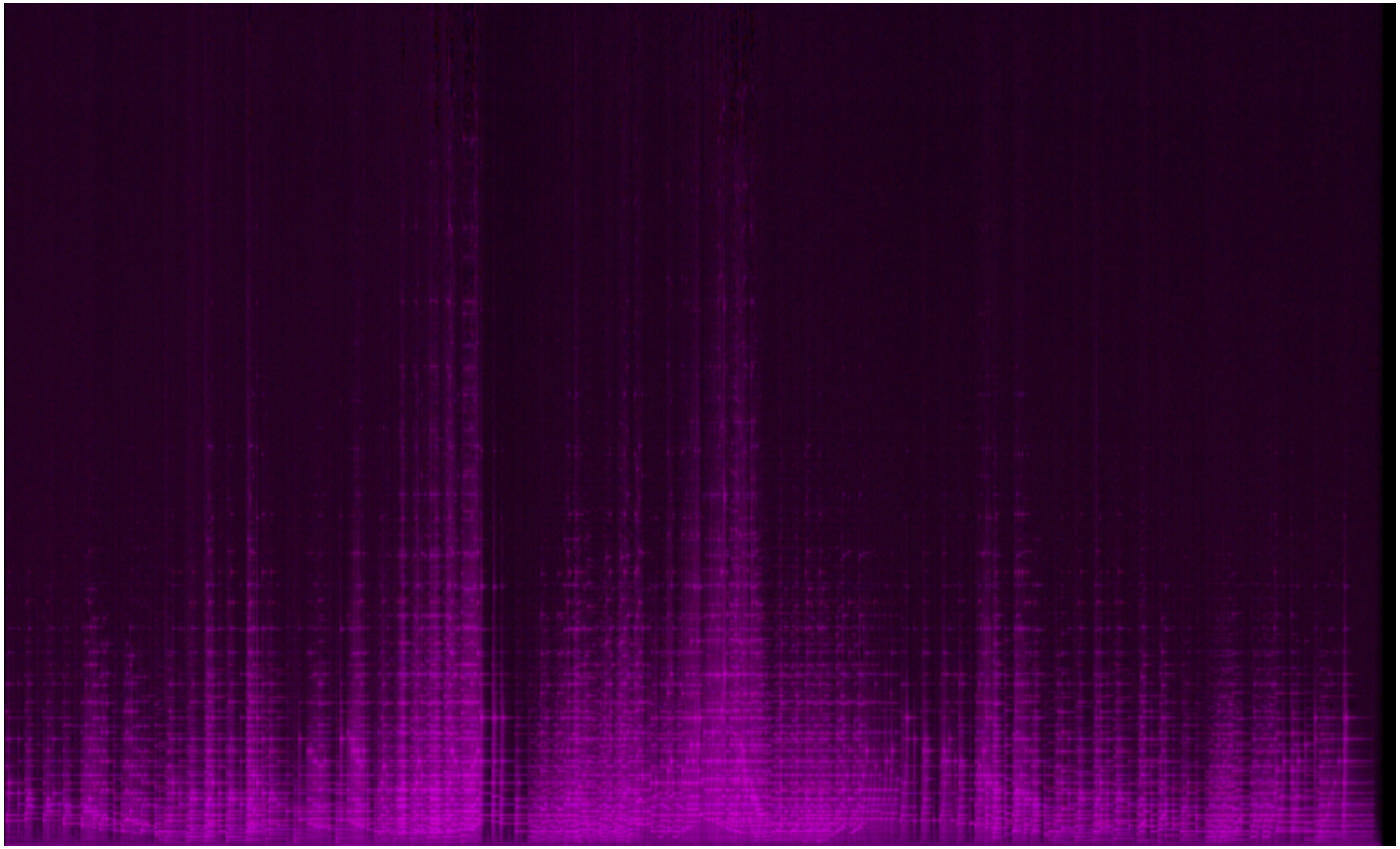


Why does this matter?

Spatial domain filtering

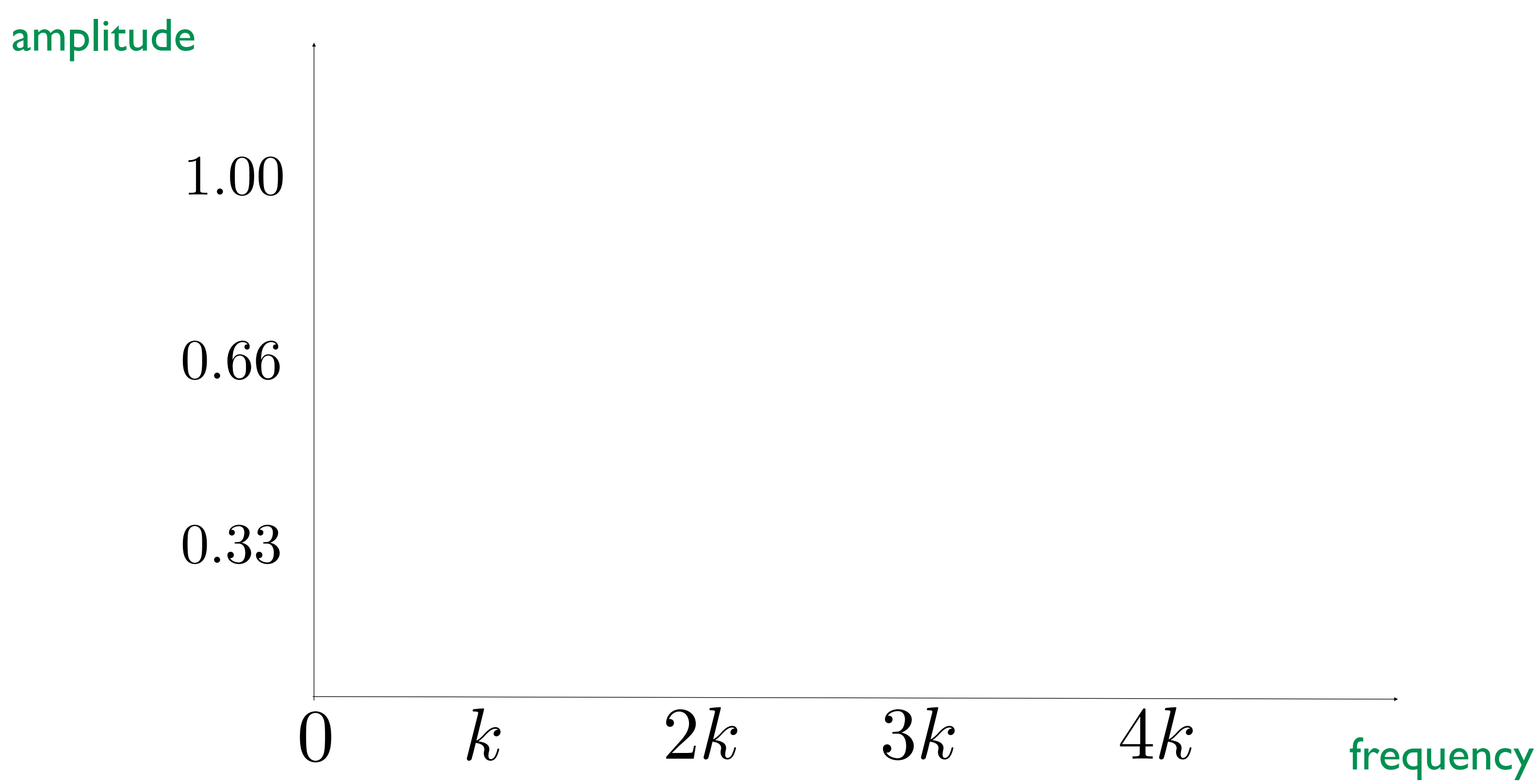


Frequency domain filtering



Frequency Spectrum

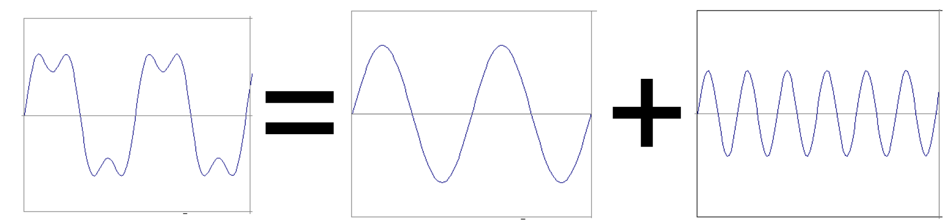
Visualizing the frequency spectrum

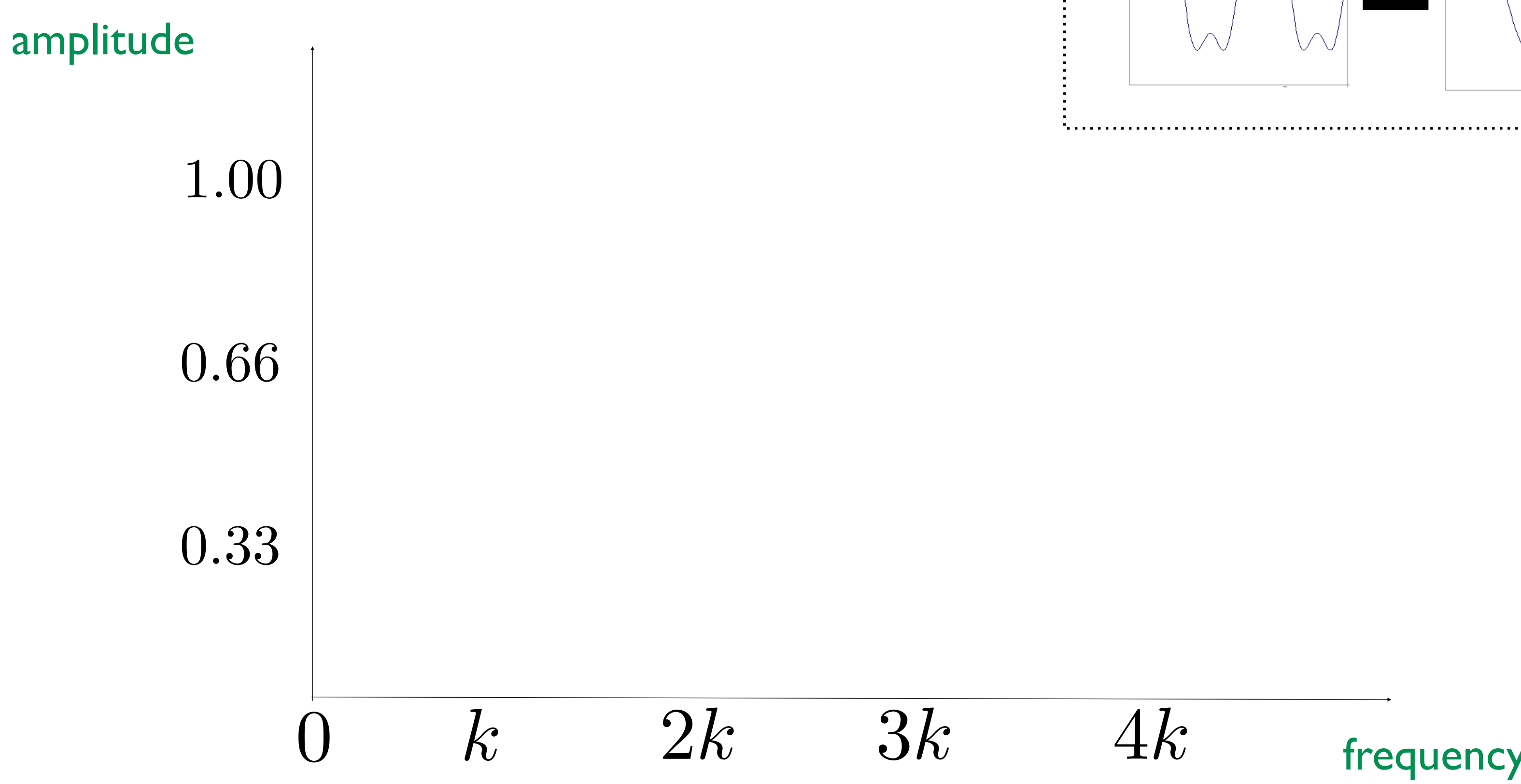


Visualizing the frequency spectrum

Recall the temporal domain visualization

$$f(x) = \sin(2\pi kx) + \frac{1}{3}\sin(2\pi 3kx)$$

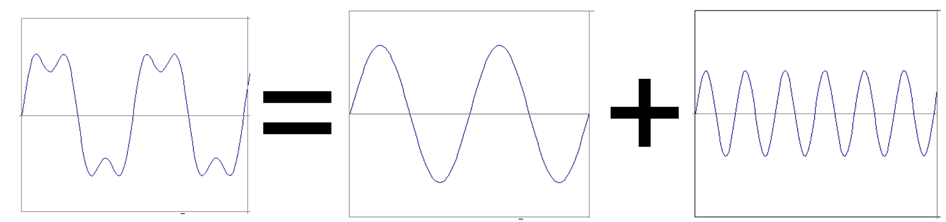


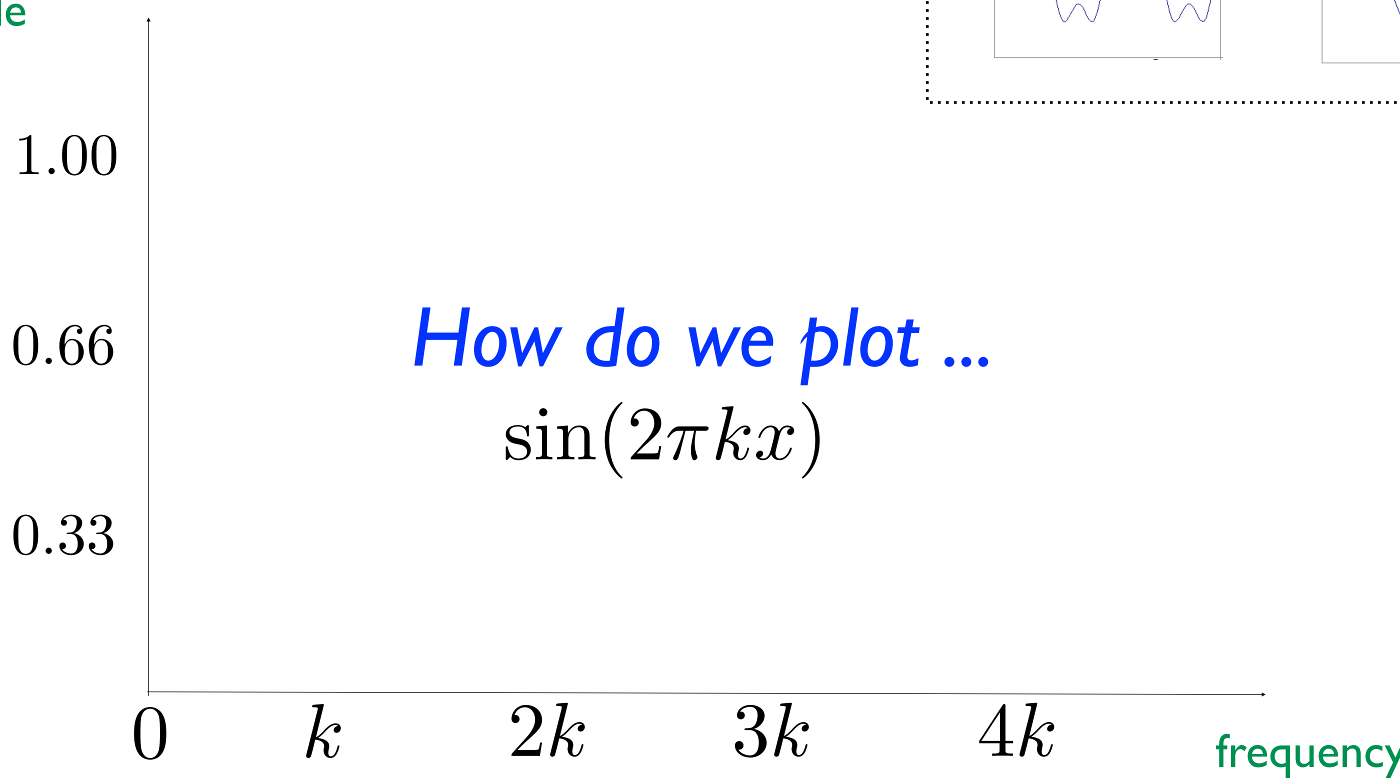


Visualizing the frequency spectrum

Recall the temporal domain visualization

$$f(x) = \sin(2\pi kx) + \frac{1}{3}\sin(2\pi 3kx)$$





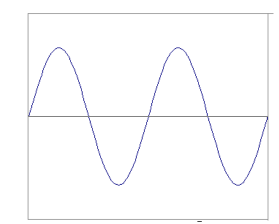
How do we plot ...

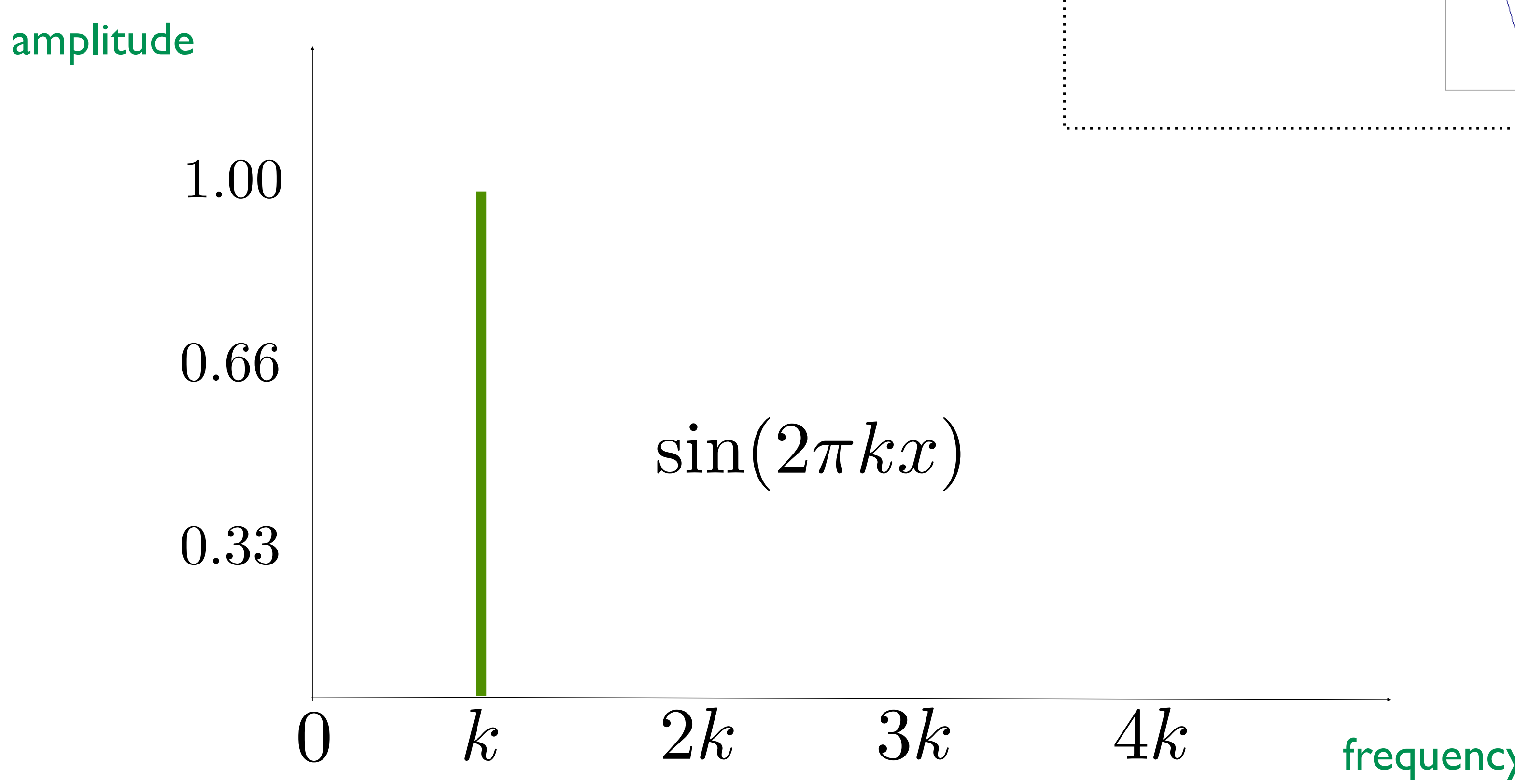
$$\sin(2\pi kx)$$

Visualizing the frequency spectrum

Recall the temporal domain visualization

$$f(x) = \sin(2\pi kx) + \frac{1}{3}\sin(2\pi 3kx)$$

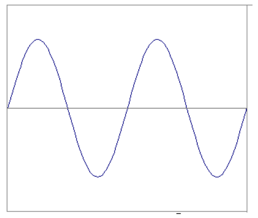


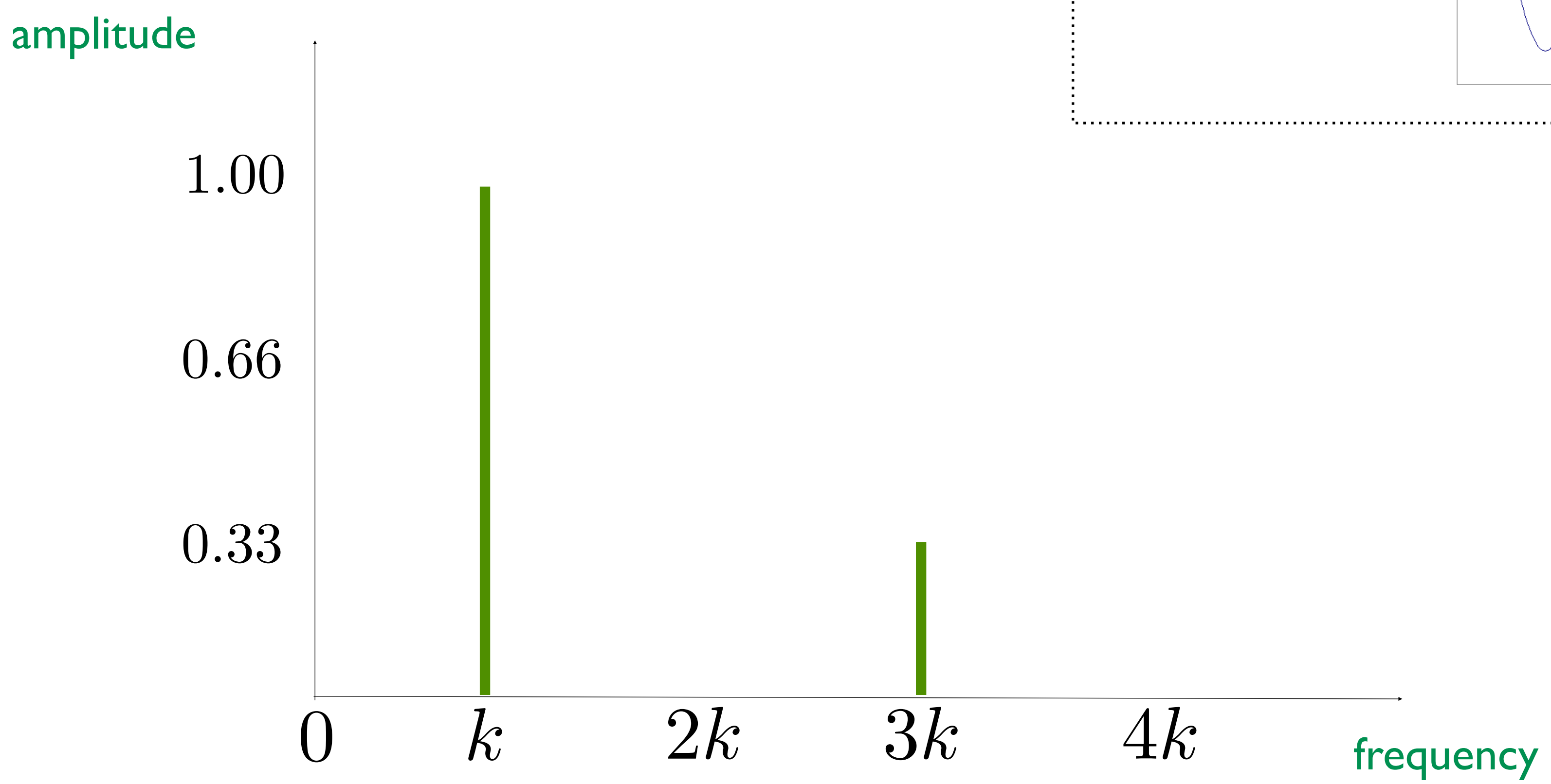


Visualizing the frequency spectrum

Recall the temporal domain visualization

$$f(x) = \sin(2\pi kx) + \frac{1}{3}\sin(2\pi 3kx)$$

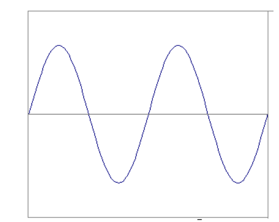


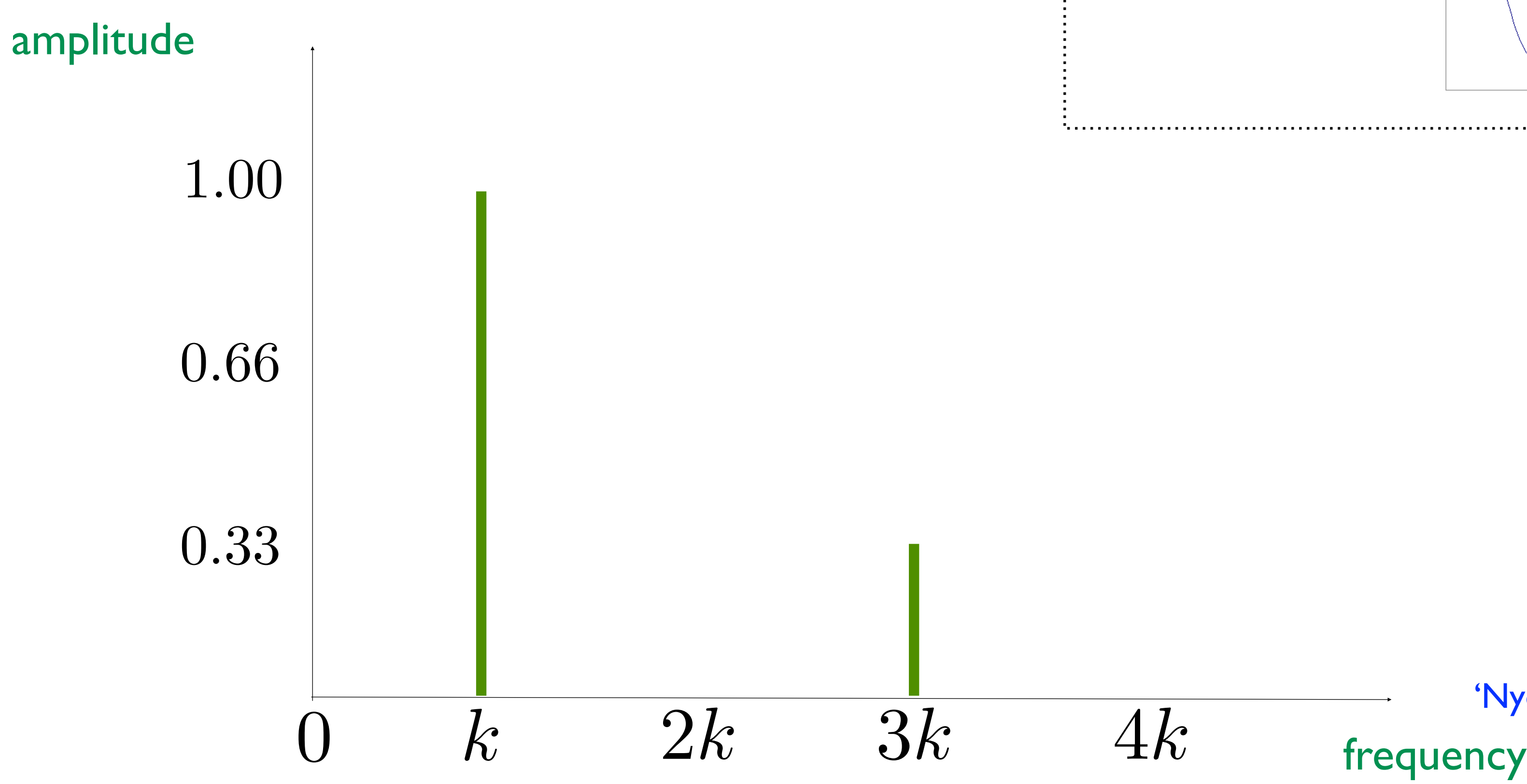


Visualizing the frequency spectrum

Recall the temporal domain visualization

$$f(x) = \sin(2\pi kx) + \frac{1}{3}\sin(2\pi 3kx)$$





not visualizing the symmetric negative part

'Nyquist' frequency

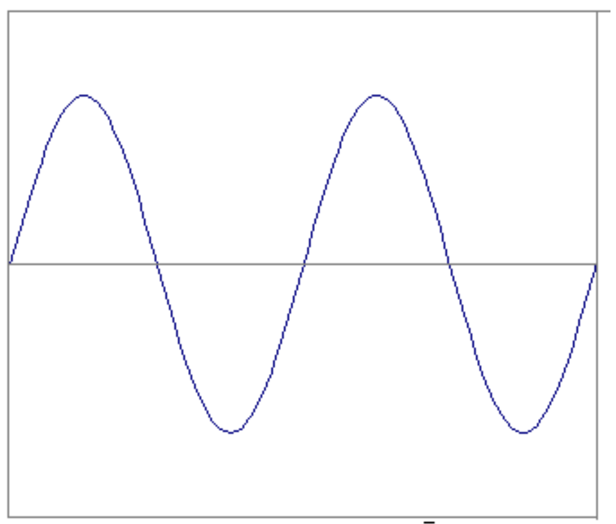
Signal average
(zero for a sine wave with no offset)

Need to understand this to understand the 2D version...

Spatial domain visualization

Frequency domain visualization

1D



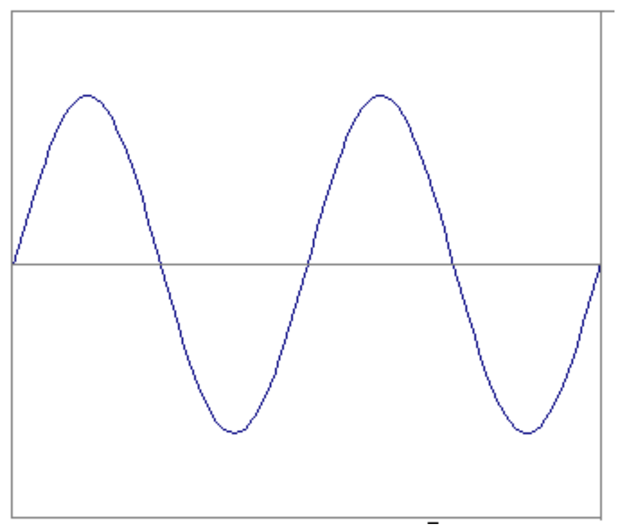
2D

?

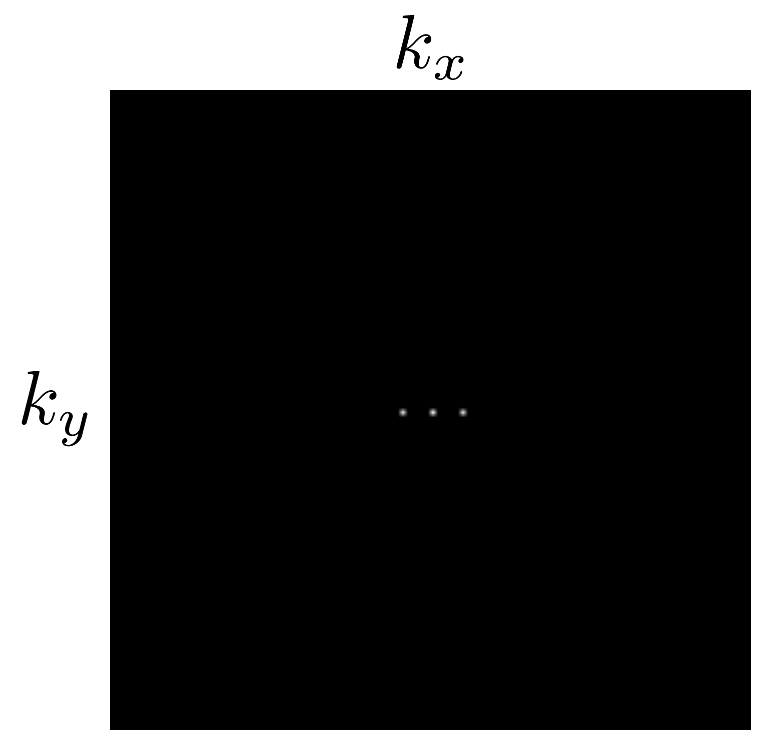
Spatial domain visualization

Frequency domain visualization

1D

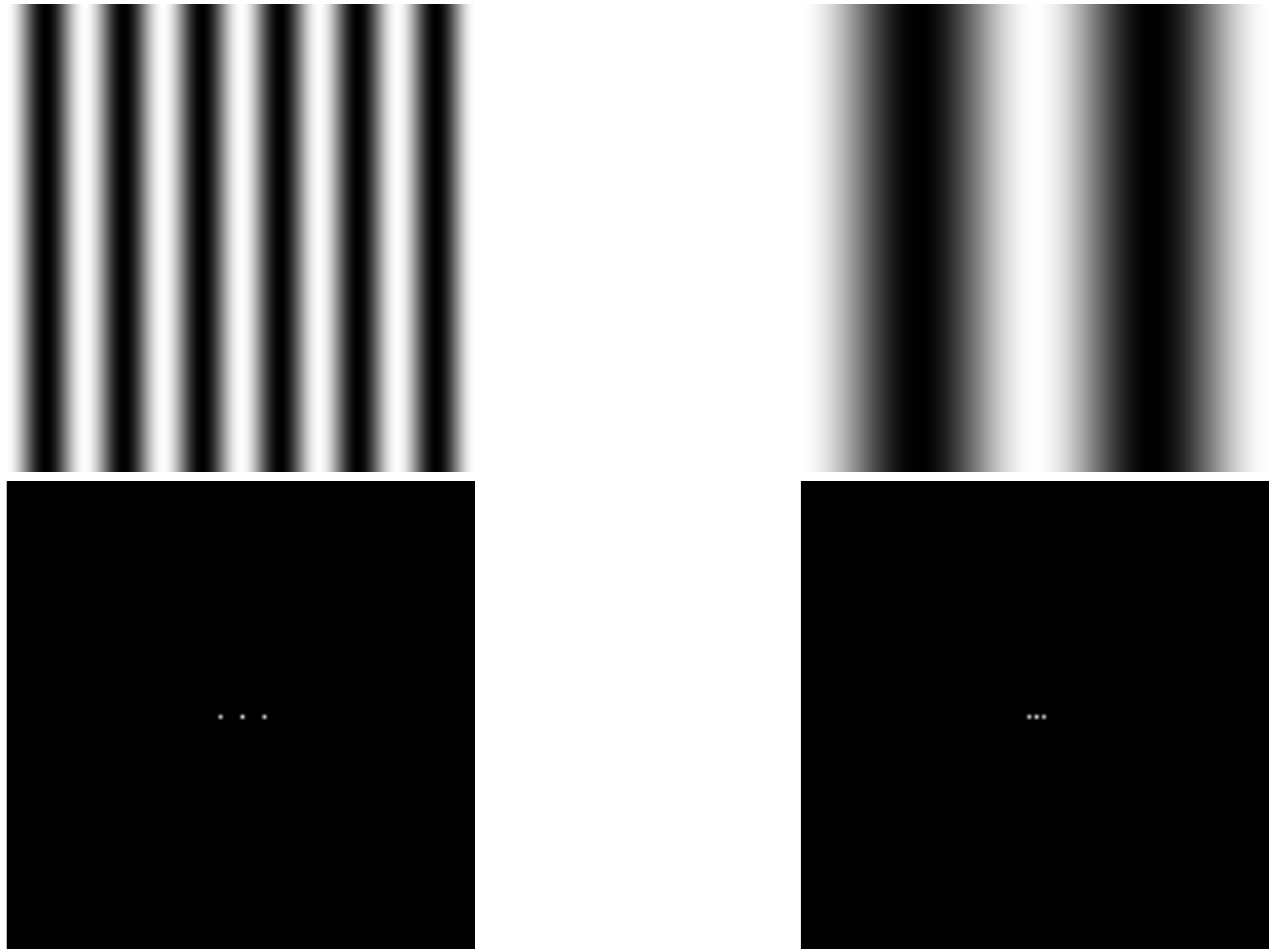


2D



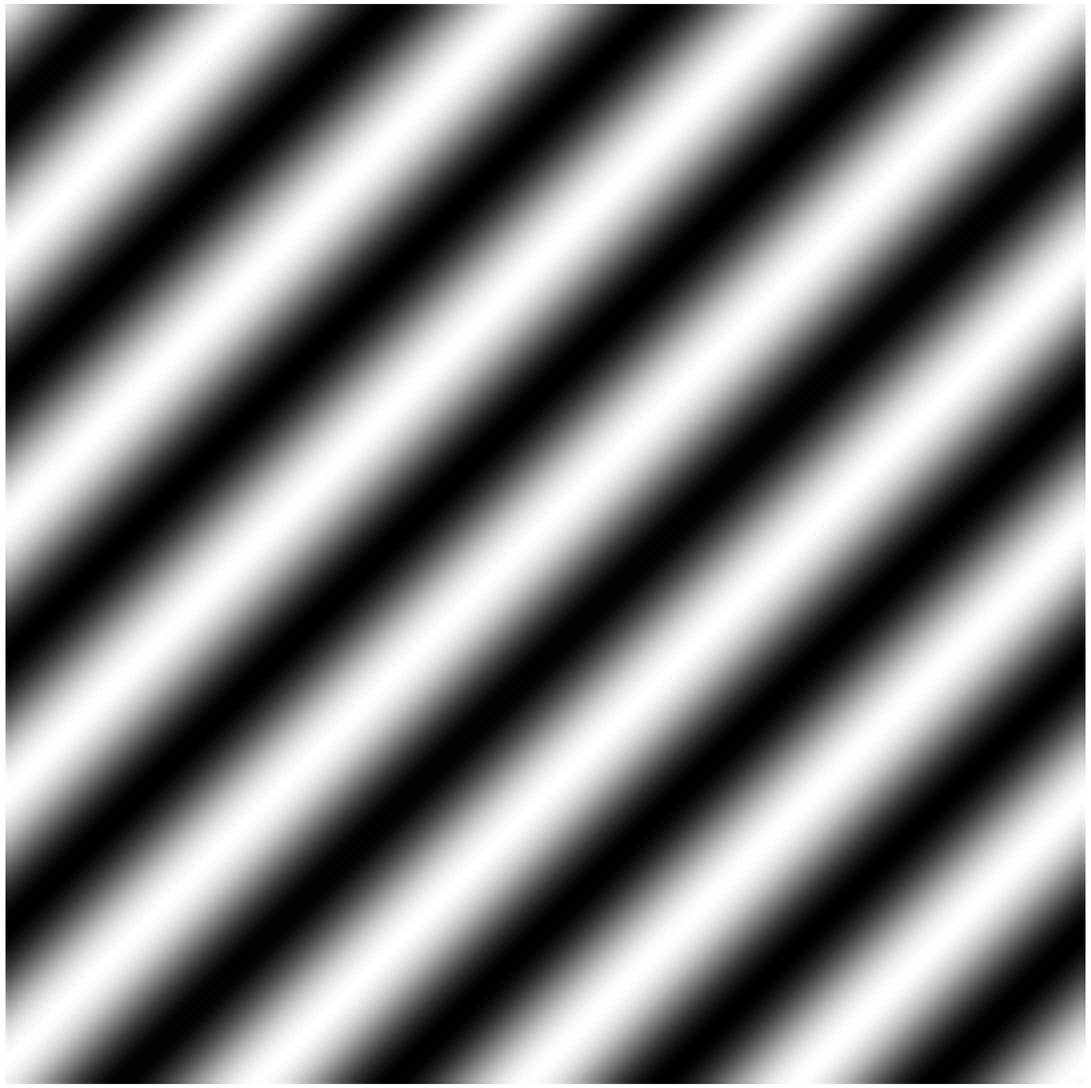
Intensity in the spatial domain

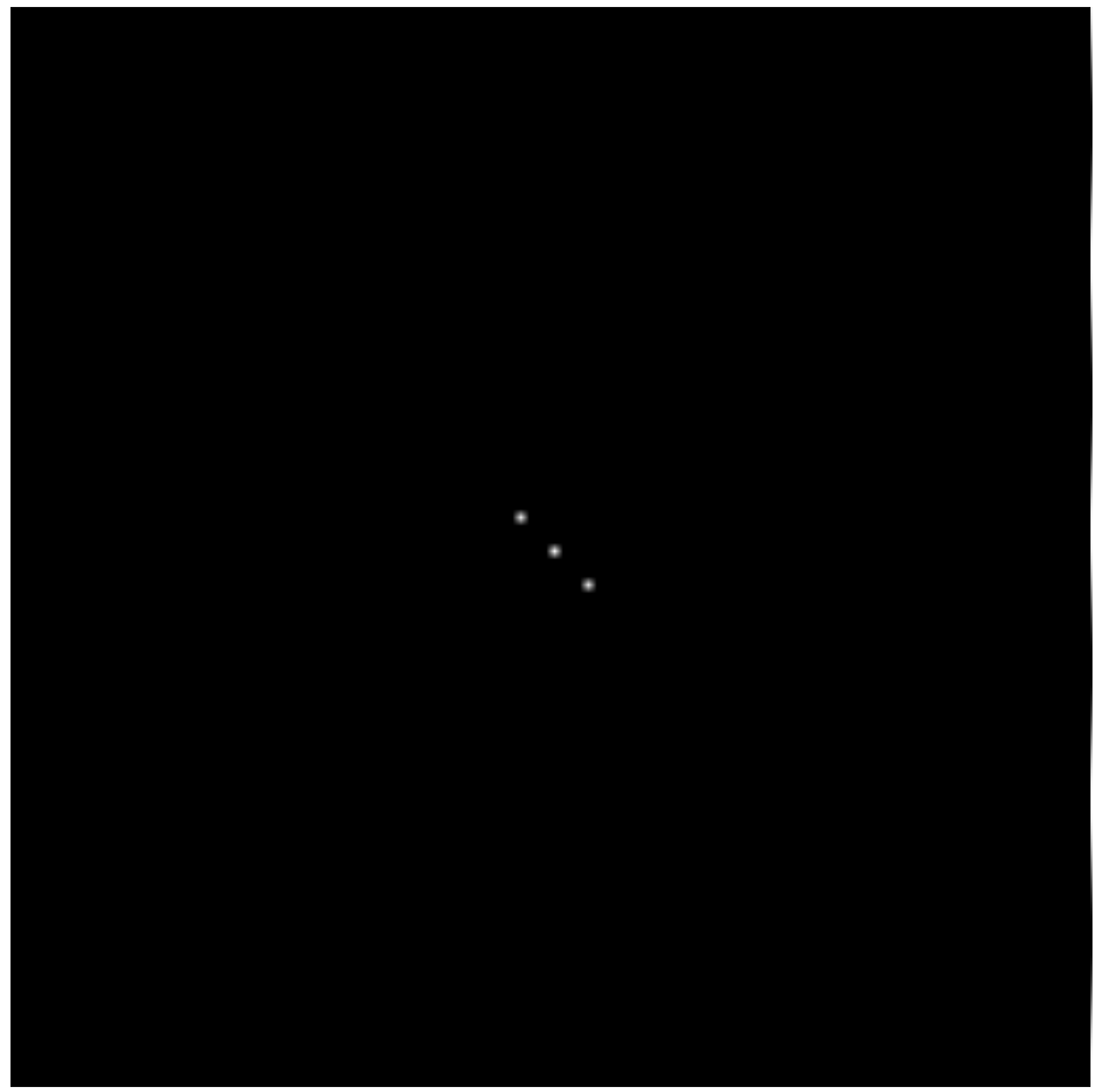
Amplitude in the frequency domain



http://cns-alumni.bu.edu/~slehar/fourier/fourier.html

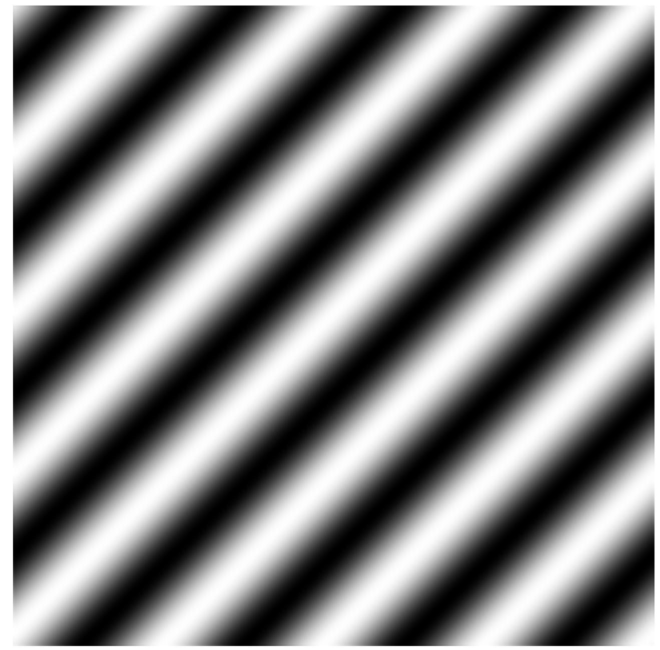
How would you generate this image with sine waves?





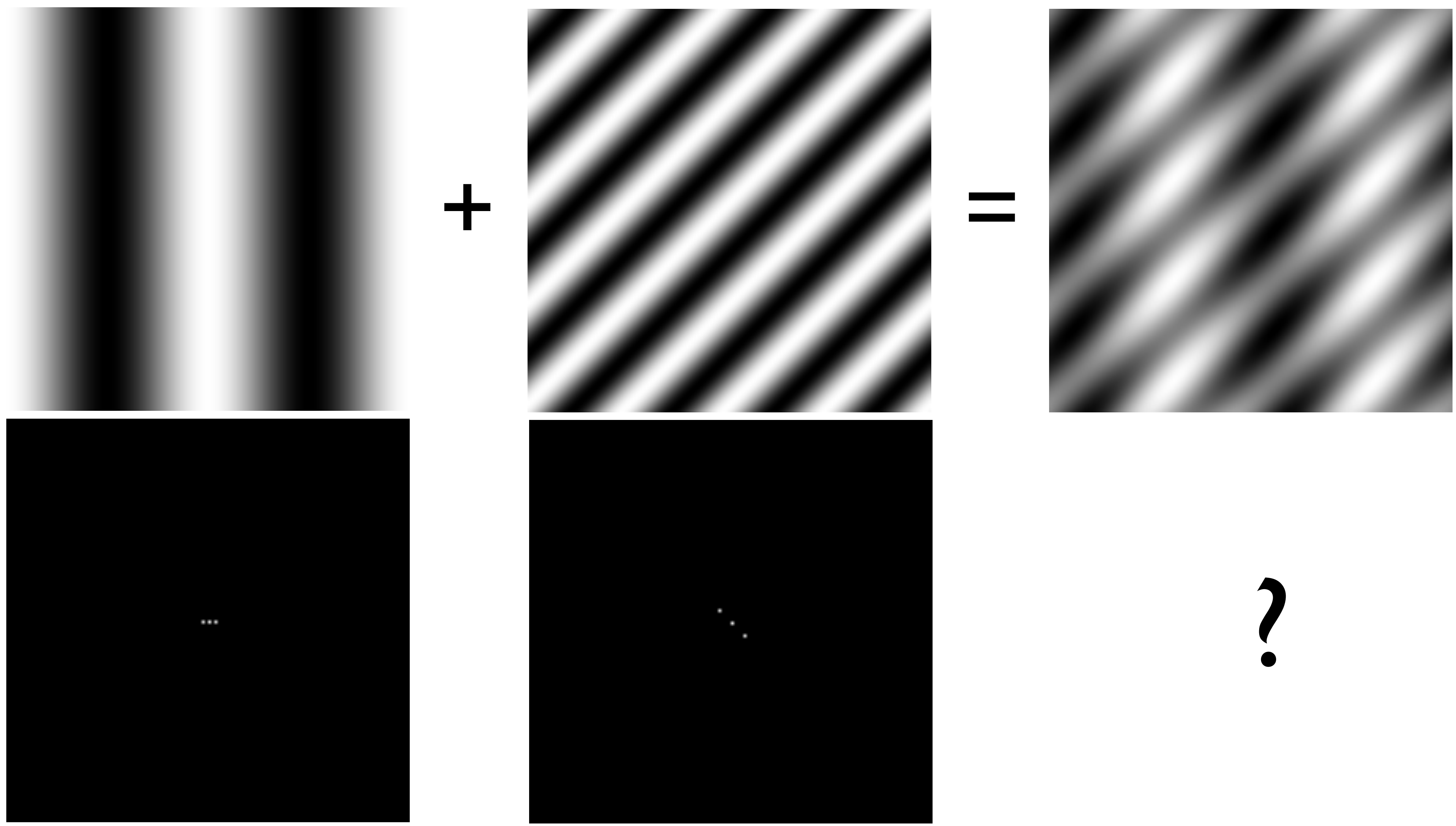
Has both an x and y component

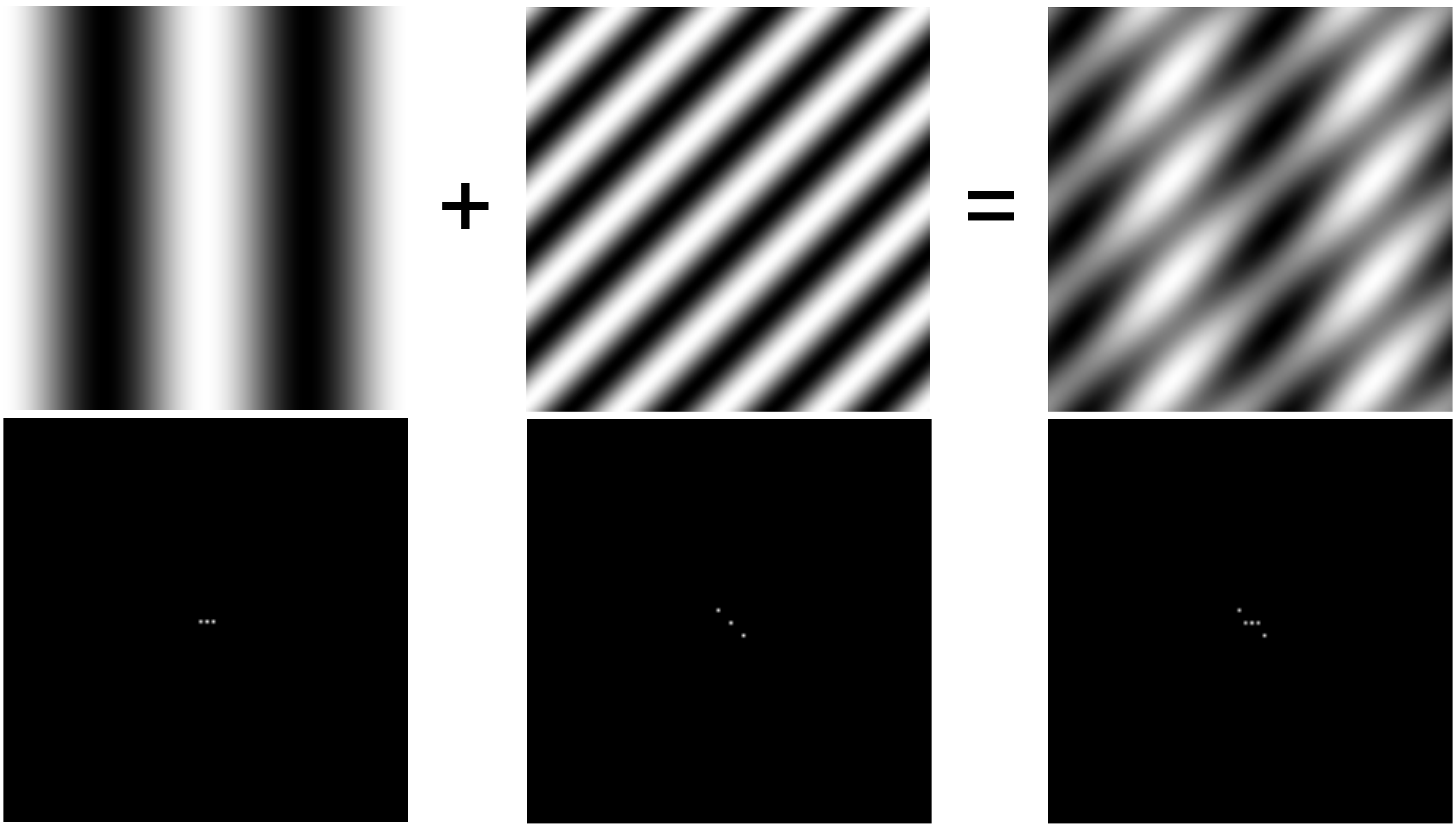
+



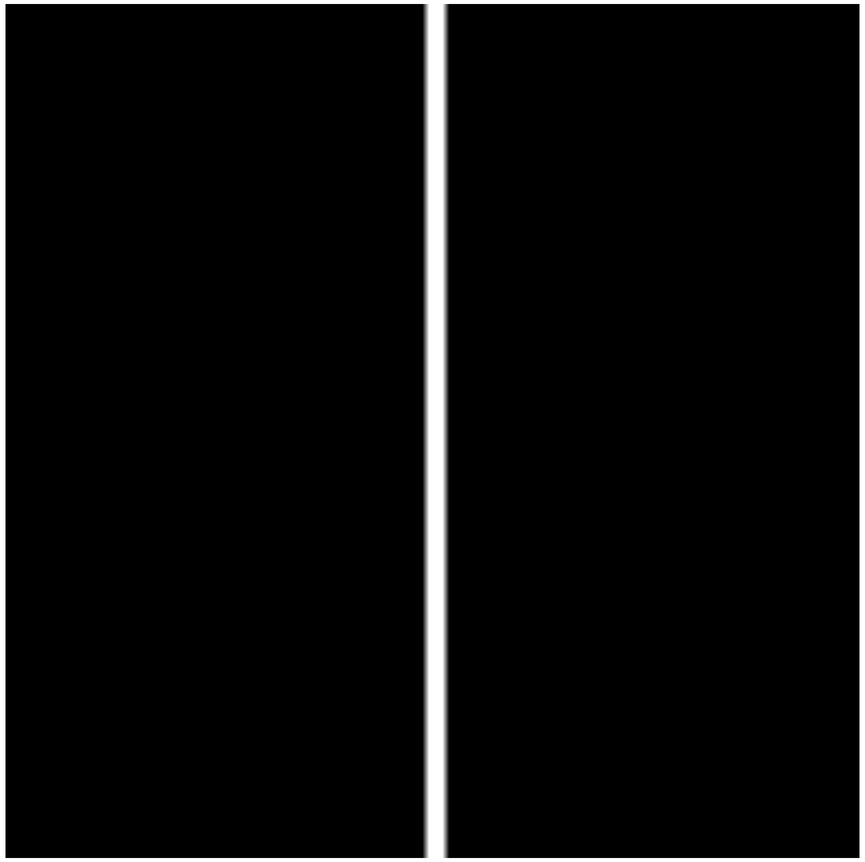
=

?





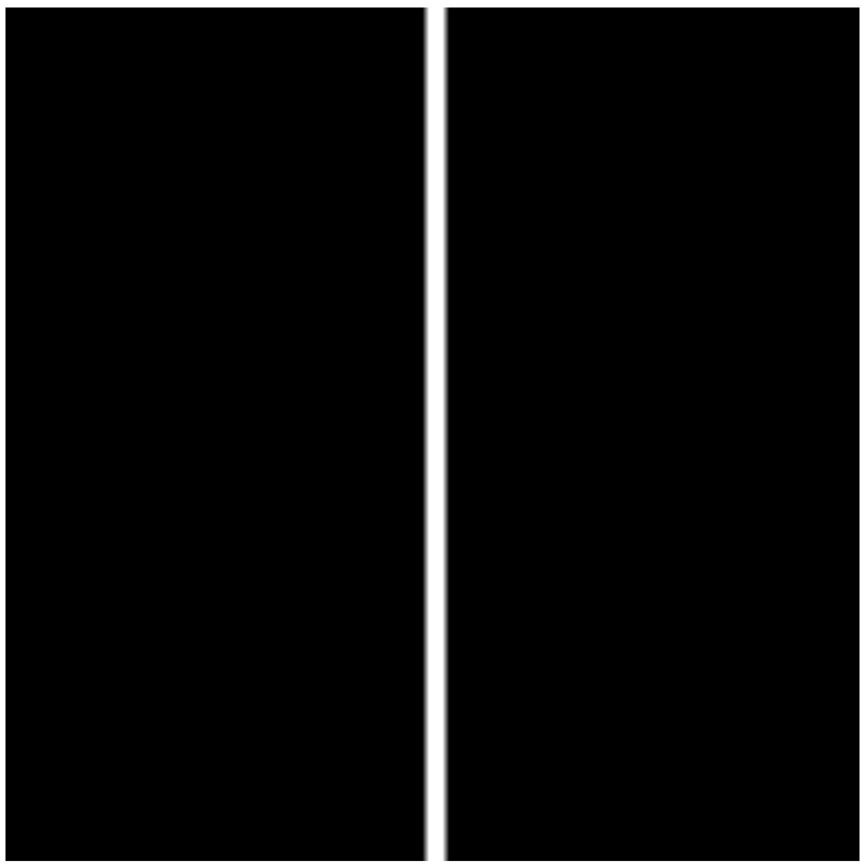
spatial domain visualization



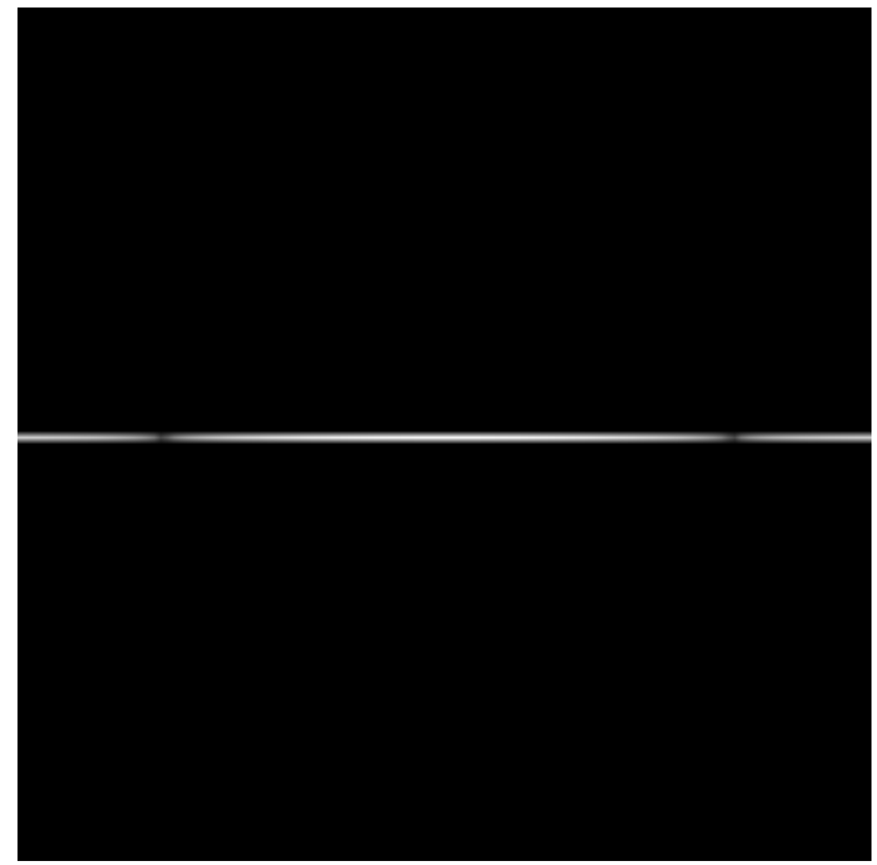
frequency domain visualization

?

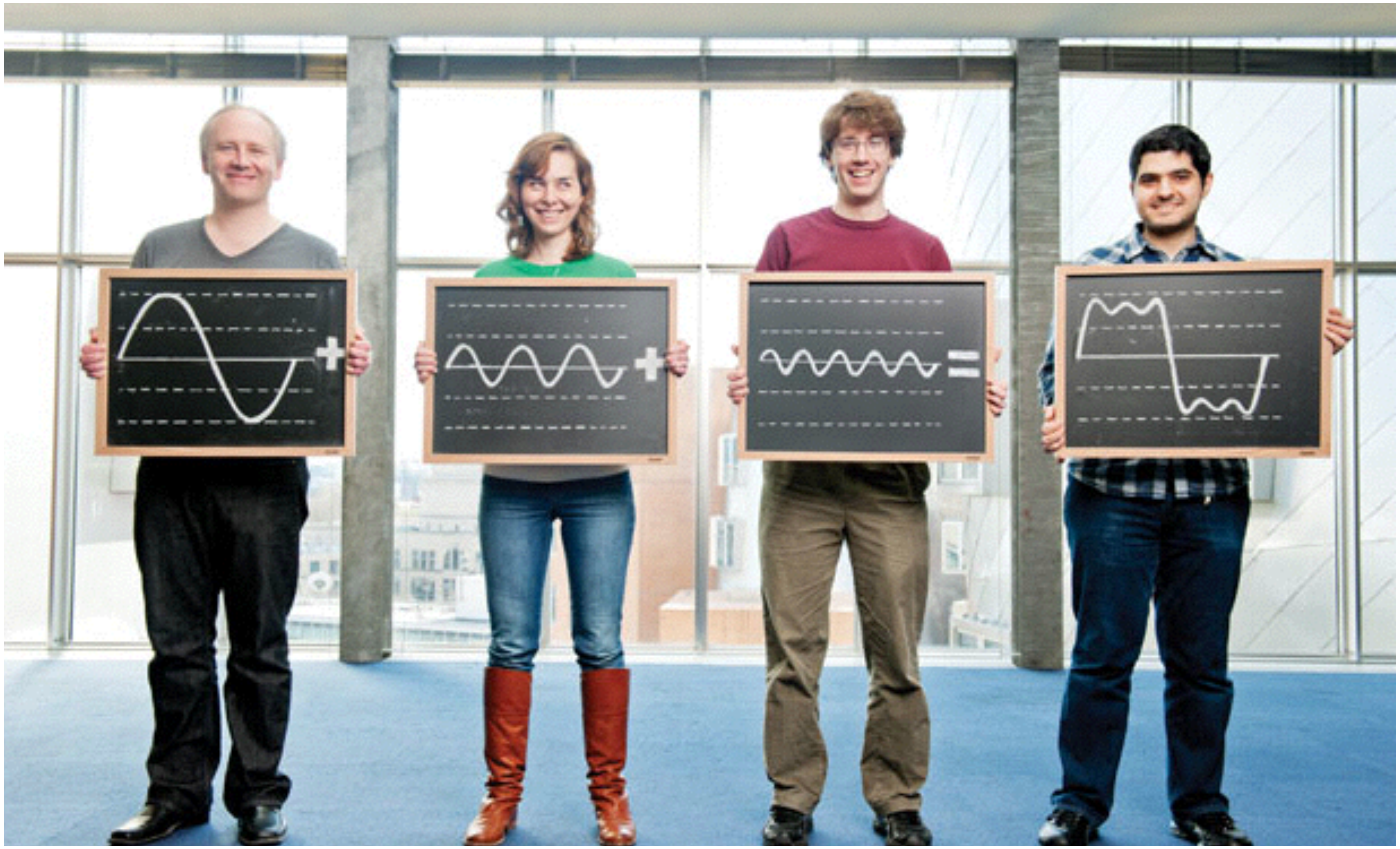
spatial domain visualization



frequency domain visualization

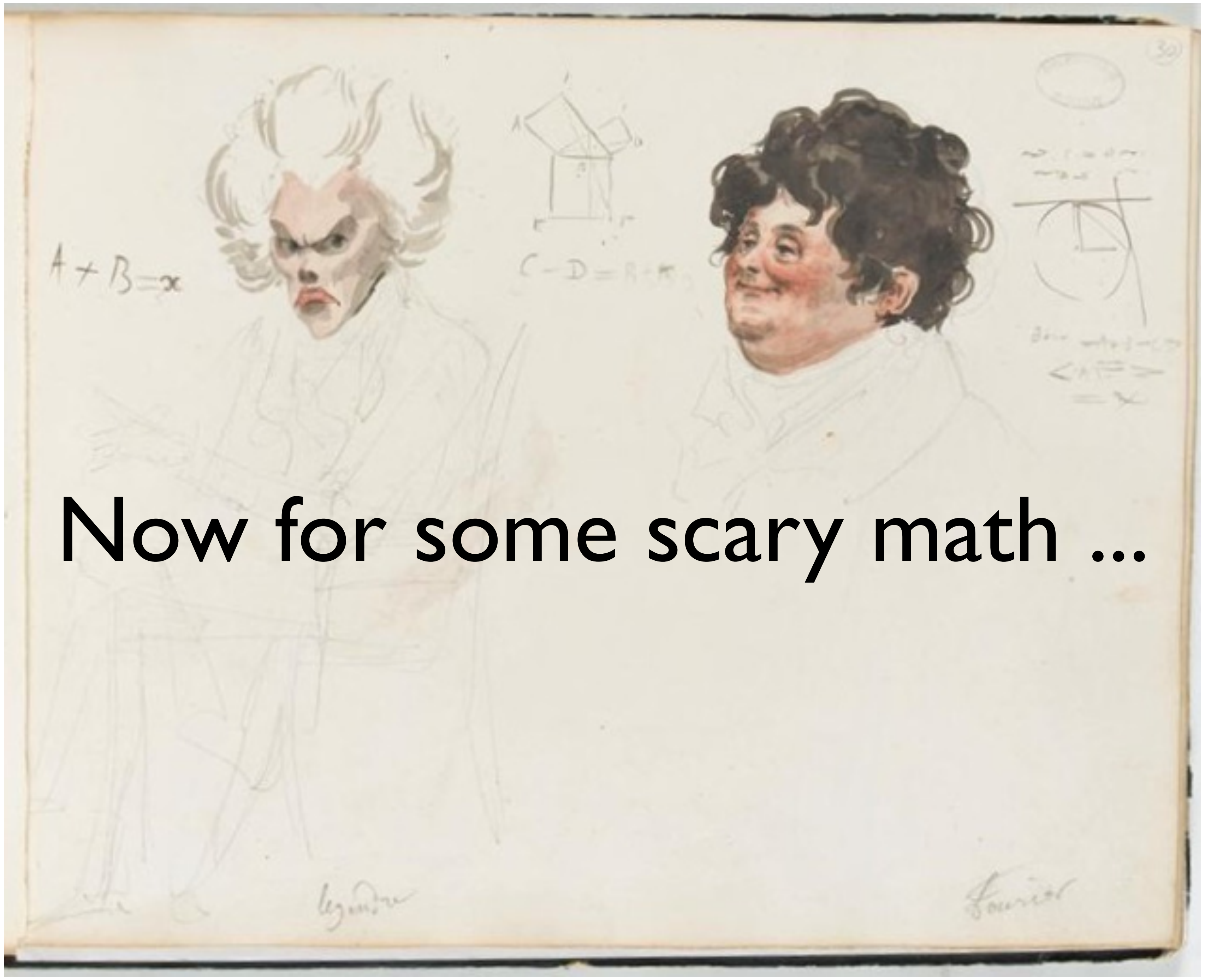


Need to be able to interpret 2D spectra to understand frequency filtering ...



Fourier Transform

16-385 Computer Vision



Now for some scary math ...

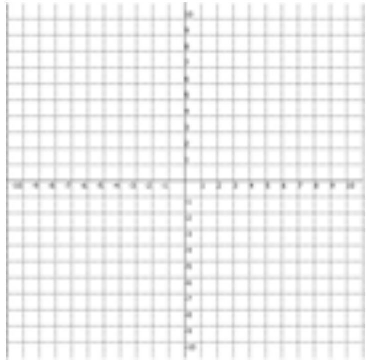
Recalling some basics...

Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

what's this? *what's this?*



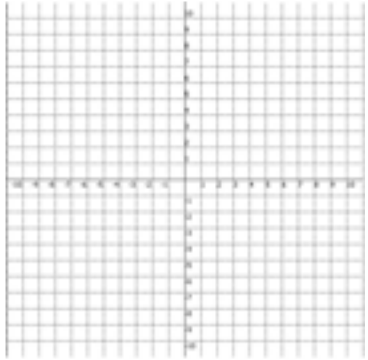
Recalling some basics...

Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

real imaginary



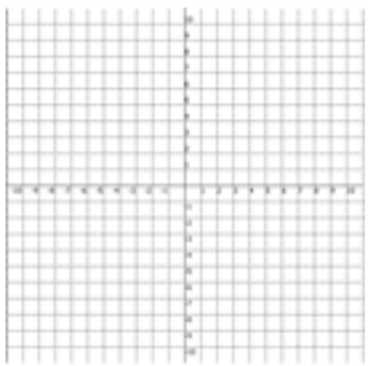
Recalling some basics...

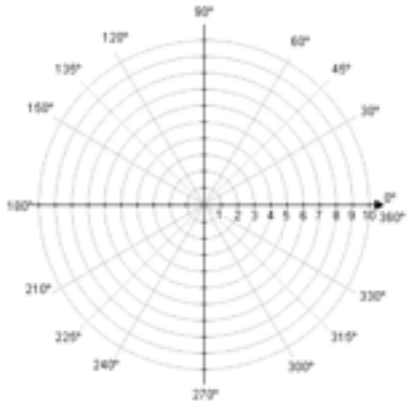
Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

real imaginary





what kind of transform is this?

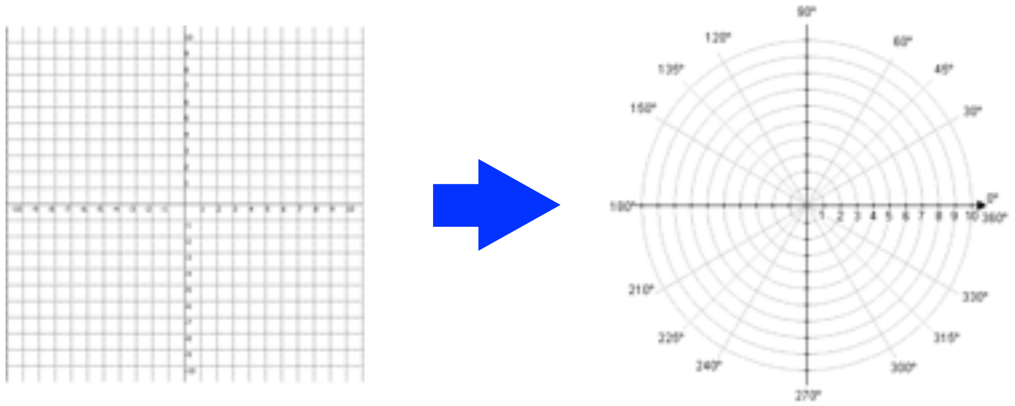
Recalling some basics...

Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

real imaginary



Polar

Recalling some basics...

Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

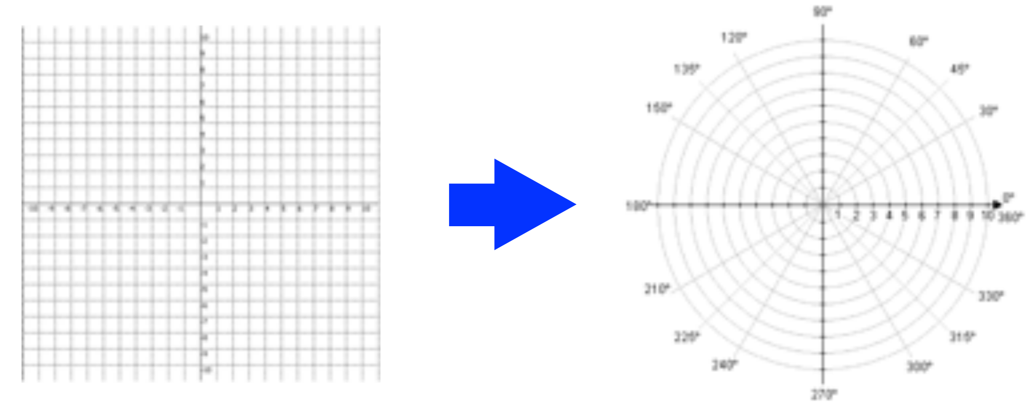
real imaginary

Alternative re-parameterization:

polar coordinates

$$r(\cos\theta + j\sin\theta)$$

How do you compute r and theta?



Recalling some basics...

Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

real imaginary

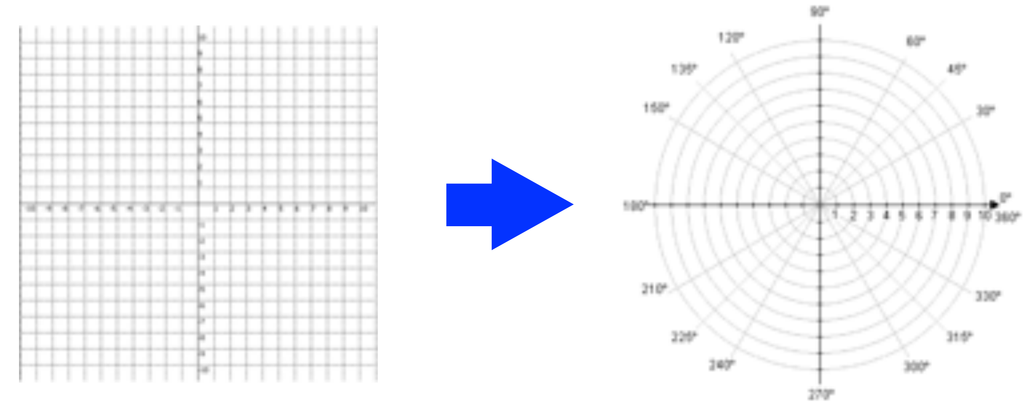
Alternative re-parameterization:

polar coordinates

$$r(\cos\theta + j\sin\theta)$$

polar transform

$$\theta = \tan^{-1}(\frac{I}{R}) \qquad r = \sqrt{R^2 + I^2}$$



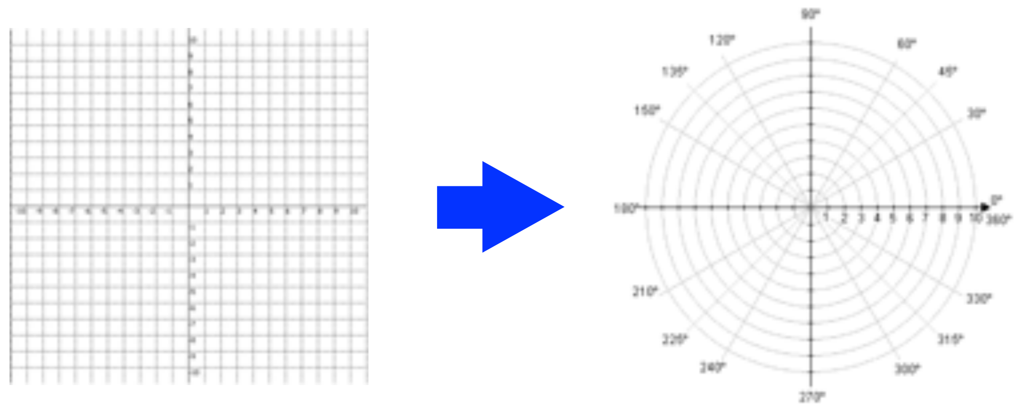
Recalling some basics...

Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

real imaginary



Alternative re-parameterization:

polar coordinates

$$r(\cos\theta + j\sin\theta)$$

polar transform

$$\theta = \tan^{-1}(\frac{I}{R}) \qquad r = \sqrt{R^2 + I^2}$$

How do you write this in exponential form?

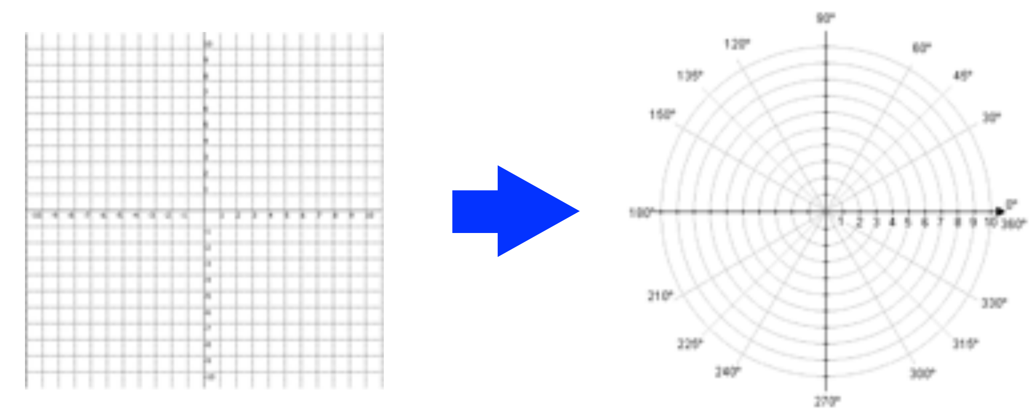
Recalling some basics...

Complex numbers have two parts:

rectangular coordinates

$$R + jI$$

real imaginary



Alternative re-parameterization:

polar coordinates

$$r(\cos\theta + j\sin\theta)$$

polar transform

$$\theta = \tan^{-1}\left(\frac{I}{R}\right) \qquad r = \sqrt{R^2 + I^2}$$

OR

exponential form

$$re^{j\theta}$$

'Euler's formula'

$$e^{j\theta} = \cos\theta + j\sin\theta$$

This will help us understanding of the Fourier transform equations ...

	Fourier transform	Inverse Fourier transform
Continuous	$F(k) = \int_{\infty}^{-\infty} f(x) e^{-j2\pi kx} dx$	$f(x) = \int_{\infty}^{-\infty} F(k) e^{j2\pi kx} dk$
Discrete	$F(k) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-j2\pi kx/N}$ $k = 0, 1, 2, \ldots, N-1$	$f(x) = \sum_{k=0}^{N-1} F(k) e^{j2\pi kx/N}$ $x = 0, 1, 2, \ldots, N-1$

Where is the connection to the 'summation of sine waves' idea?

	Fourier transform	Inverse Fourier transform
Continuous	$F(k) = \int_{\infty}^{-\infty} f(x) e^{-j2\pi kx} dx$	$f(x) = \int_{\infty}^{-\infty} F(k) e^{j2\pi kx} dk$
Discrete	$F(k) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-j2\pi kx/N}$ $k = 0, 1, 2, \ldots, N-1$	$f(x) = \sum_{k=0}^{N-1} F(k) e^{j2\pi kx/N}$ $x = 0, 1, 2, \ldots, N-1$

Where is the connection to the 'summation of sine waves' idea?

Where is the connection to the 'summation of sine waves' idea?

$$f(x) = \sum_{k=0}^{N-1} F(k) e^{j2\pi kx/N}$$

'Euler's formula'

$$e^{j\theta} = \cos\theta + j\sin\theta$$

scaling parameter

$$f(x) = \sum_{k=0}^{N-1} F(k) \left\{ \cos(2\pi kx) + j\sin(2\pi kx) \right\}$$

sum over frequencies

wave components

"So how do you actually compute the DFT?"

–A. Student

Computing the Discrete Fourier Transform…

$$F(k) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-j2\pi kx/N}$$

…is just a matrix multiplication.

$$\boldsymbol{F} = \boldsymbol{W} \boldsymbol{f}$$

$$\begin{bmatrix} F(0) \\ F(1) \\ F(2) \\ F(3) \\ \vdots \\ F(N-1) \end{bmatrix} = \begin{bmatrix} W^0 & W^0 & W^0 & W^0 & \cdots & W^0 \\ W^0 & W^1 & W^2 & W^3 & \cdots & W^{N-1} \\ W^0 & W^2 & W^4 & W^6 & \cdots & W^{N-2} \\ W^0 & W^3 & W^6 & W^9 & \cdots & W^{N-3} \\ \vdots & & & & \ddots & \vdots \\ W^0 & W^{N-1} & W^{N-2} & W^{N-3} & \cdots & W^1 \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \\ \vdots \\ f(N-1) \end{bmatrix}$$

$$W = e^{-j2\pi/N} \qquad W = W^{2N}$$

Example

input signal

$$\begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \end{bmatrix} = \begin{bmatrix} 8 \\ 4 \\ 8 \\ 0 \end{bmatrix}$$

DFT

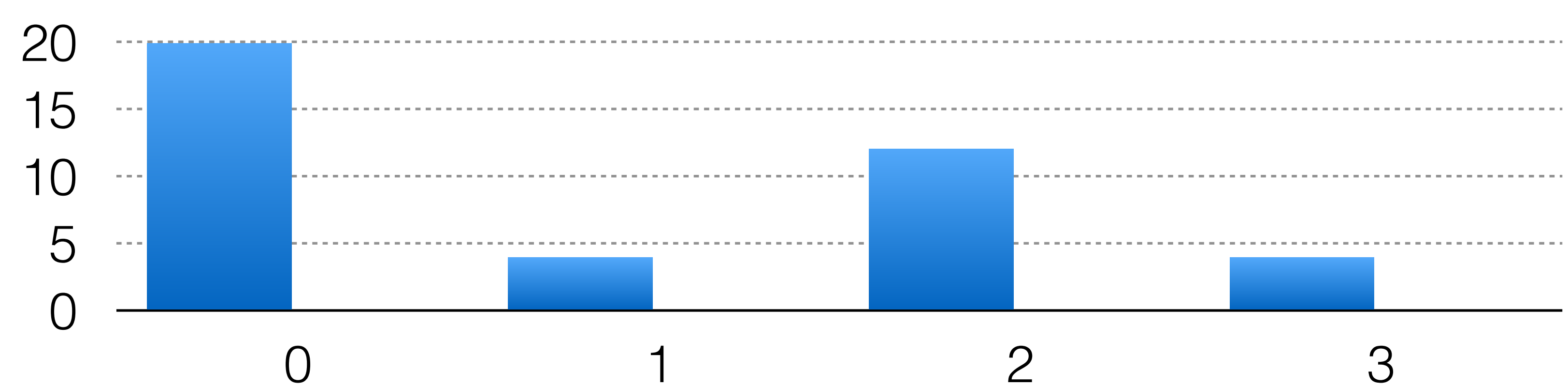
$$F(k) = \sum_{x=0}^{3} f(x) e^{-j2\pi xk/4}$$

$$= \sum_{x=0}^{3} f(x)(-j)^{xk}$$

Frequency Domain representation

$$\begin{bmatrix} F(0) \\ F(1) \\ F(2) \\ F(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \end{bmatrix} = \begin{bmatrix} 20 \\ -j4 \\ 12 \\ j4 \end{bmatrix}$$

Frequency spectrum



The Convolution Theorem

- The Fourier transform of the convolution of two functions is the product of their Fourier transforms

$$\mathcal{F}\{g \star h\} = \mathcal{F}\{g\}\mathcal{F}\{h\}$$

- The inverse Fourier transform of the product of two Fourier transforms is the convolution of the two inverse Fourier transforms

$$\mathcal{F}^{-1}\{gh\} = \mathcal{F}^{-1}\{g\} \star \mathcal{F}^{-1}\{h\}$$

- **Convolution** in spatial domain is equivalent to **multiplication** in frequency domain!