Classification: Bags of Words

Gary Overett (Slides adapted from CMU 16-720 (2014 course))

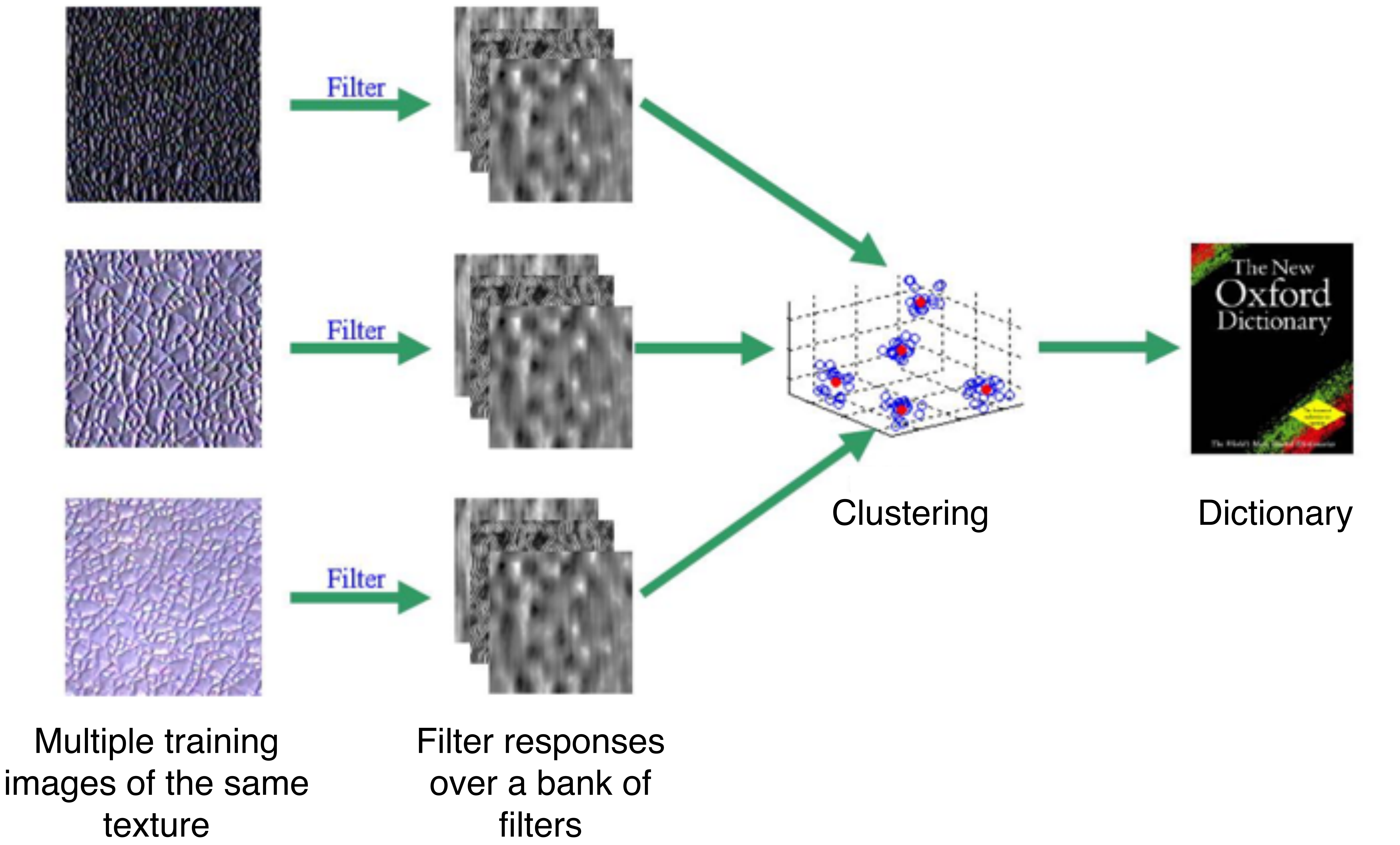
Example: Texture Classification

- Profound observation: Cows and buildings don’t look the same!
- Basic idea: Model the distribution of “texture” over the image (or over a region) and classify in different classes based on the texture models learned from training examples.

Cow

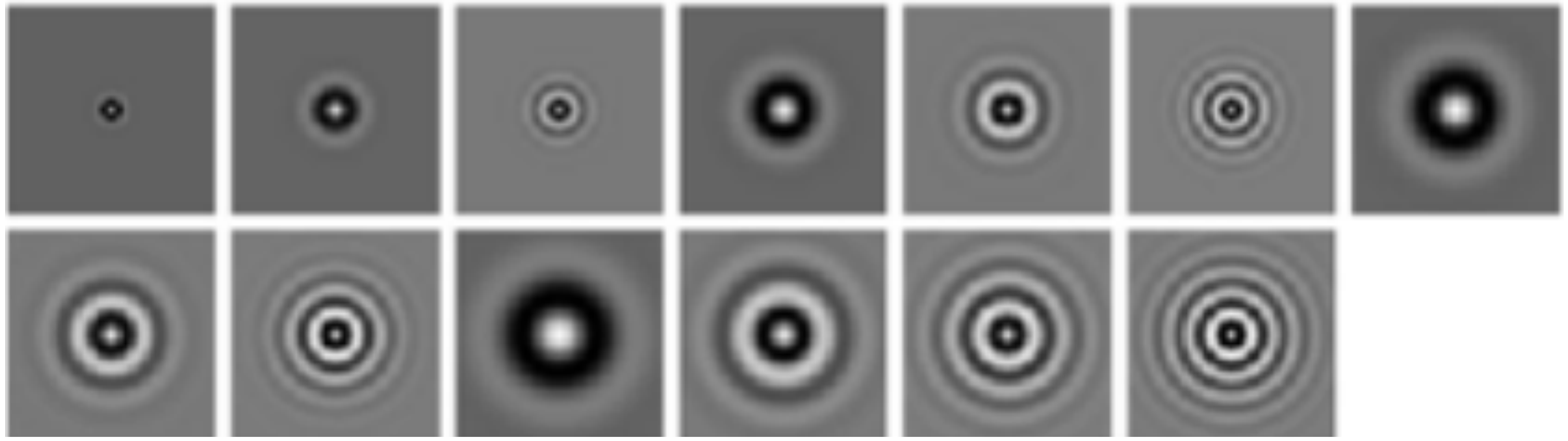
Building

The Concept of visual words

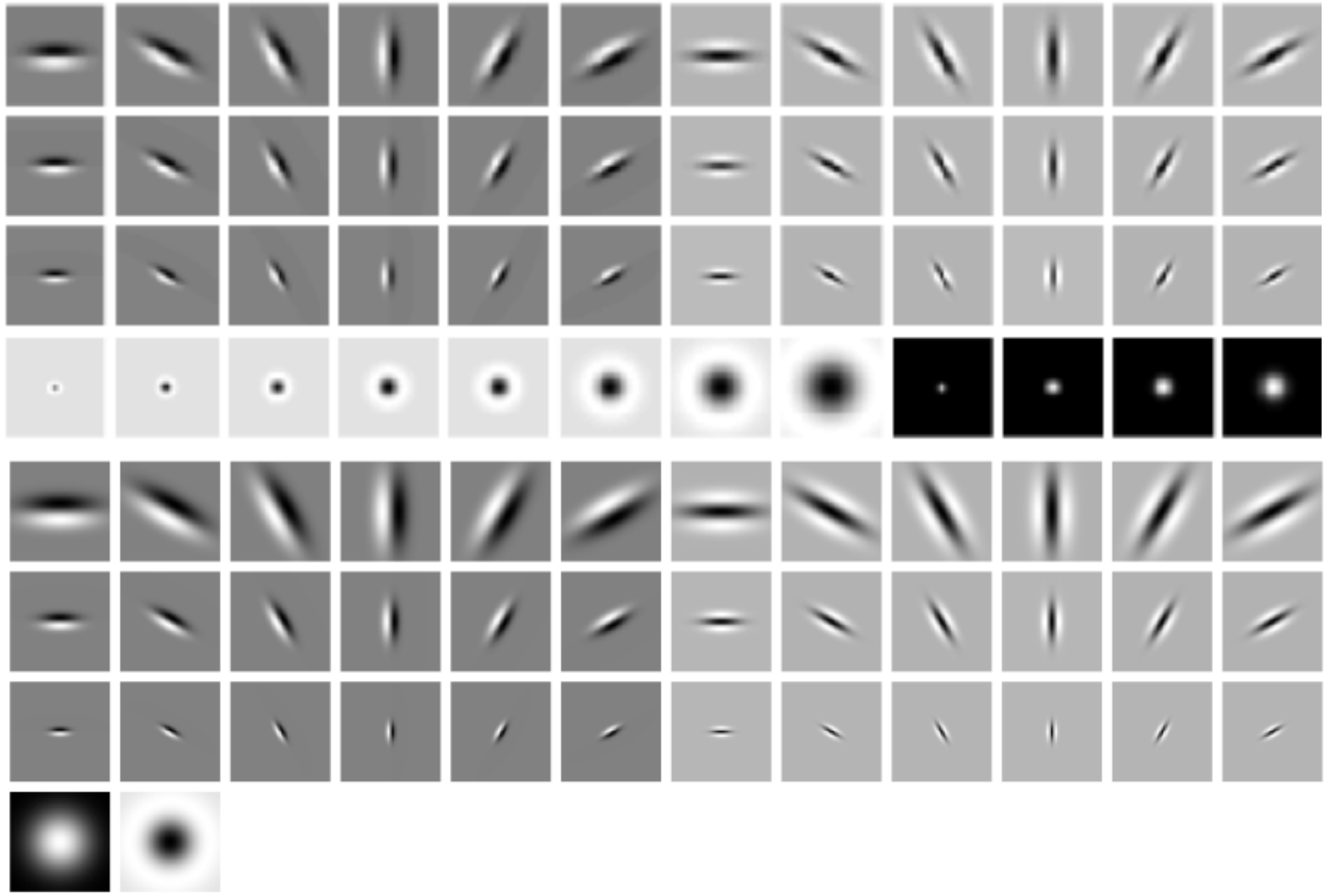


Example of Filter Banks

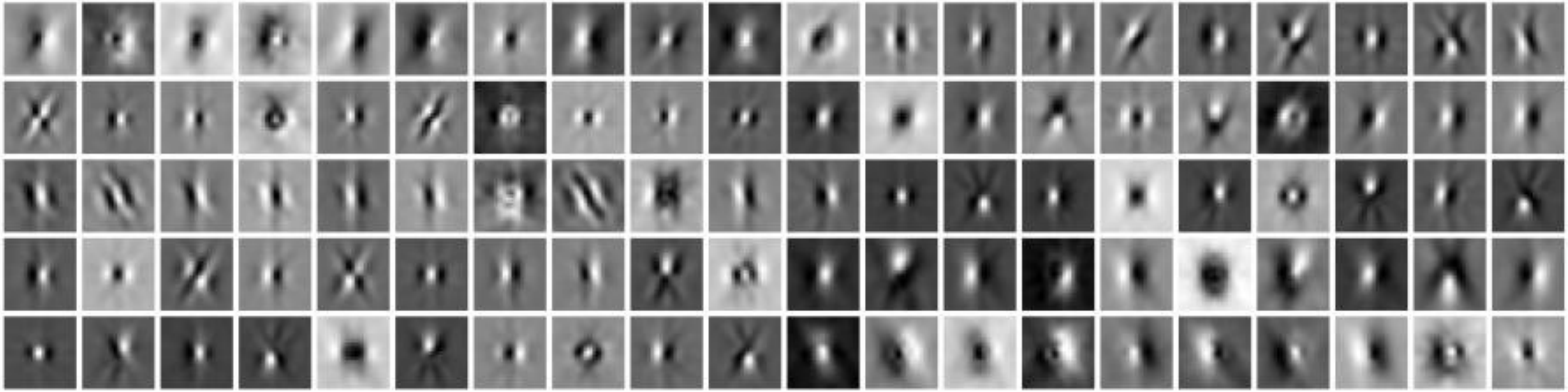
Isotropic Gabor



Gaussian derivatives at different scales and orientations



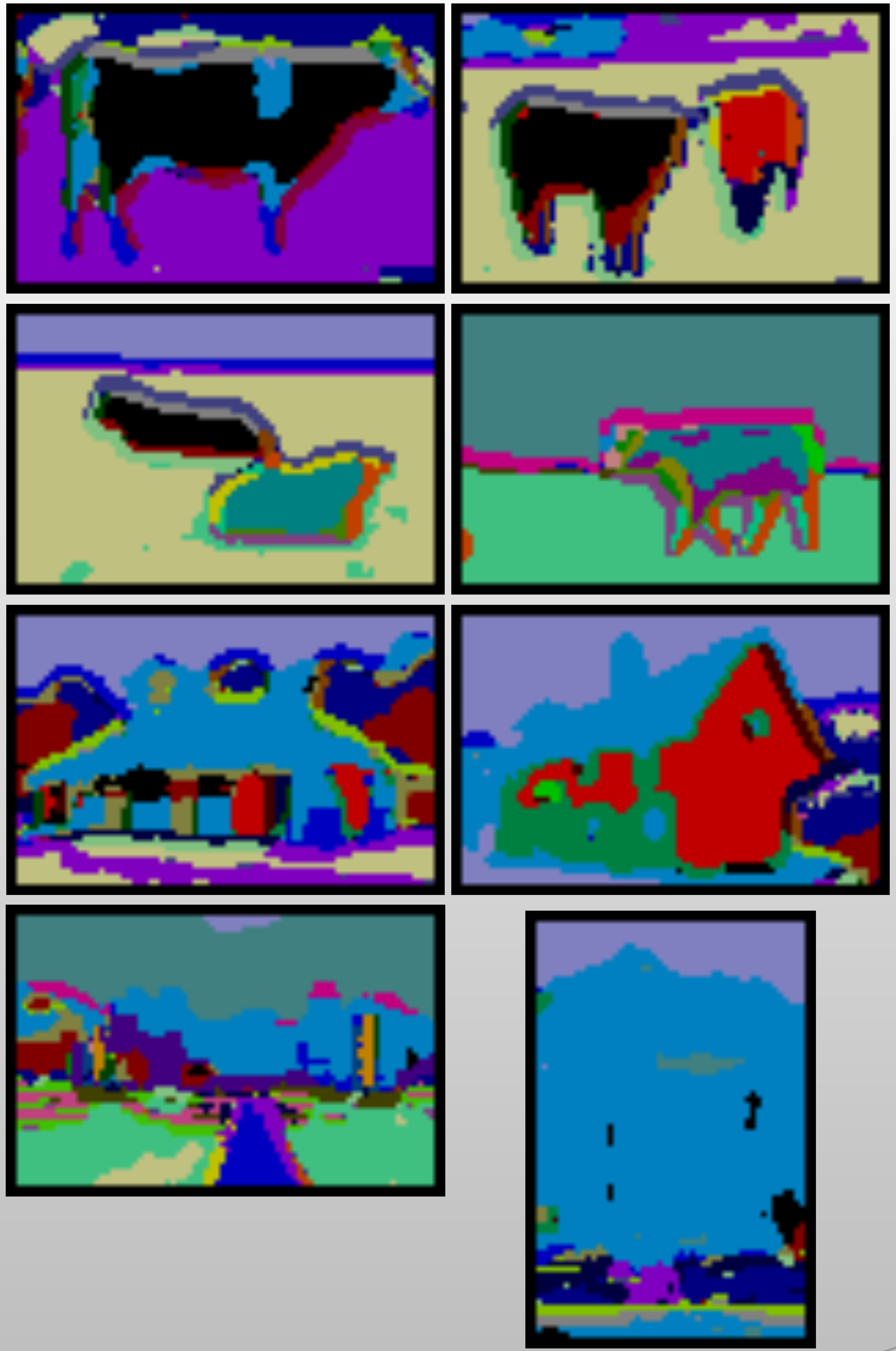
Example Textons



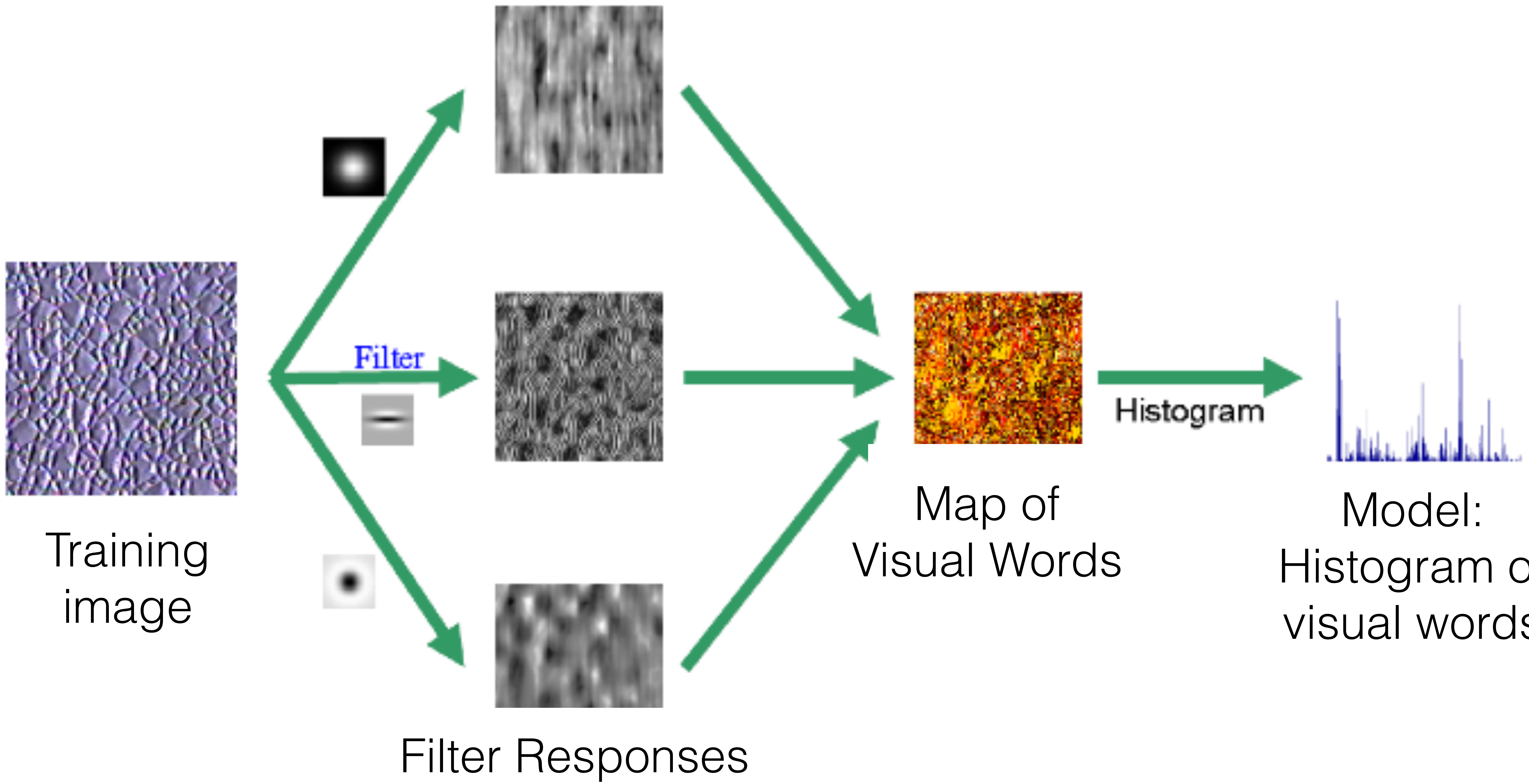
Example: Visual words in photographs

Images

Word Map



Modeling "Word" Distributions



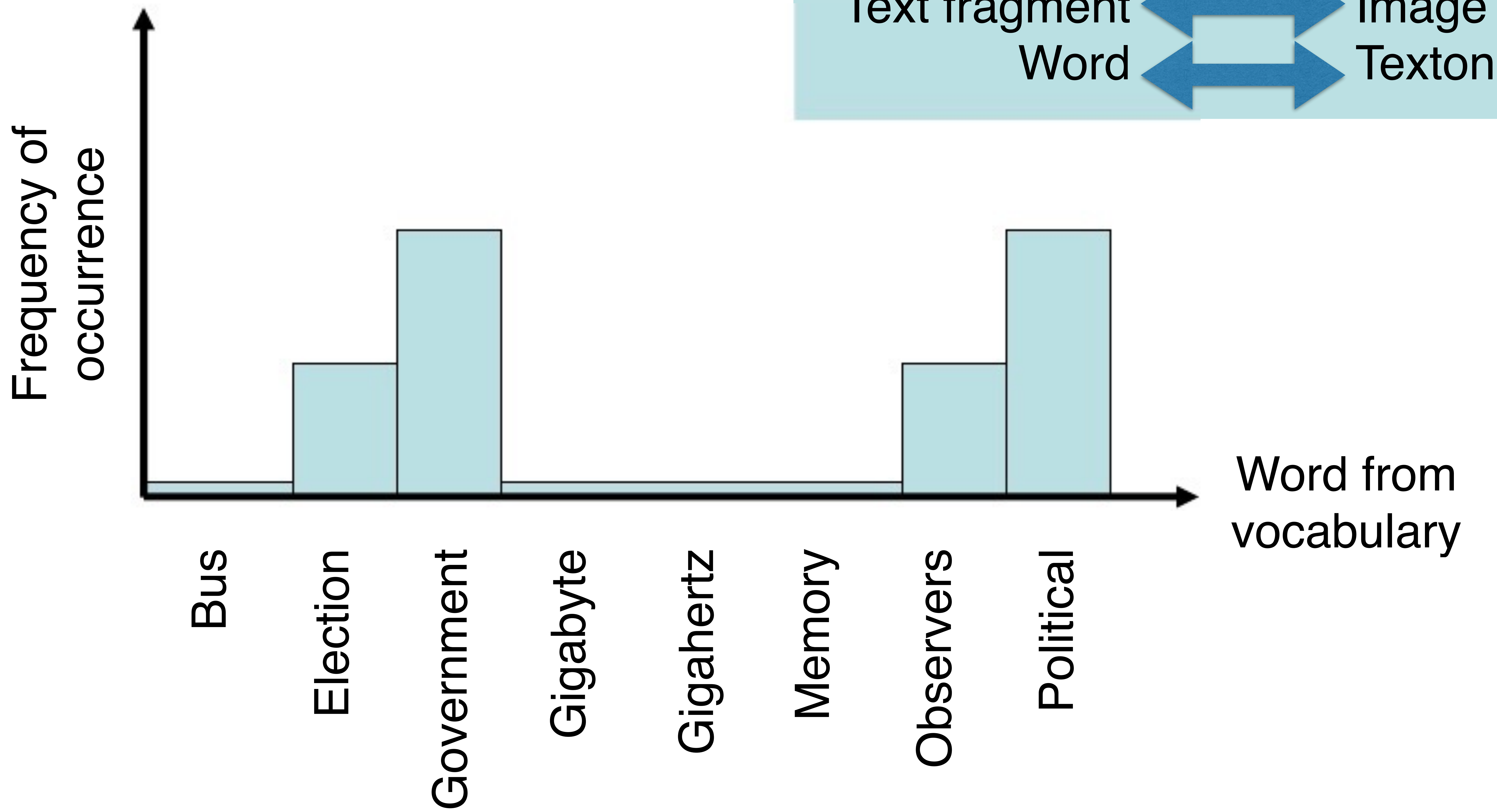
Analogy with Text Analysis

Analogy:

Text fragment ⟷ Image region

Word ⟷ Texton

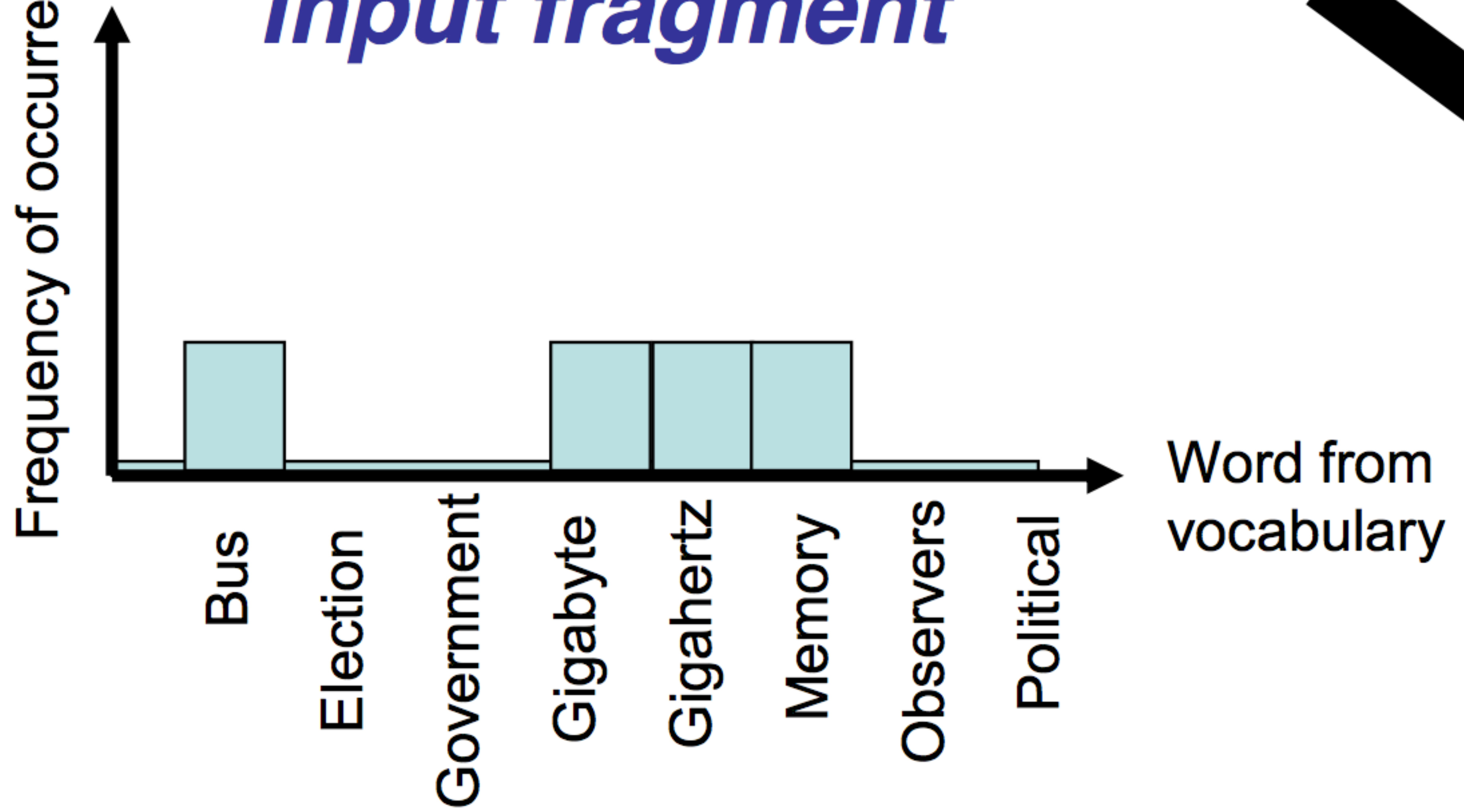
Political observers say that the government of Zorgia does not control the political situation. The government will not hold elections ...



Analogy with Text Analysis

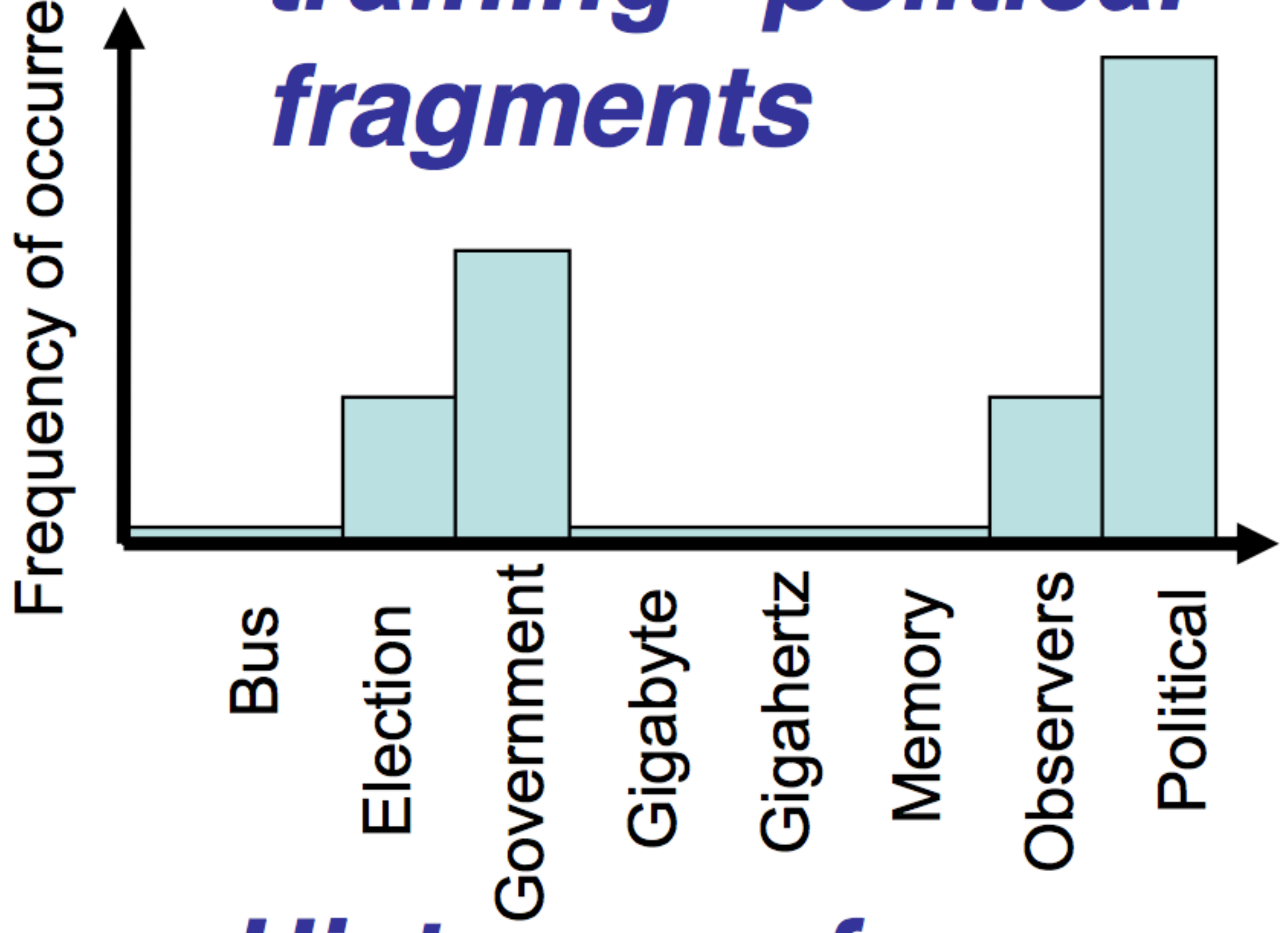
The ZH-20 unit is a 200Gigahertz processor with 2Gigabyte memory. Its strength is its bus and high-speed memory……

Histogram from input fragment

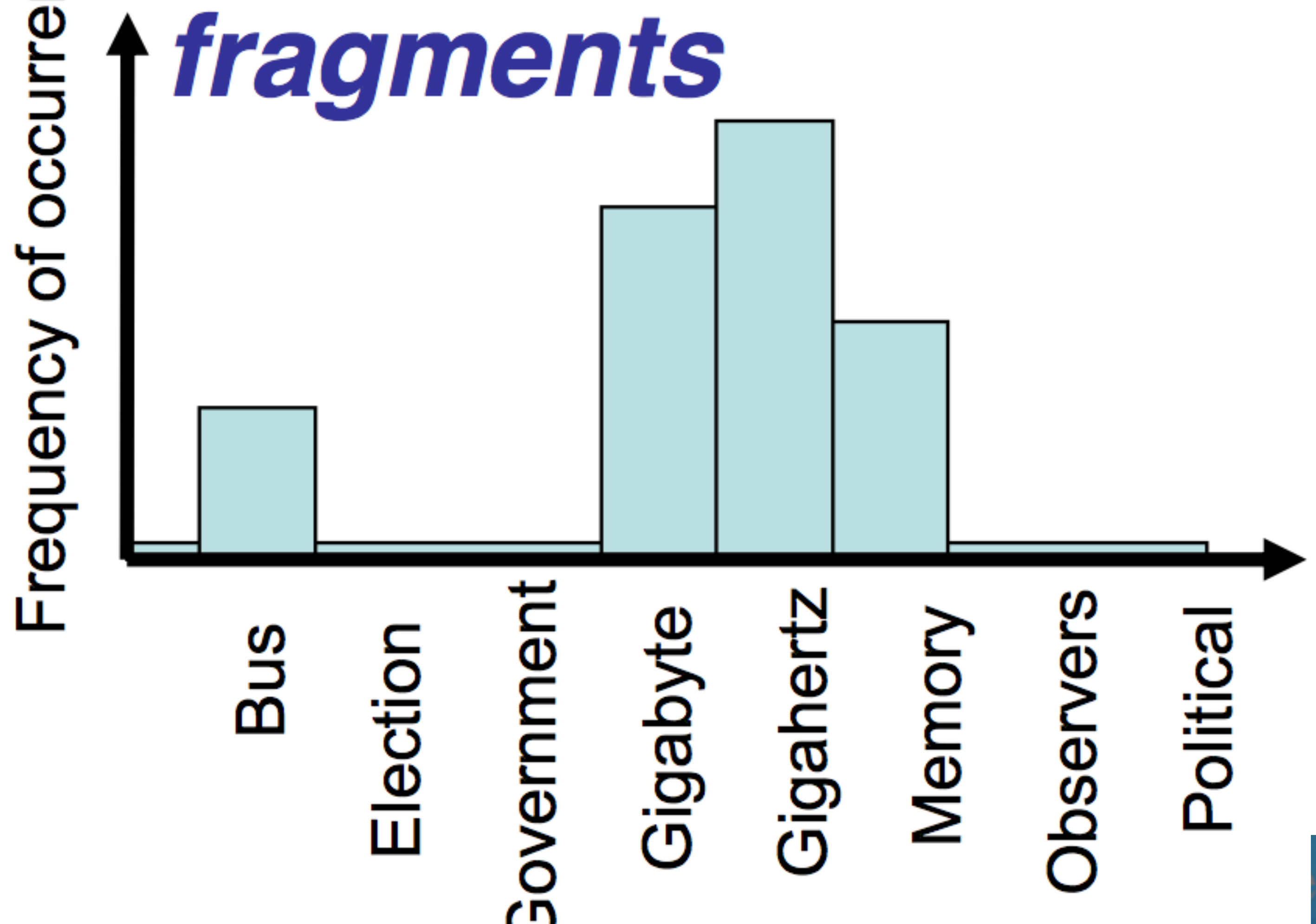


Compare

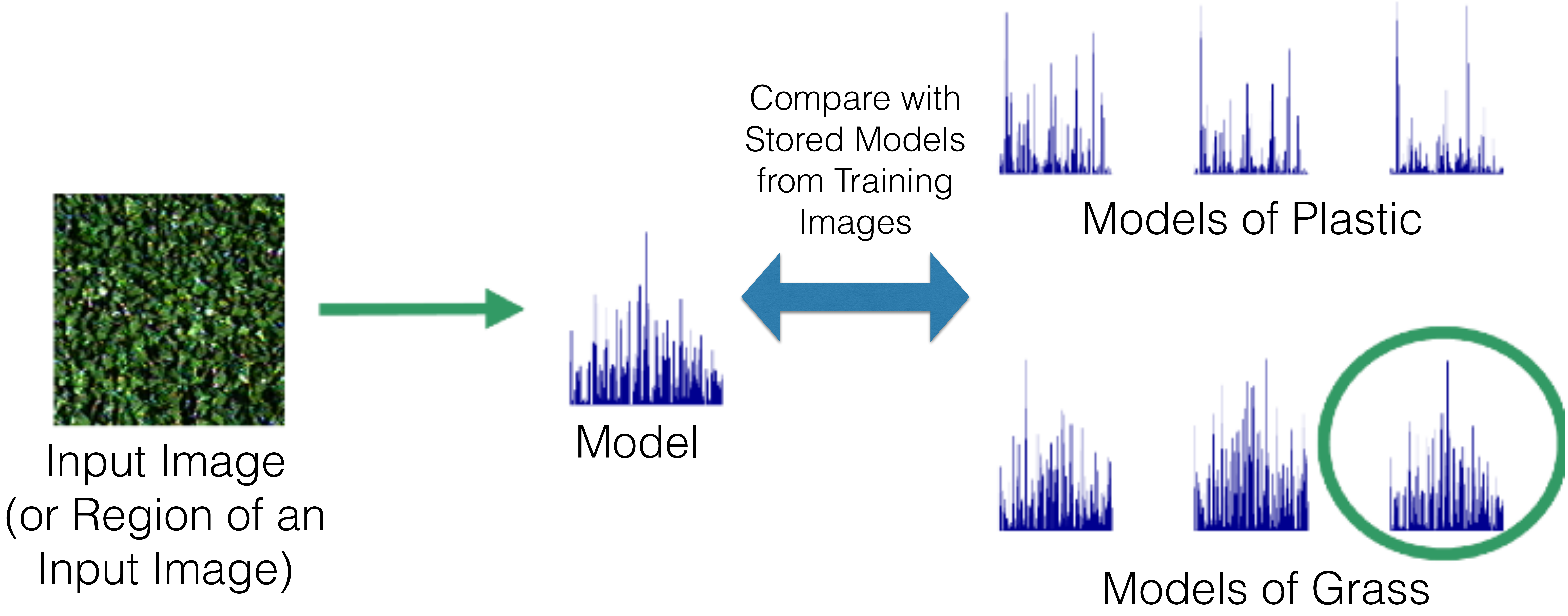
Histogram from training "political" fragments



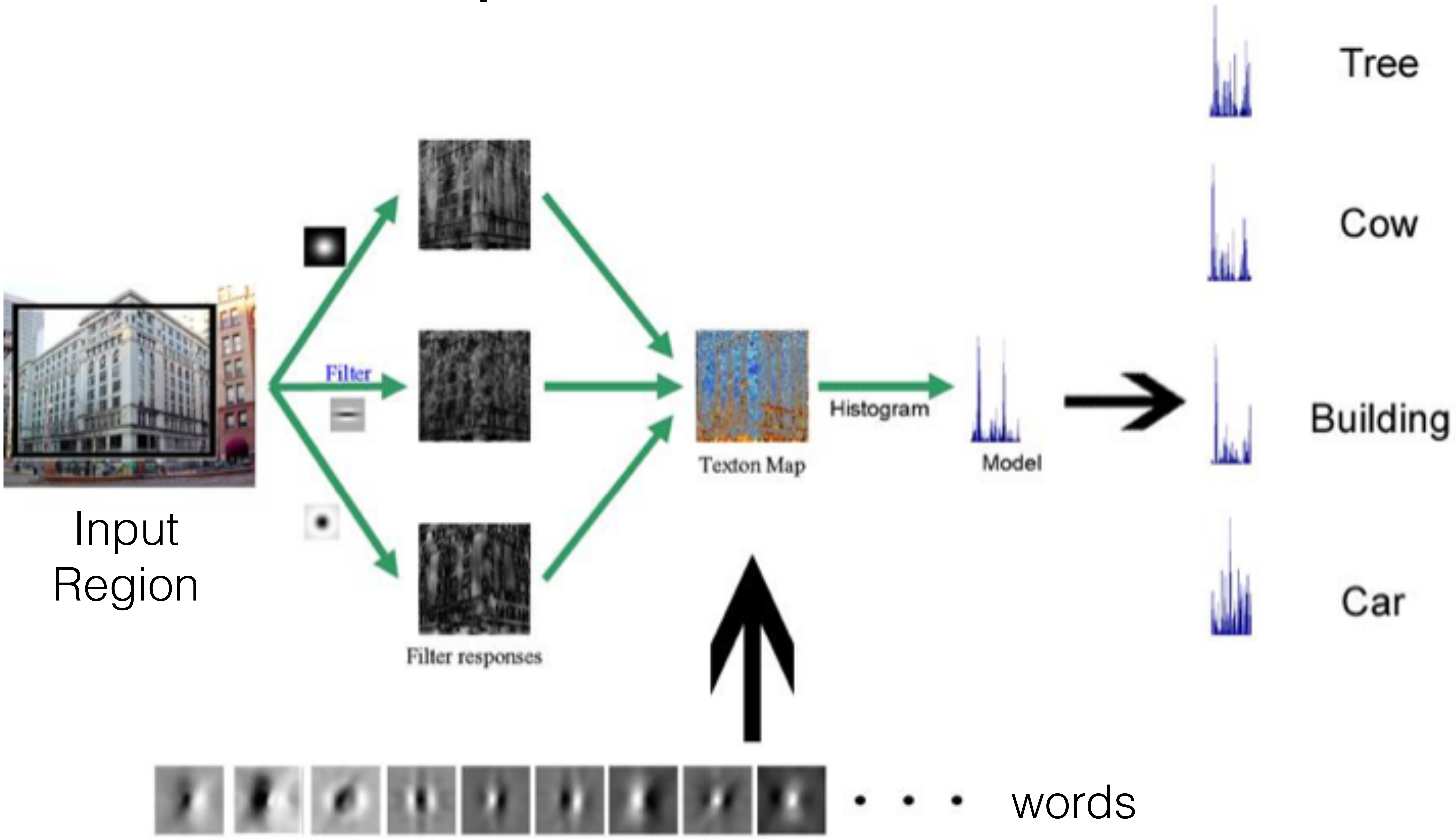
Histogram from training "computer" fragments



Classification

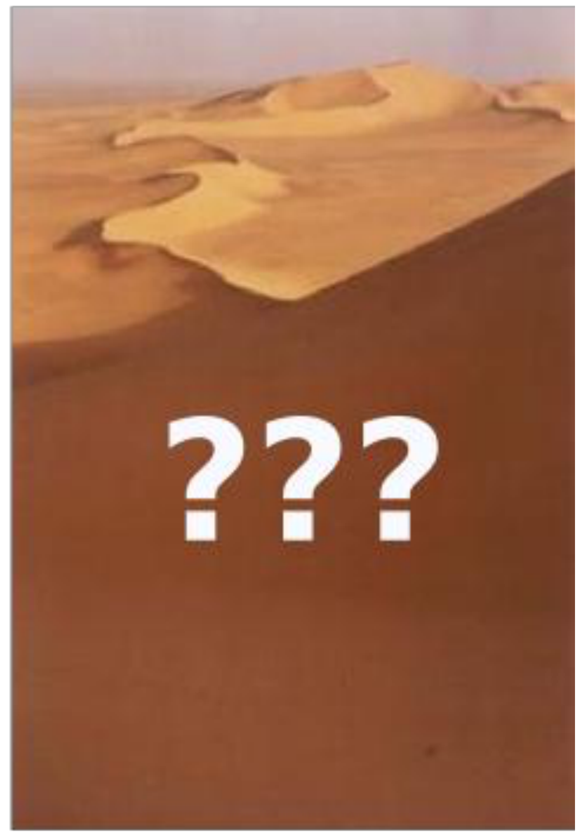


Example Classification



Examples


cow
cow
grass
sky
aeropla
grass
tree
car
bike
sky
building
grass
face
cow
sky
aeroplane
tree
car
car
bike
bike
building
tree
face
grass



- How many textons/words?
- What filters?
- How to construct clusters?
- How to compare histogram distributions?
- How to exploit the spatial distribution of textons (these examples completely ignore the relative positions of textons in the image)?

Frequency analysis, pyramids, texture analysis, applications

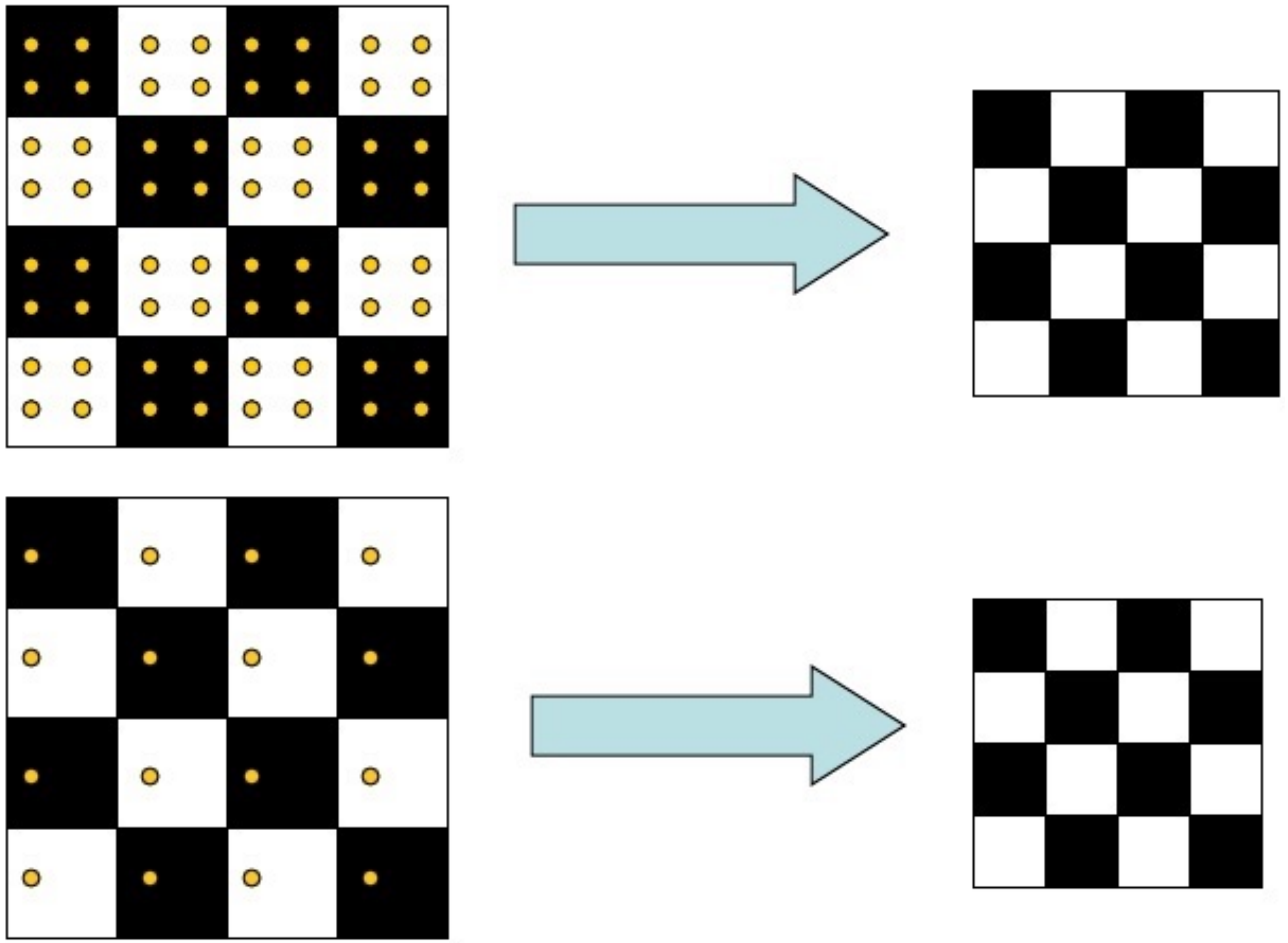
Outline

- Measuring frequencies in images: Definitions, properties
- Sampling issues
- Relation with Gaussian smoothing
- Image pyramids
- Gabor filters and wavelets
- Examples: Texture classification and category recognition

Questions

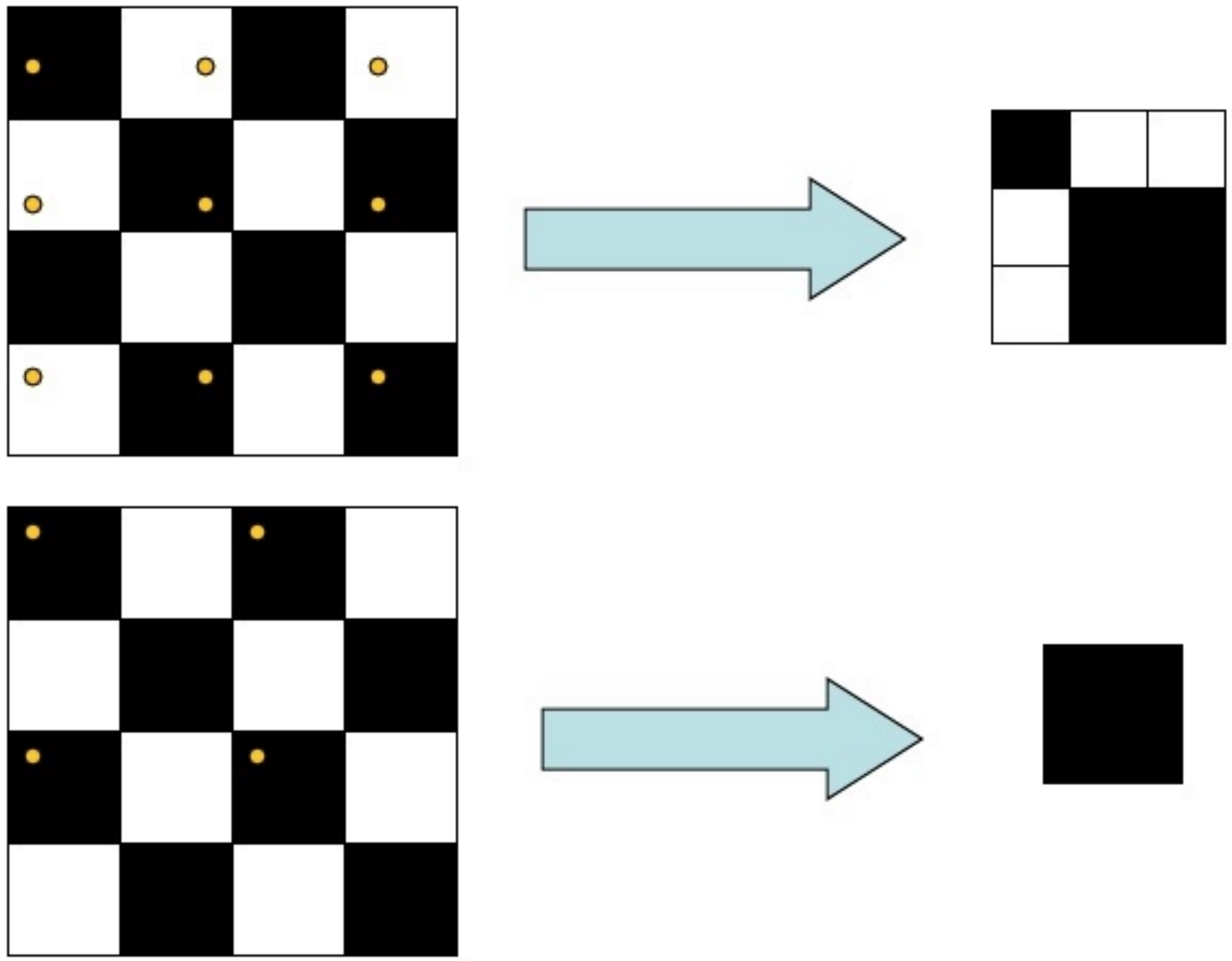
- How do discrete images differ from continuous images?
- How do we avoid aliasing while sampling?

Sampling an Image (Good Case)

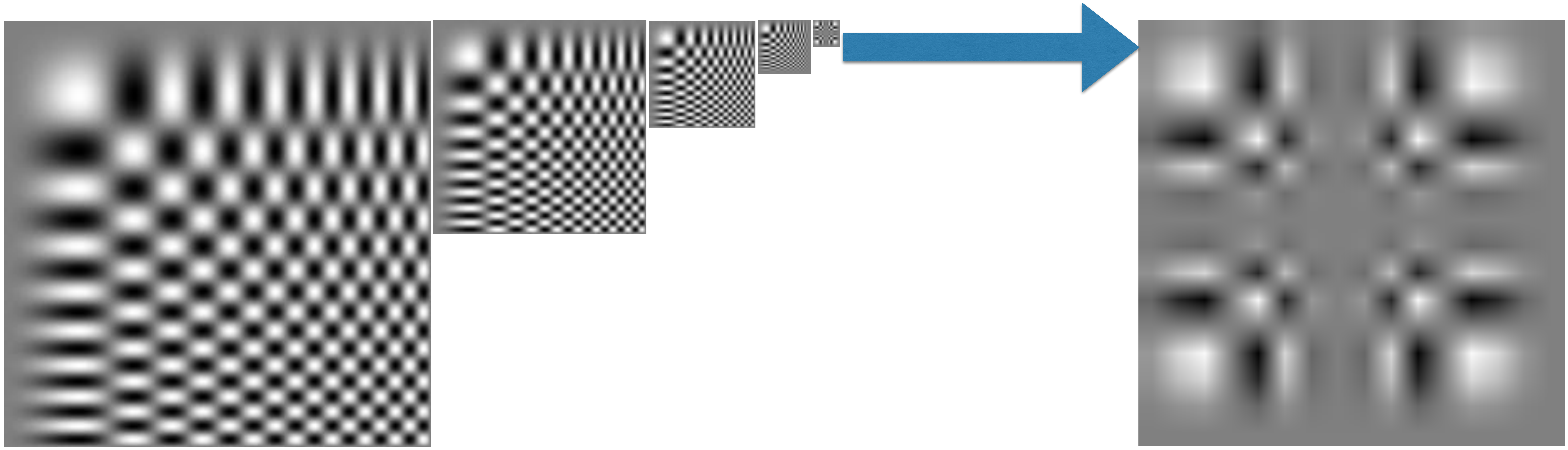


Sampling an Image (Bad Case)

Undersampling



Downsampling



Constructing a pyramid by taking every second pixel leads to layers that badly misrepresents the top layer

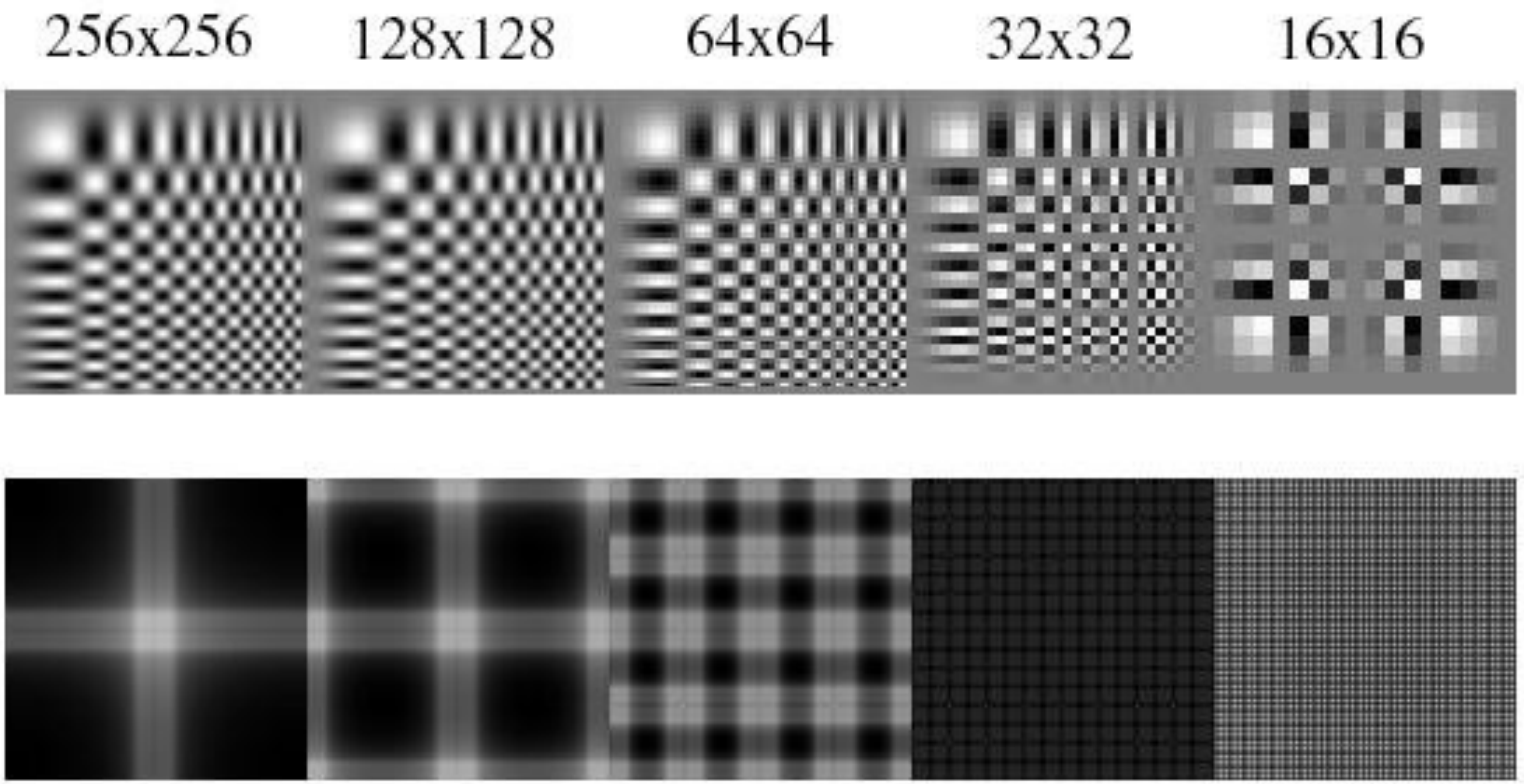
Low-pass filtering before sampling

- The minimum frequency at which we must sample a signal in order to be able to fully reconstruct it called the Nyquist frequency.
- Nyquist frequency = 2 times the maximum frequency contained in the waveform.
- The message of the FT is that high frequencies lead to trouble with sampling.

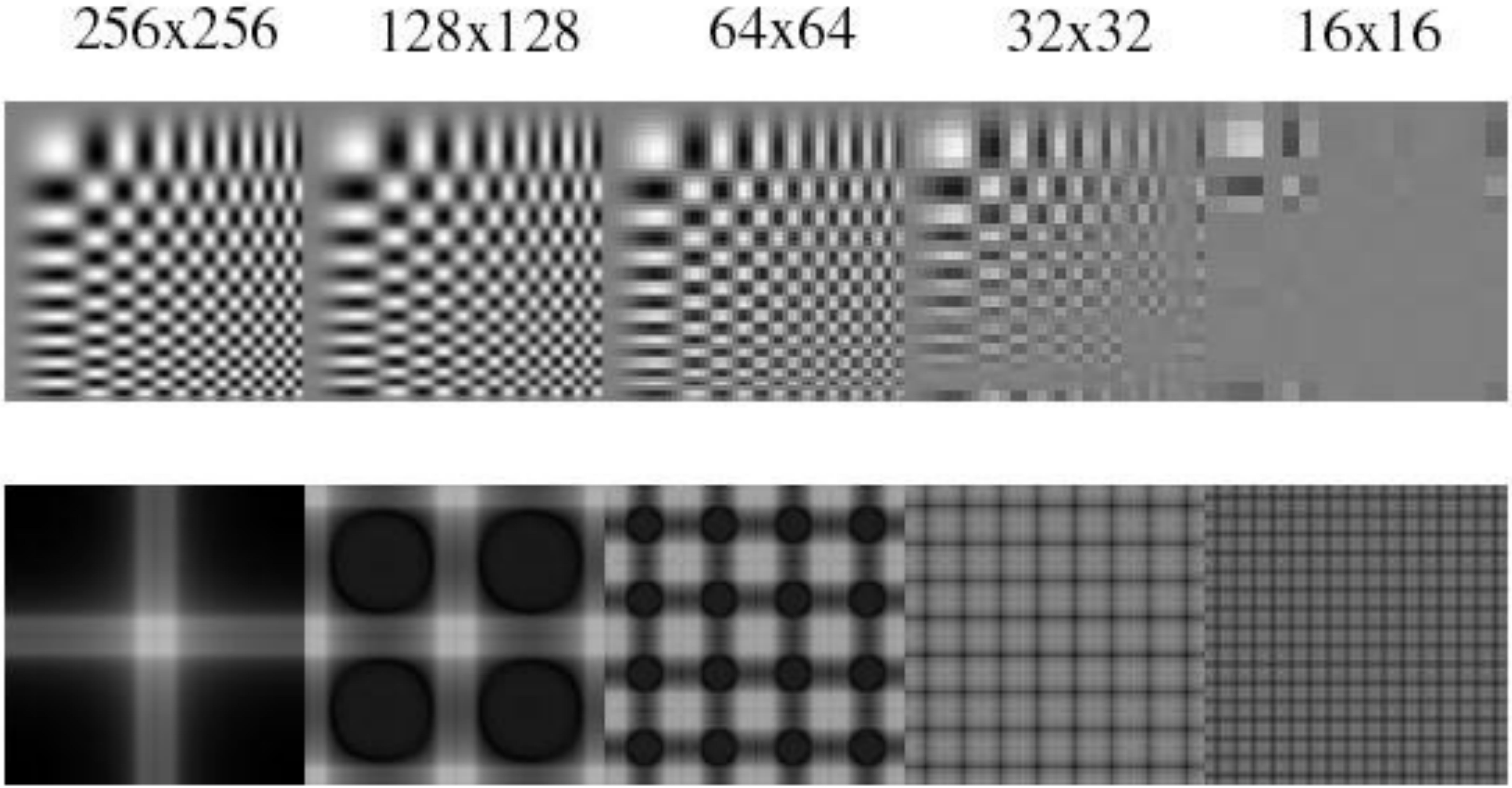
- Solution: suppress high frequencies before sampling
  - multiply the FT of the signal with something that suppresses high frequencies
  - or convolve with a low-pass filter
- Common solution: use a Gaussian
  - multiplying FT by Gaussian is equivalent to convolving image with Gaussian.

How can we represent an image at multiple scales?

Sampling without smoothing. Top row shows the images, sampled at every second pixel to get the next; bottom row shows the magnitude spectrum of these images.



Sampling with smoothing. Top row shows the images. We get the next image by smoothing the image with a Gaussian with sigma 1 pixel, then sampling at every second pixel to get the next; bottom row



Representation of scale

Representation of scale

The Gaussian pyramid

- Smooth with gaussians, because
  - a gaussian*gaussian=another gaussian
- Synthesis
  - smooth and subsample
- Analysis
  - Start with the top image (coarse) and move to lower (fine) image layers

Applications

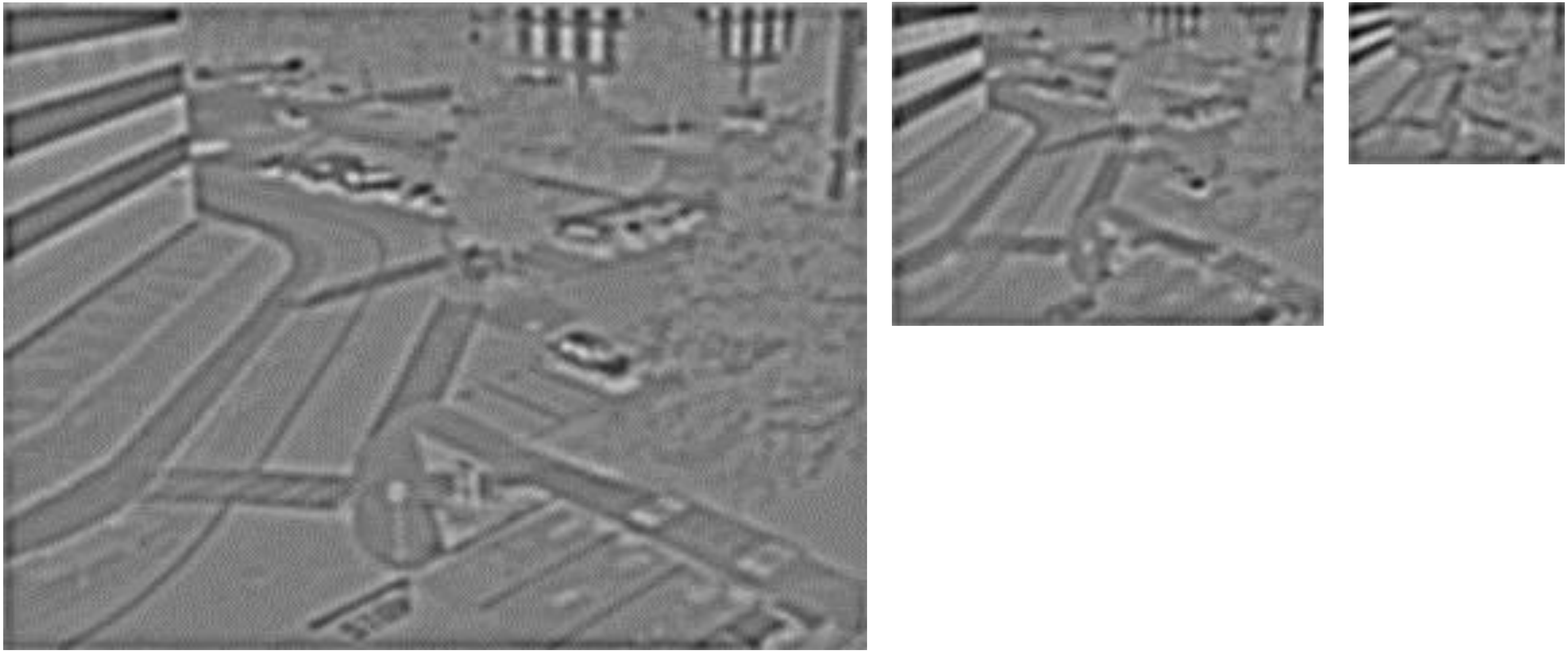
- Search for correspondence
  - look at coarse scales, then refine with finer scales
  - Lowers computational cost
- Edge tracking
- a “good” edge at a fine scale has parents at a coarser scale

512 256 128 64 32 16 8

Laplacian Pyramids

- Given input I
- Construct Gaussian pyramid $I^G{}_1,..,I^G{}_n$
- Take the difference between consecutive levels:
- $I^L{}_s = I^G{}_s - I^G{}_{s-1}$
- Image $I^L{}_s$ is an approximation of the Laplacian at scale number s
  - Laplacian is a band-pass filter: Both high frequencies (edges and noise) and low frequencies (slow variations of intensity across the image)

Laplacian Pyramids

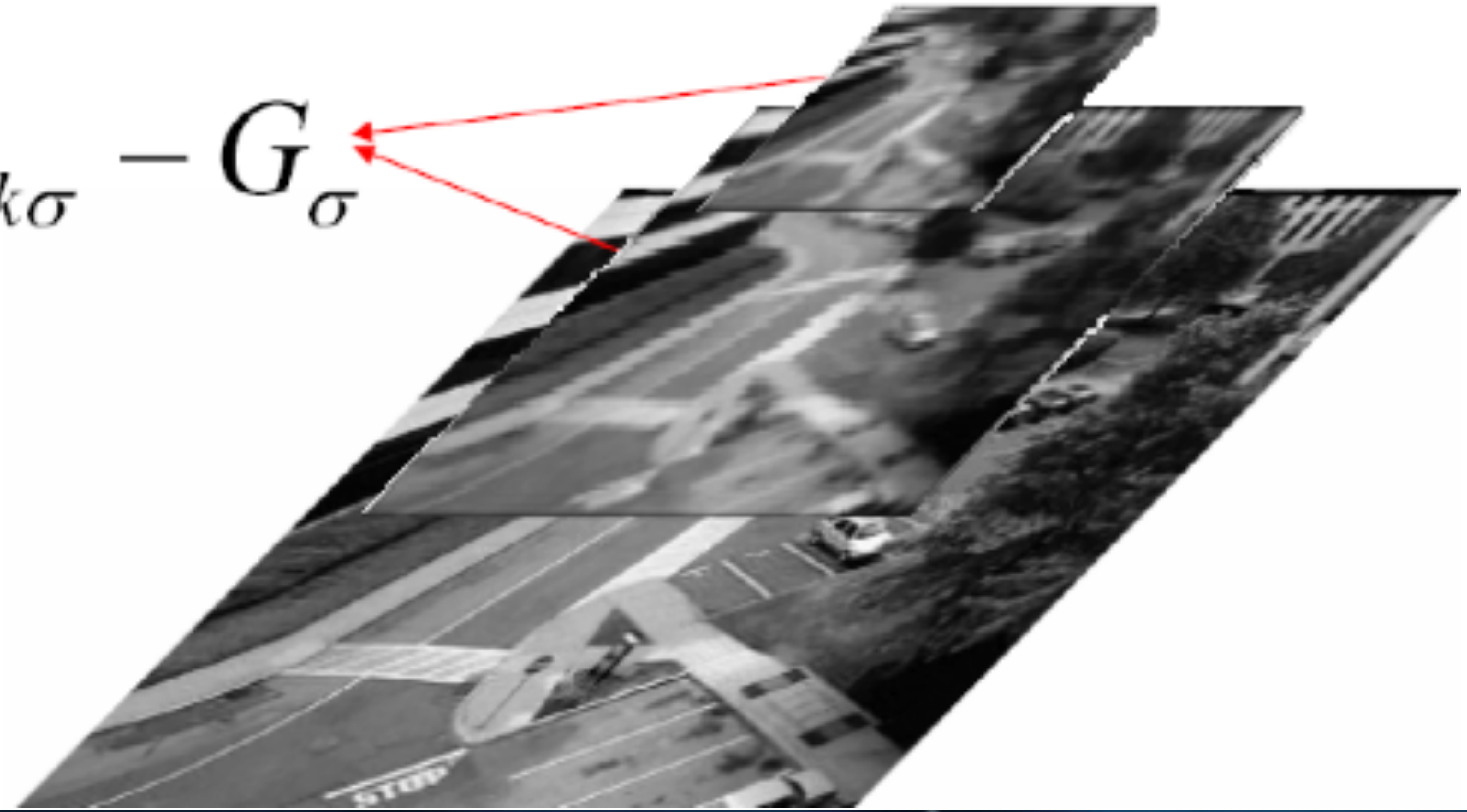


$$\boxed{\nabla_\sigma^2 I = \nabla^2 G_\sigma * I}$$

Constant independent of σ

$$\boxed{G_{k\sigma} - G_\sigma \approx (k-1)\sigma^2 \nabla^2 G_\sigma}$$

$$G_{k\sigma} - G_\sigma$$

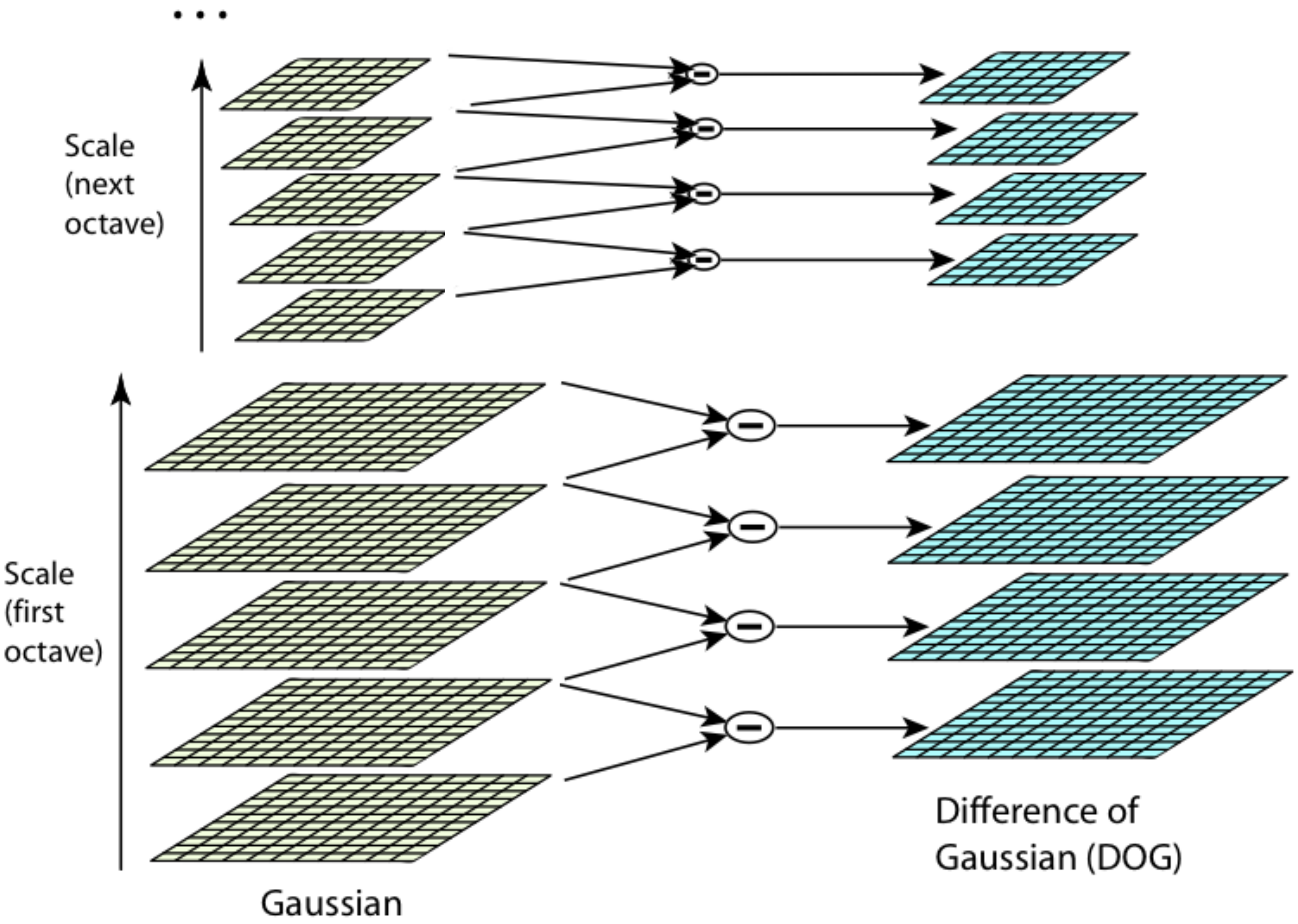


The Laplacian of a Gaussian can be approximated by the difference of two Gaussians.

To compare the Laplacian at different scales we need to explain more carefully what the approximation is.

In practice: the scaled Laplacian can be computed by taking the difference between level in a Gaussian pyramid.

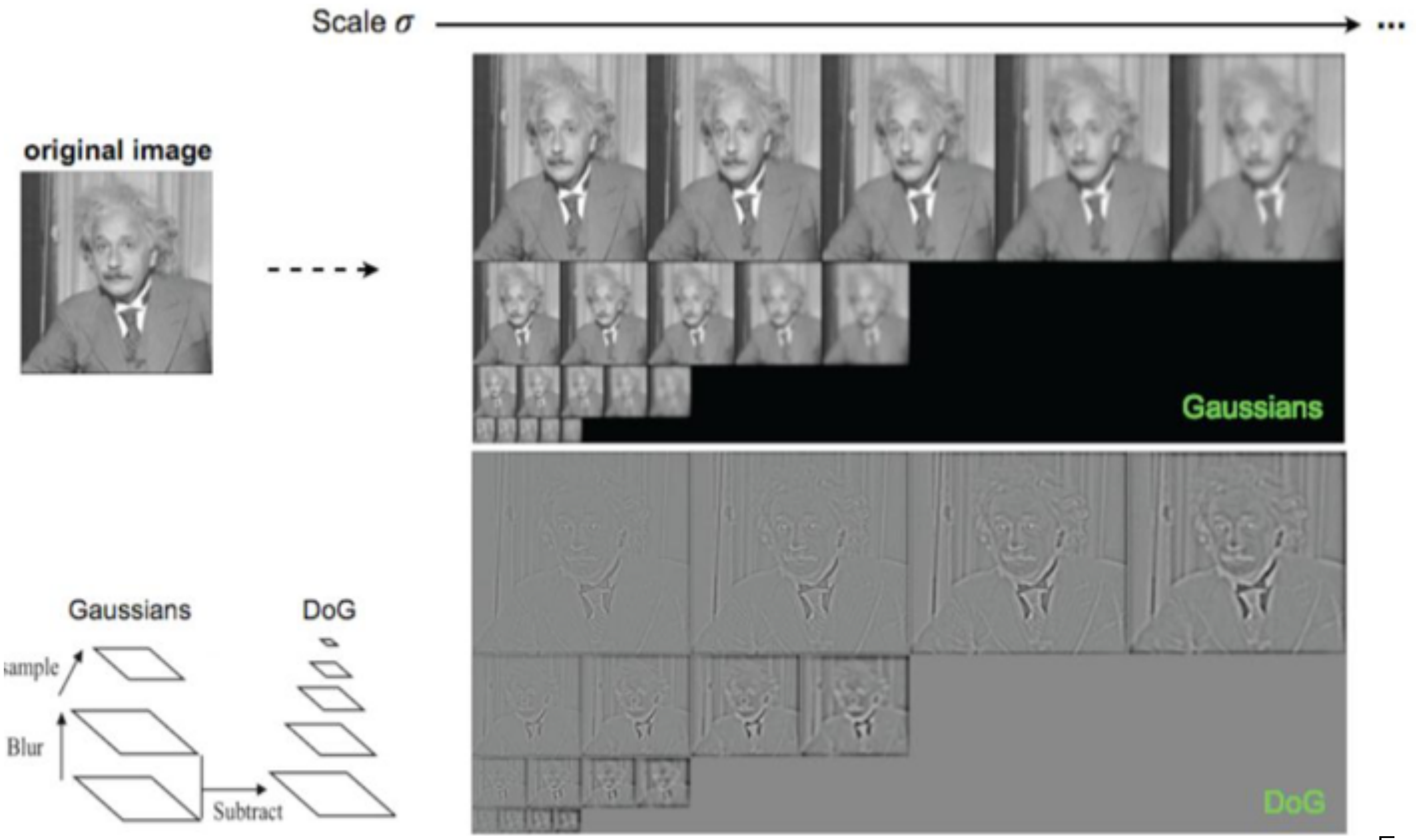
Laplacian Pyramid in Practice



$$\sigma = k^n \sigma_0$$

$$k = 2^{\frac{1}{s}}$$

Example



Example from J.Corso

Measuring frequency content

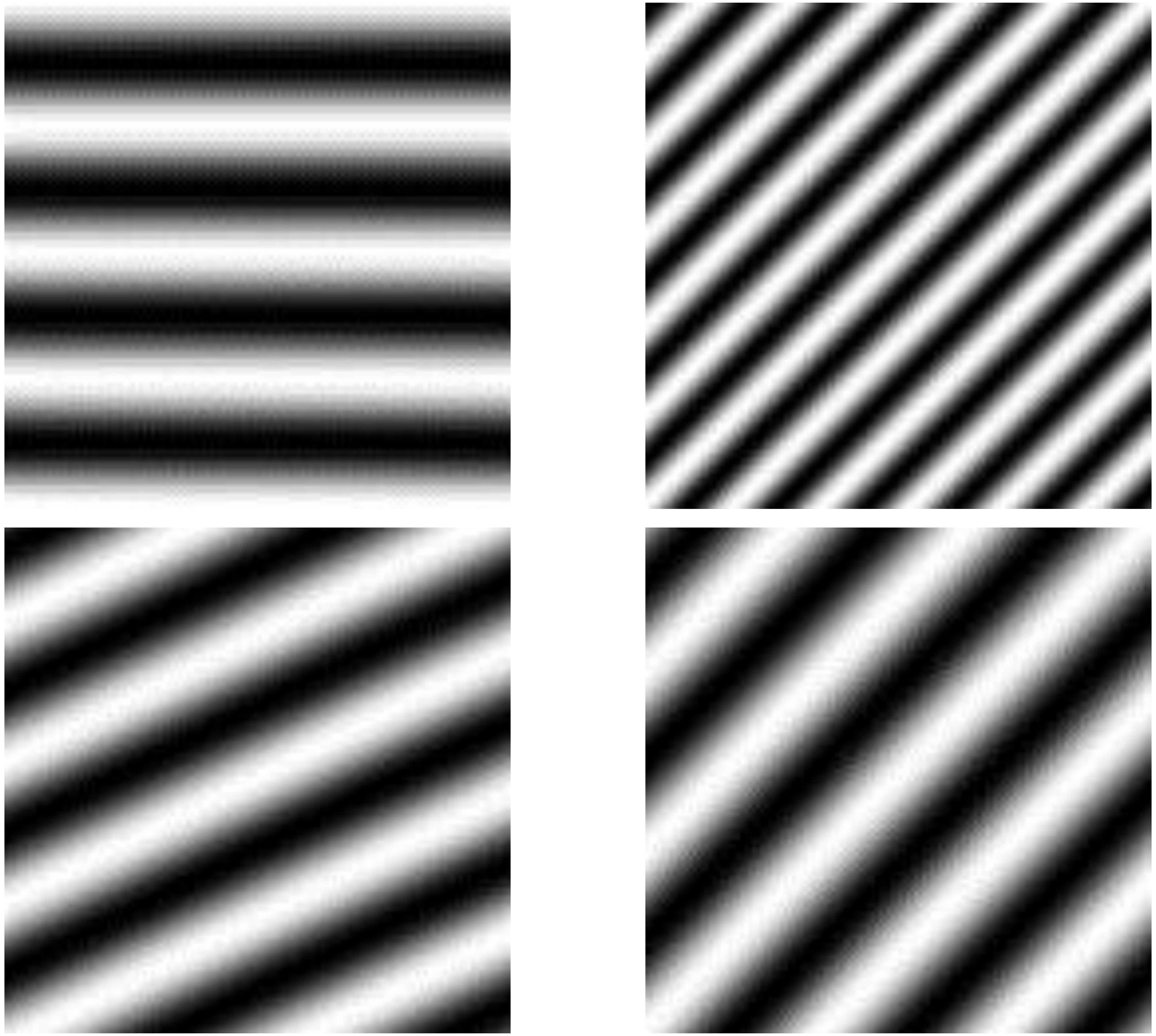
A representation for image changes

- We need a change of basis to move from pixel intensities (space domain) to frequency domain.

“dot”

= A measure of image content at this frequency and orientation

2D Bases…



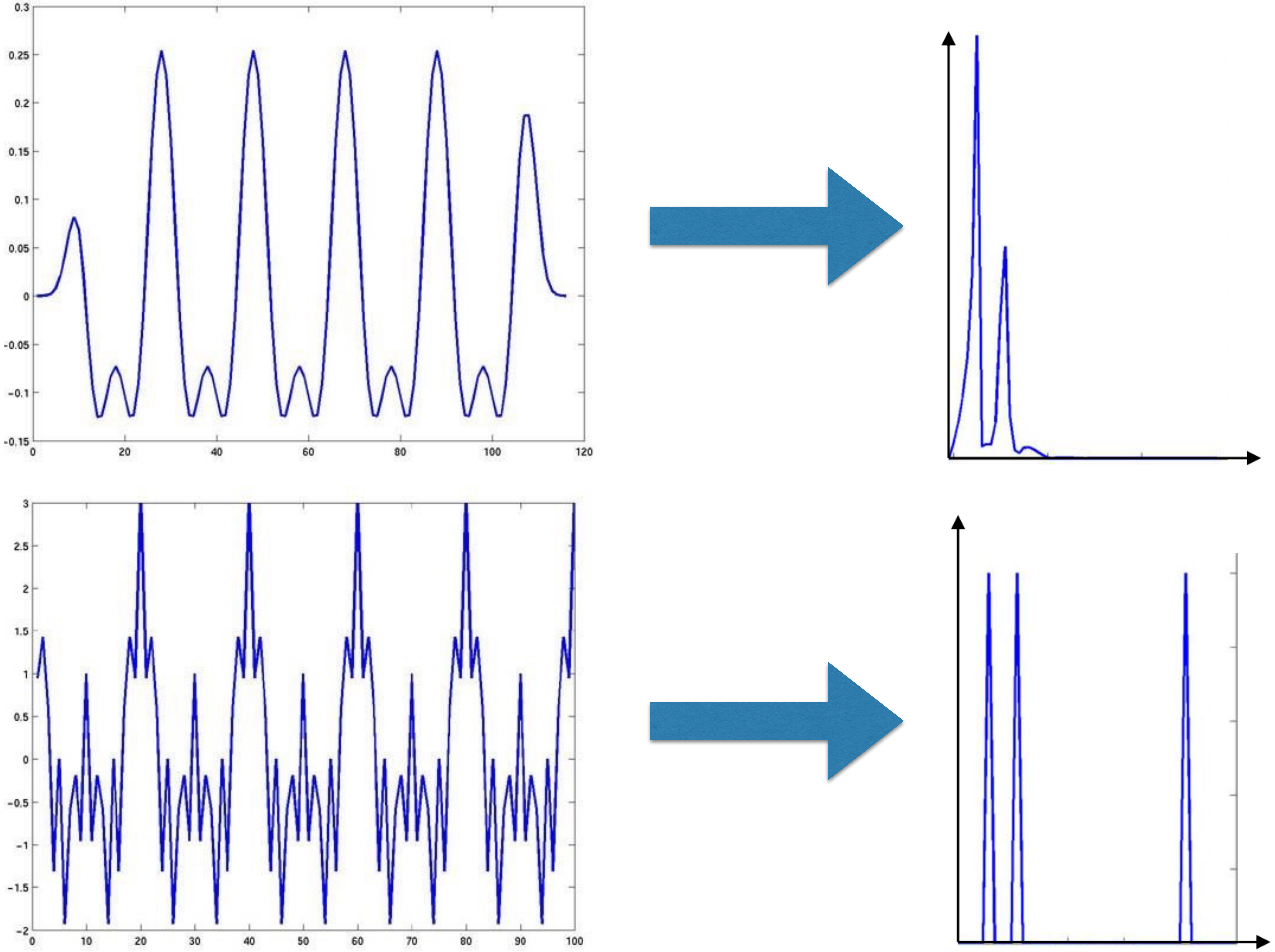
and so on…

More formally

- More formally, content at frequency *u* is obtained by taking the magnitude of the "dot product" with the function
$$\cos(2\pi u)$$
- Same with
$$\sin(2\pi u)$$

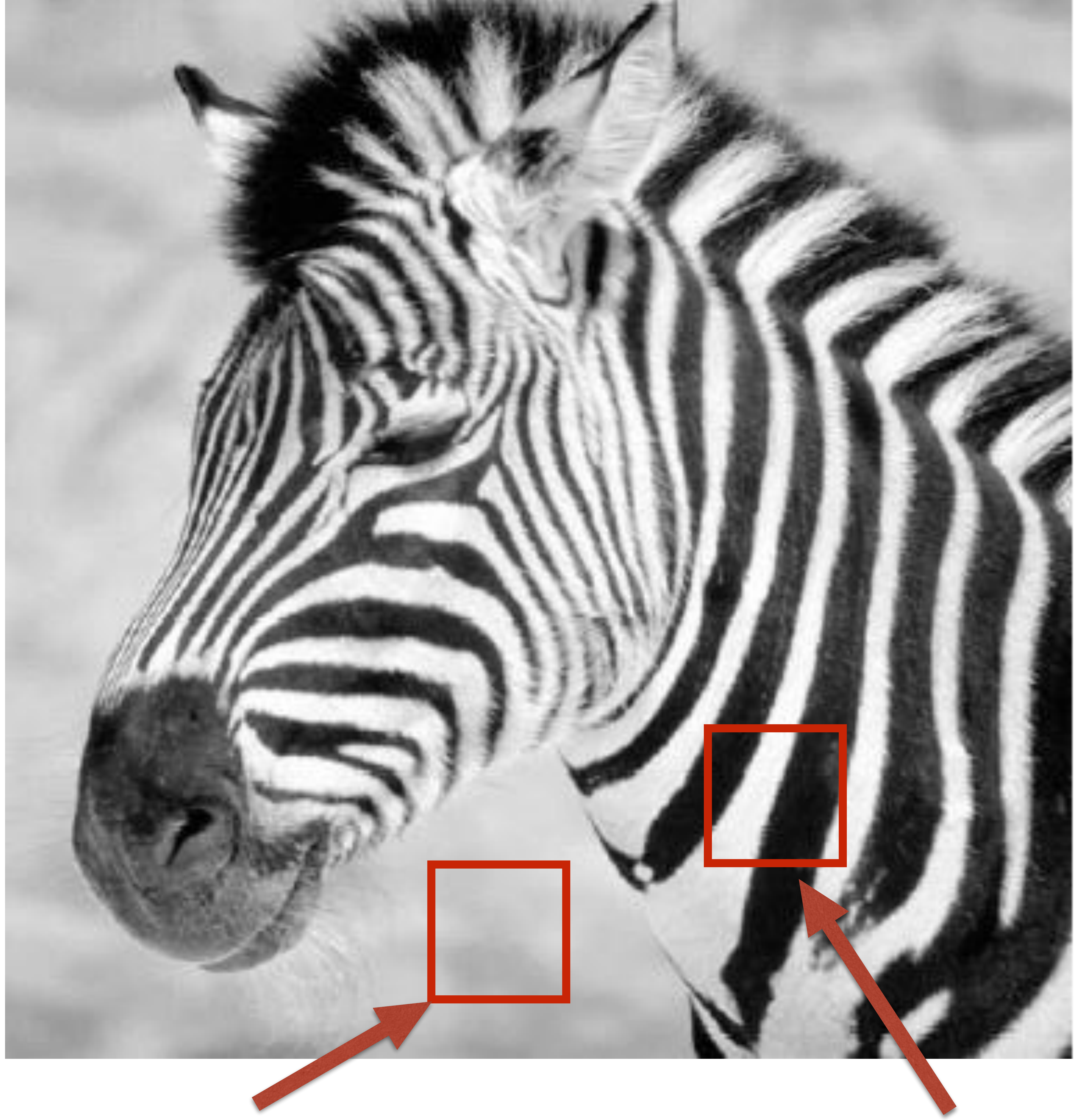
Examples 1D

Signal Domain



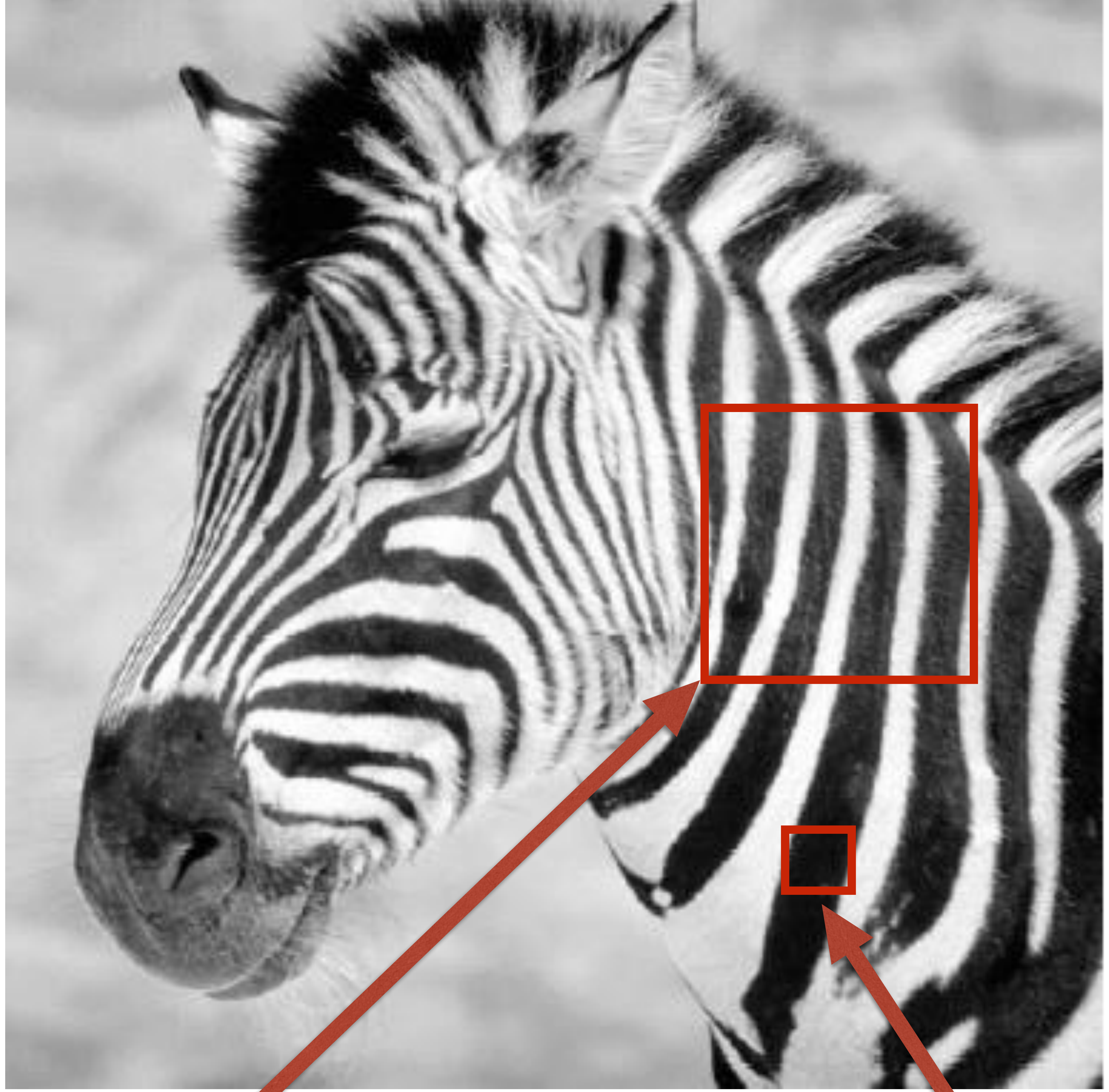
Frequency Domain

Measuring frequency *locally* at *different* scales



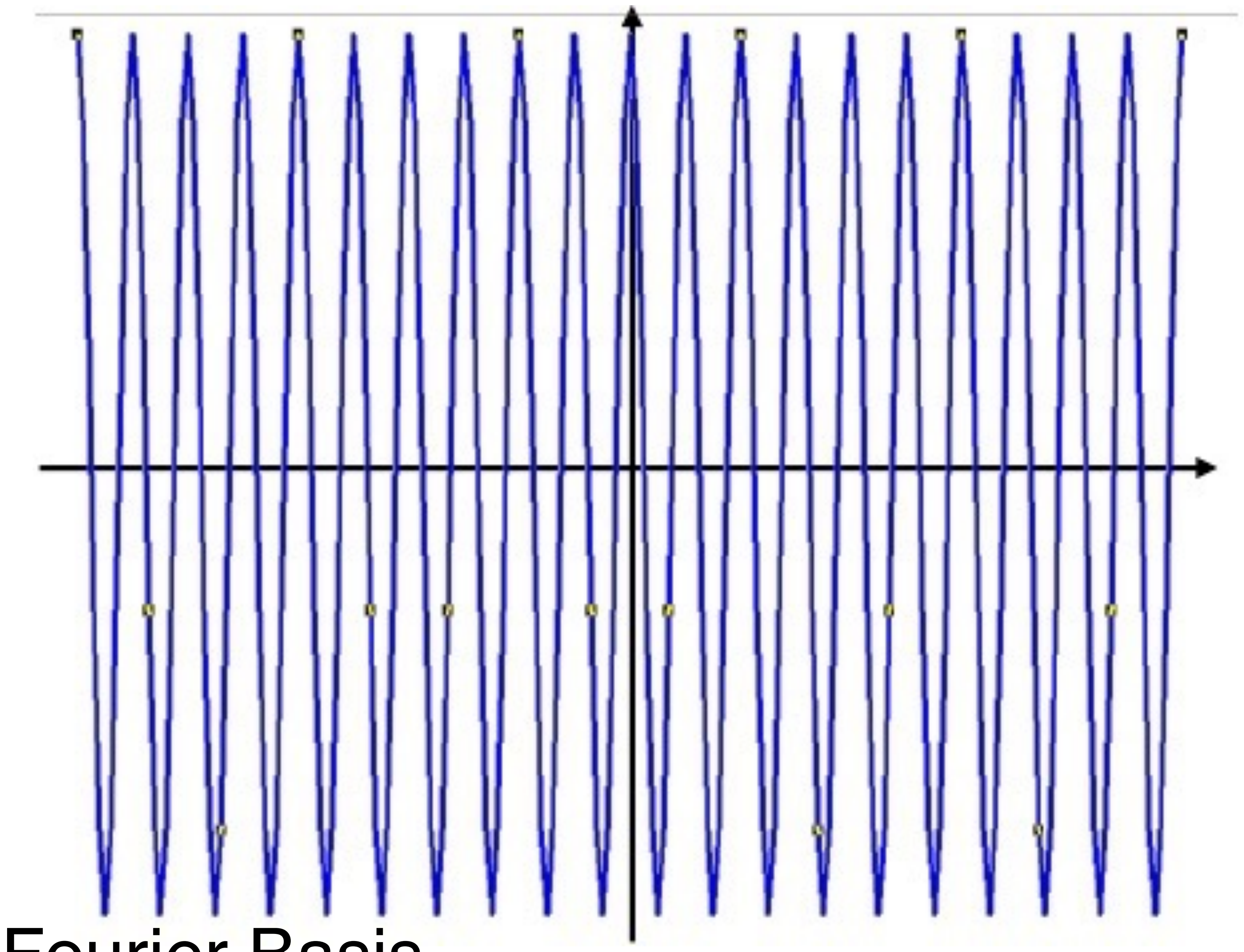
Low frequency content in all directions

High frequency at 45^o orientation



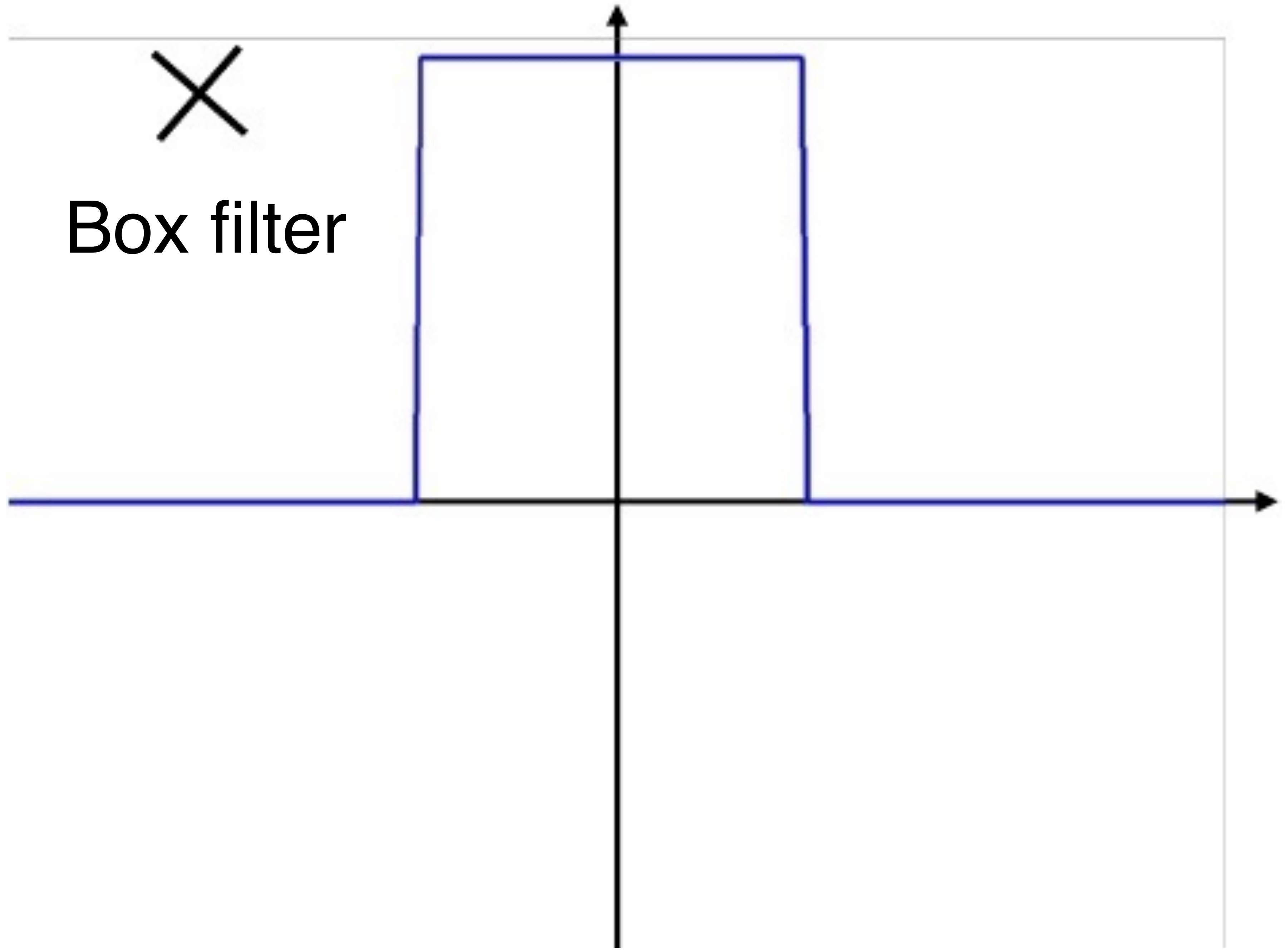
Higher frequency
content
at larger scale

Low frequency
(uniform)



Fourier Basis

×

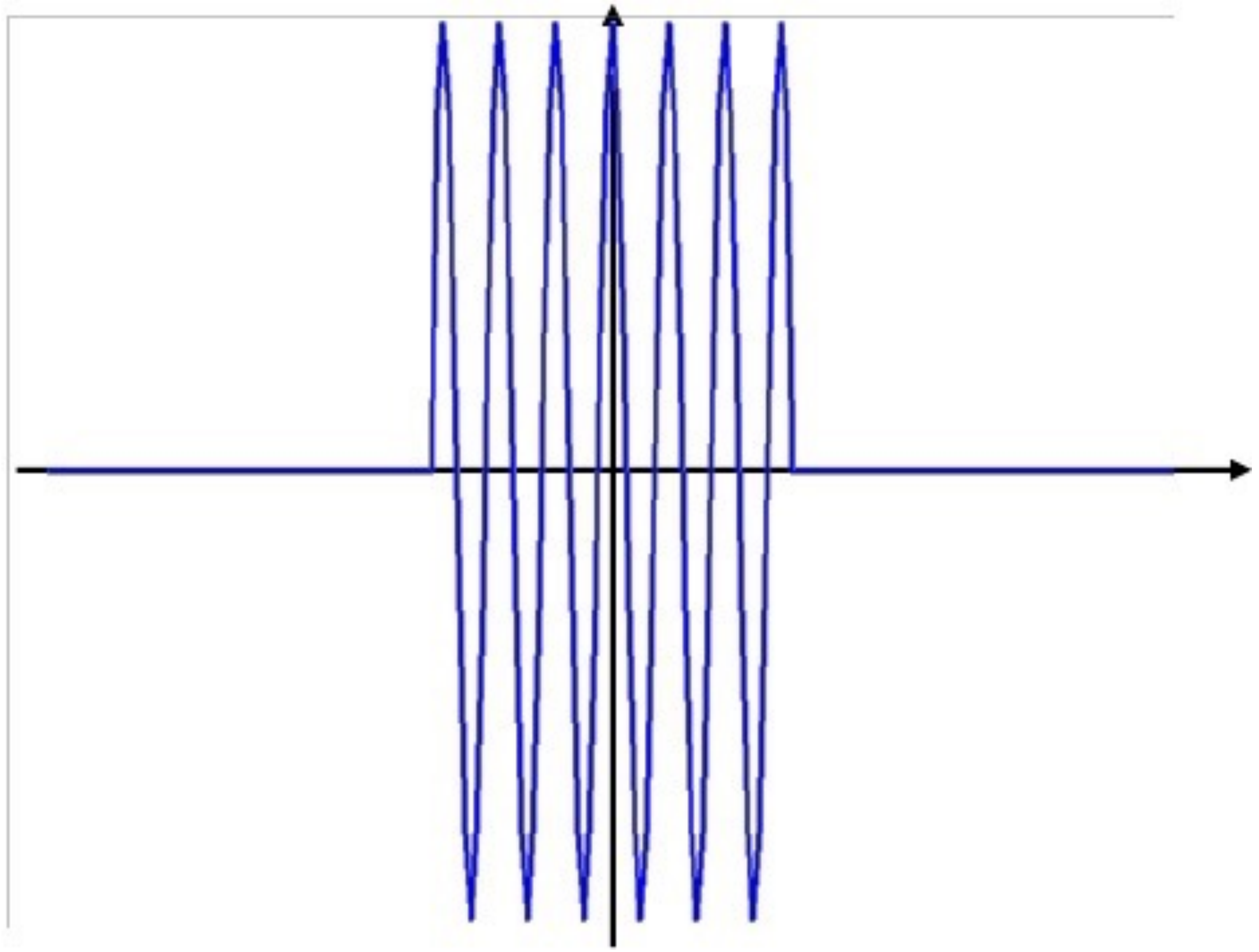


Box filter

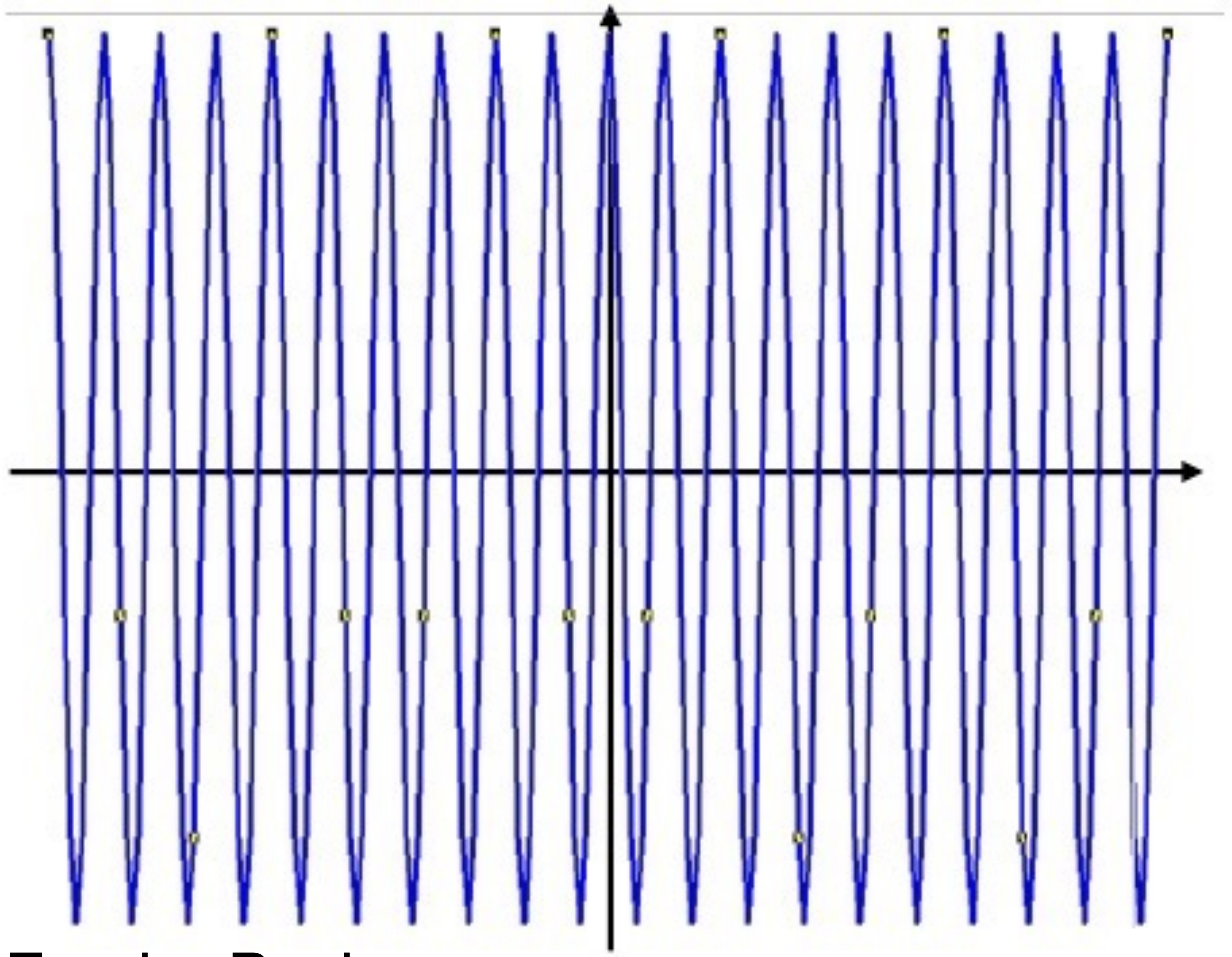
=

Frequency content
in local neighborhood
at every point in image

$I' = I * \text{STFT}$



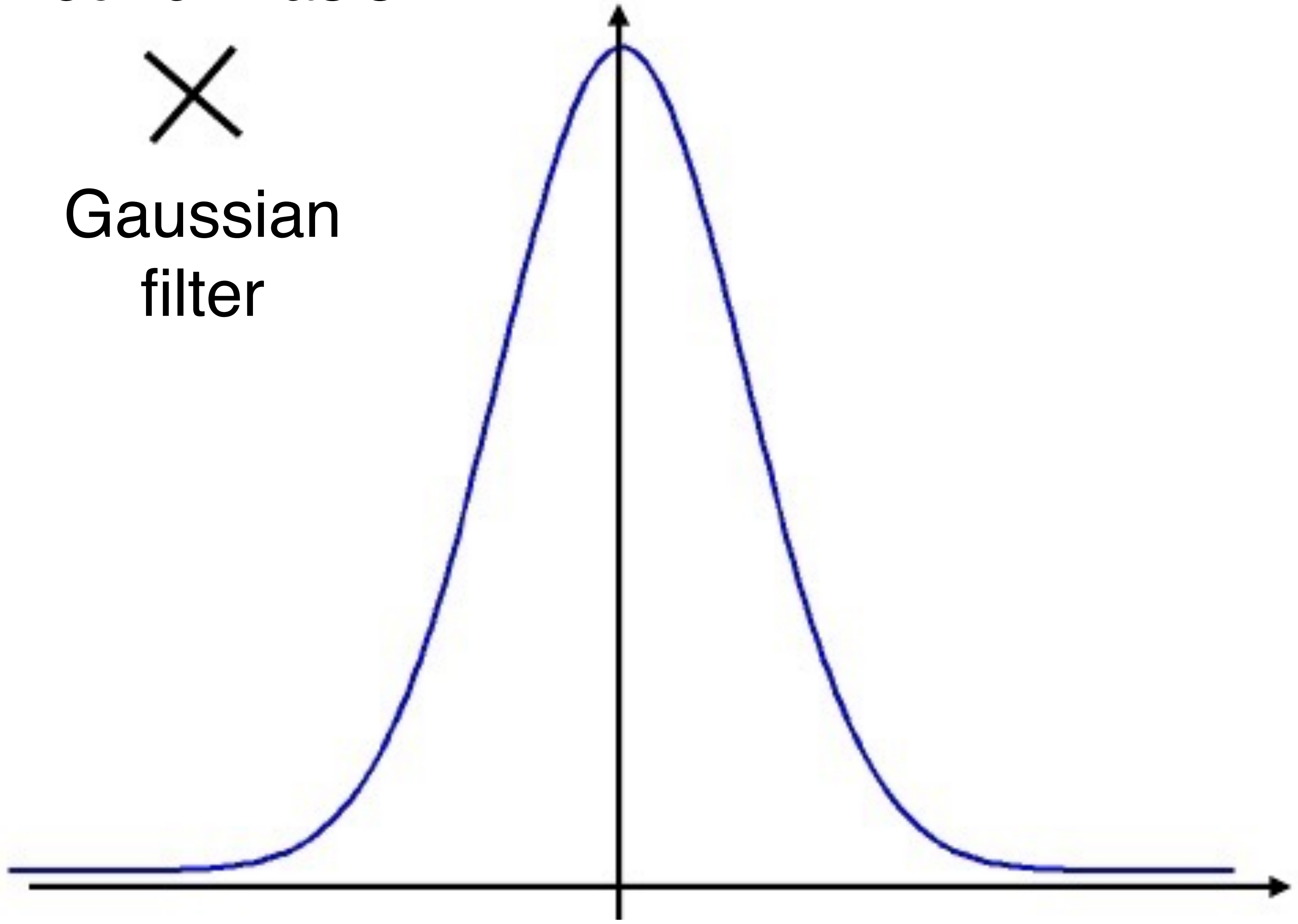
Short Time Fourier Transform
(STFT)



Fourier Basis

$\times$

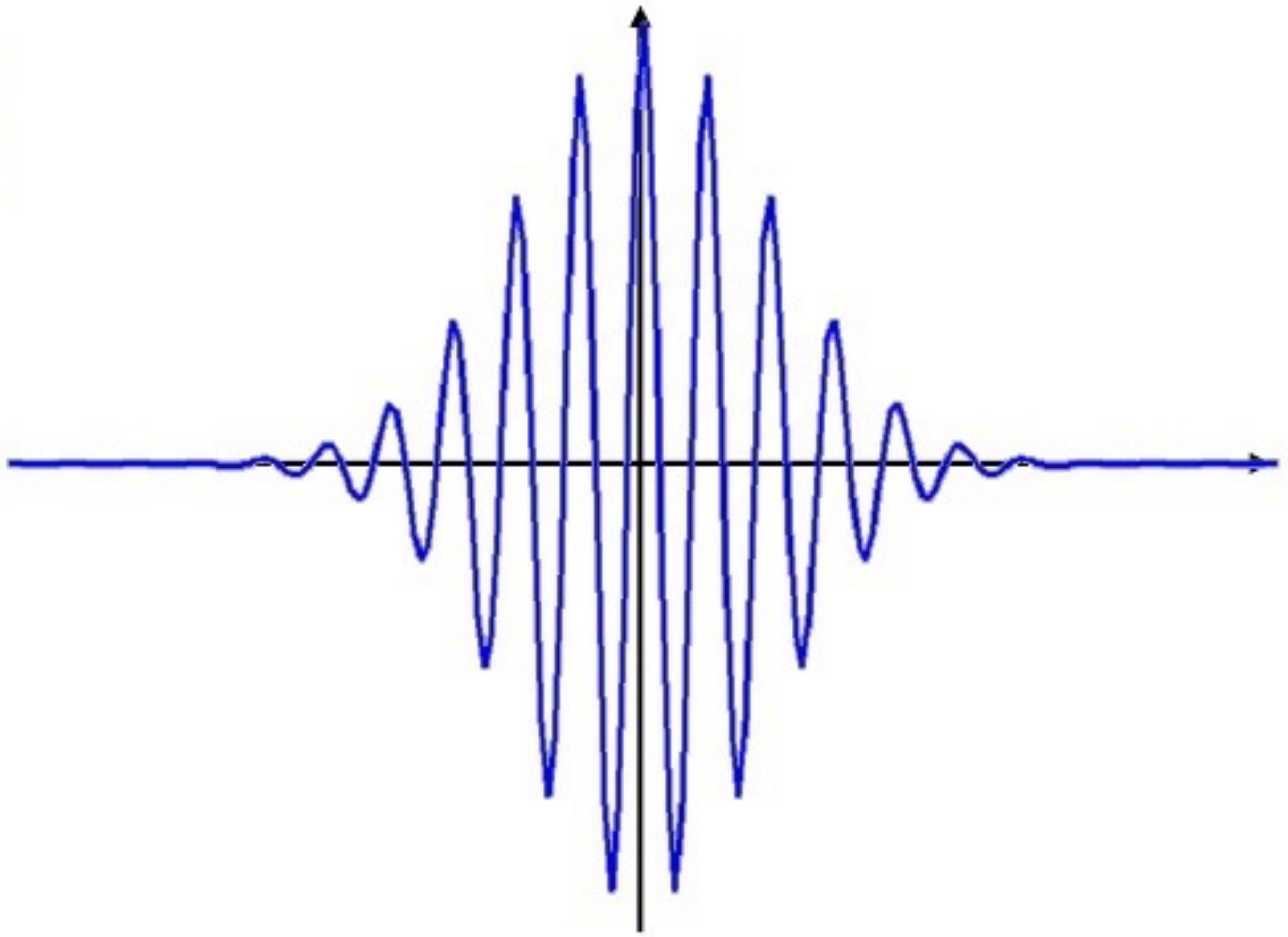
Gaussian filter



=

Frequency content
in local neighborhood
at every point in image

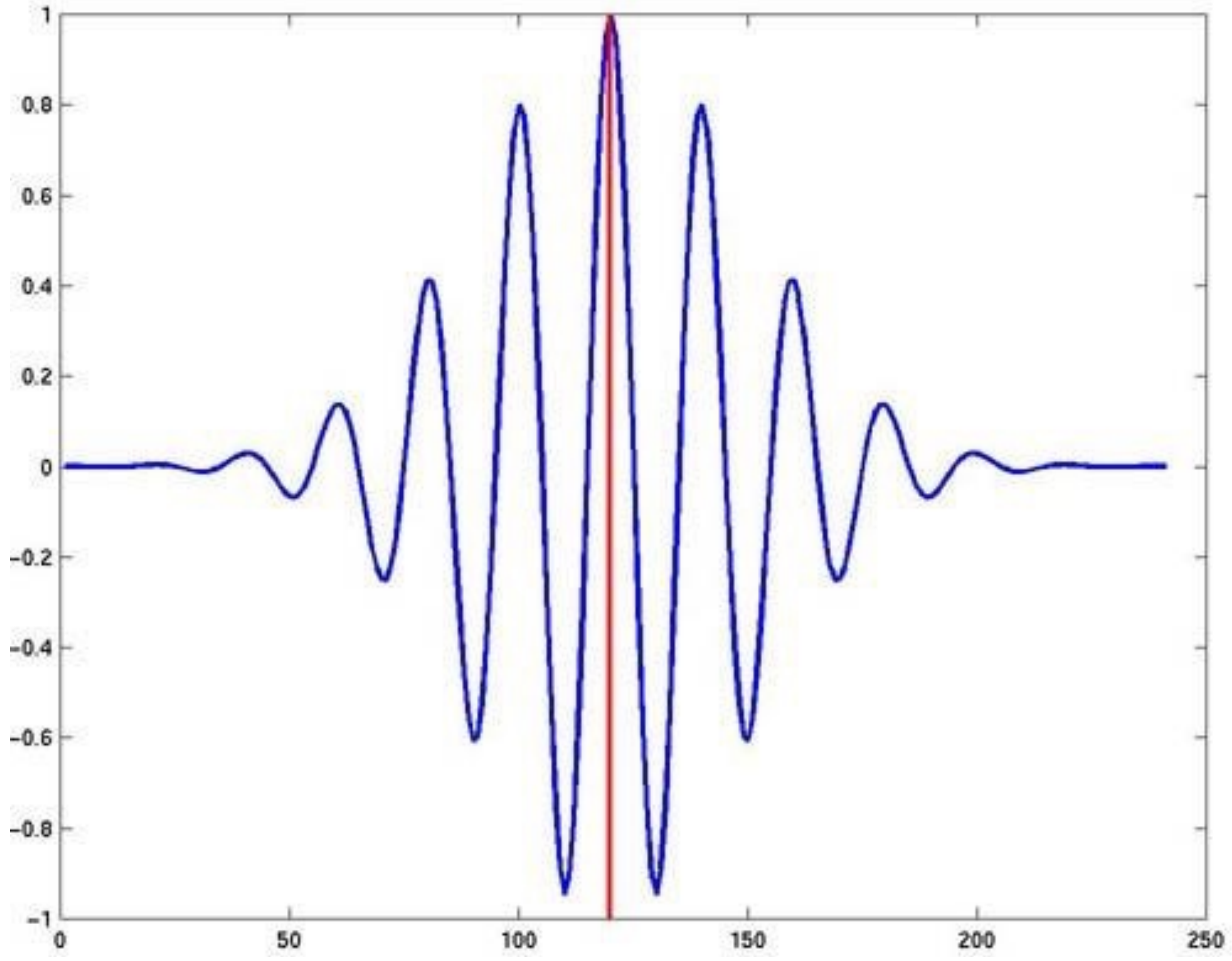
$I' = I * \text{Gabor}$



Gabor Filter

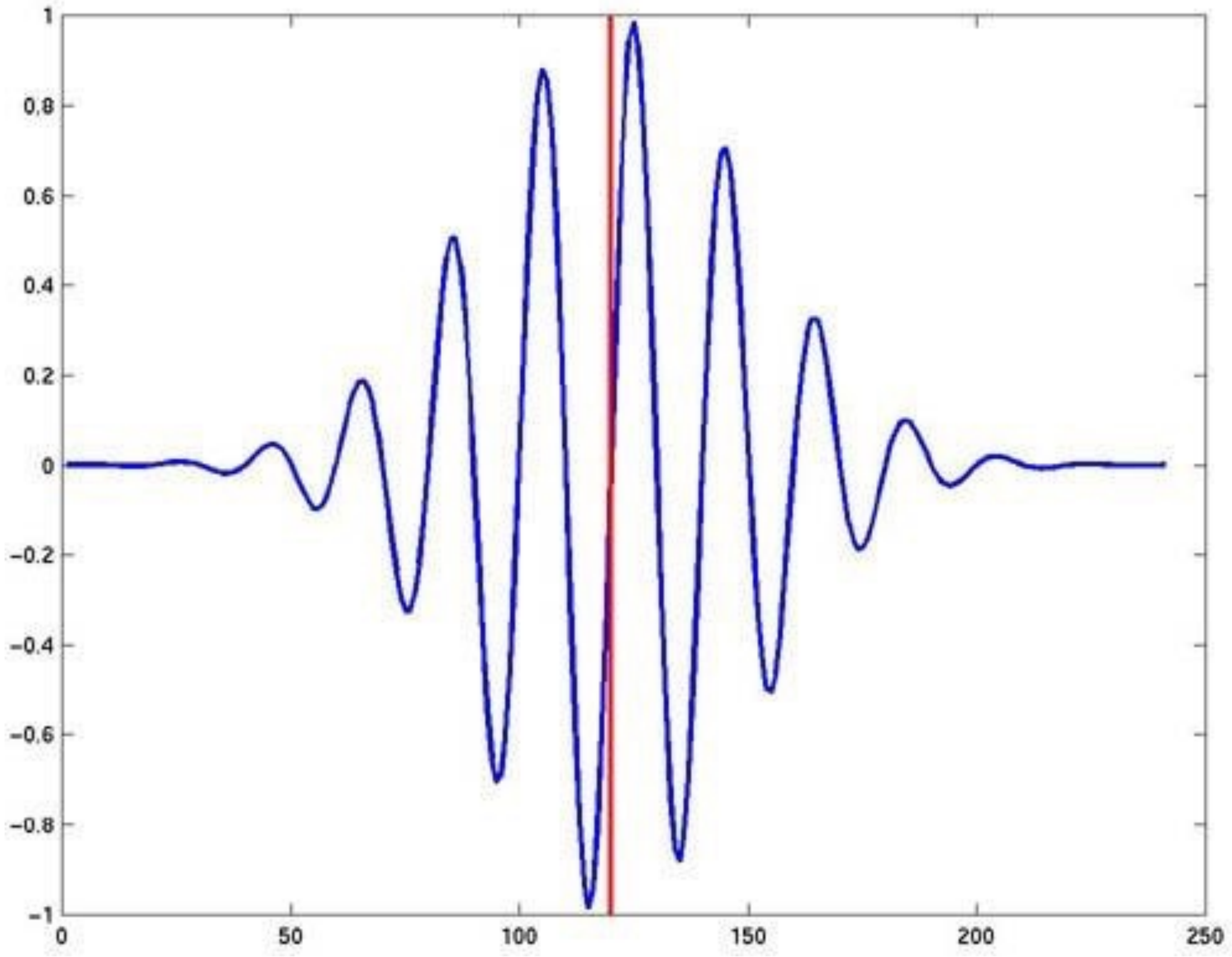
Gabor Filter Example

Odd (antisymmetic)



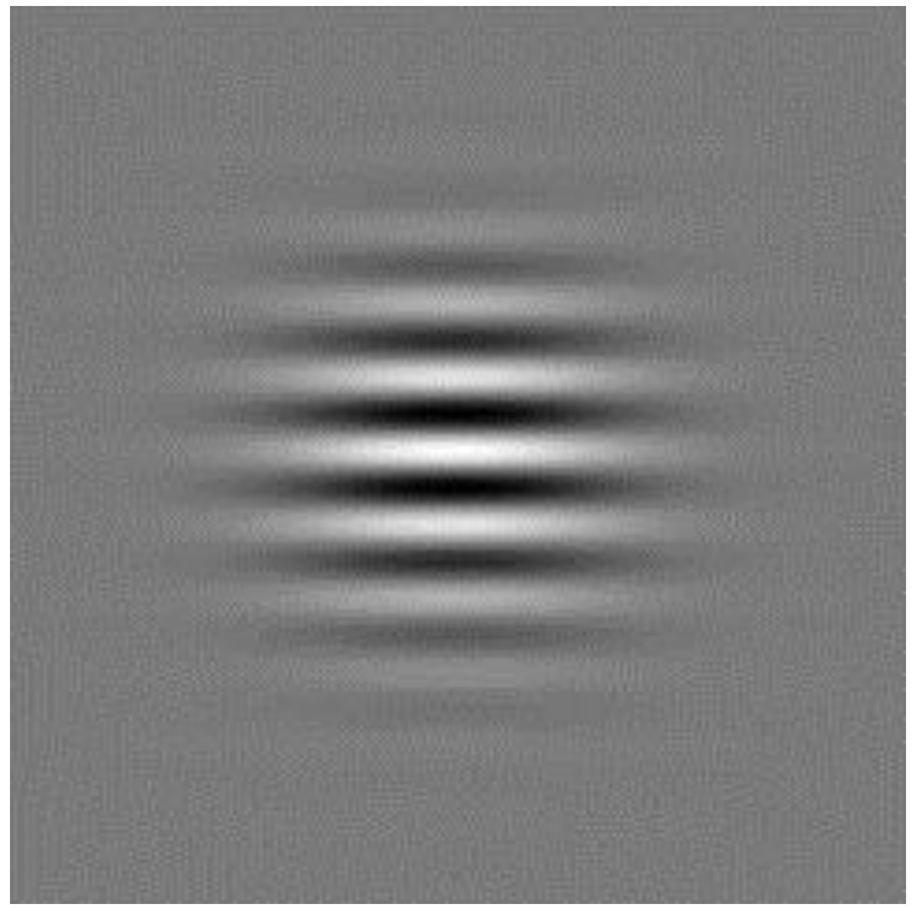
$$e^{-\frac{x^2}{2\sigma^2}} \sin(2\pi\omega x)$$

Even (Symmetric)

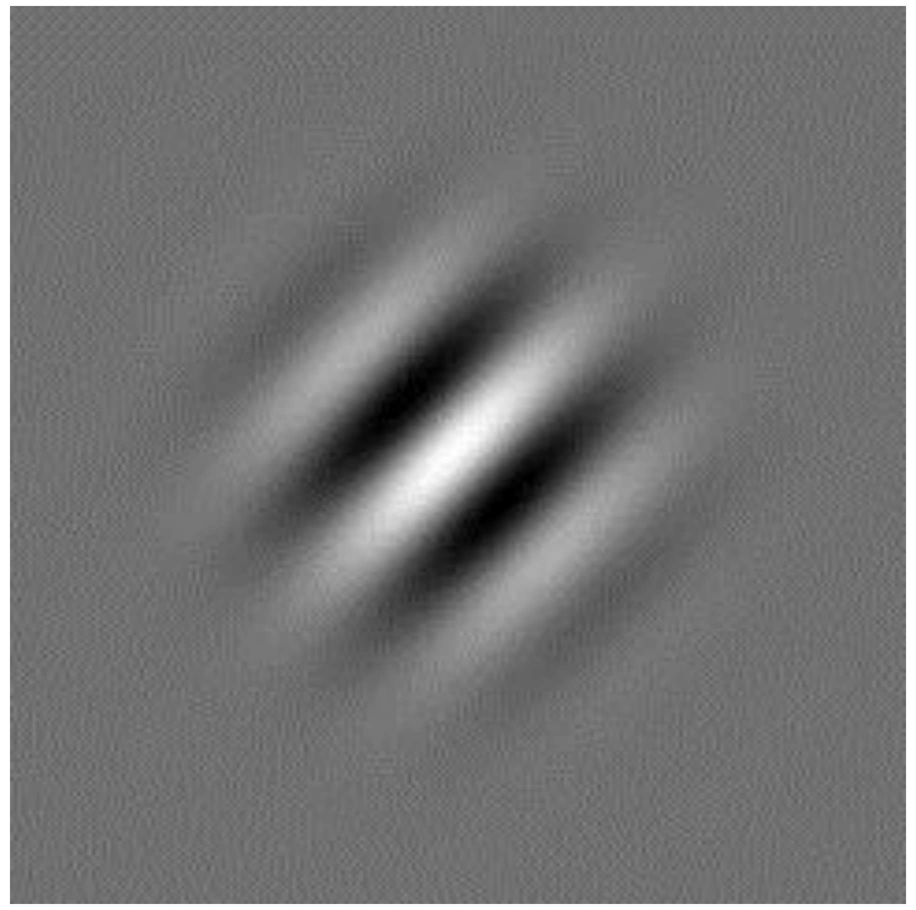


$$e^{-\frac{x^2}{2\sigma^2}} \cos(2\pi\omega x)$$

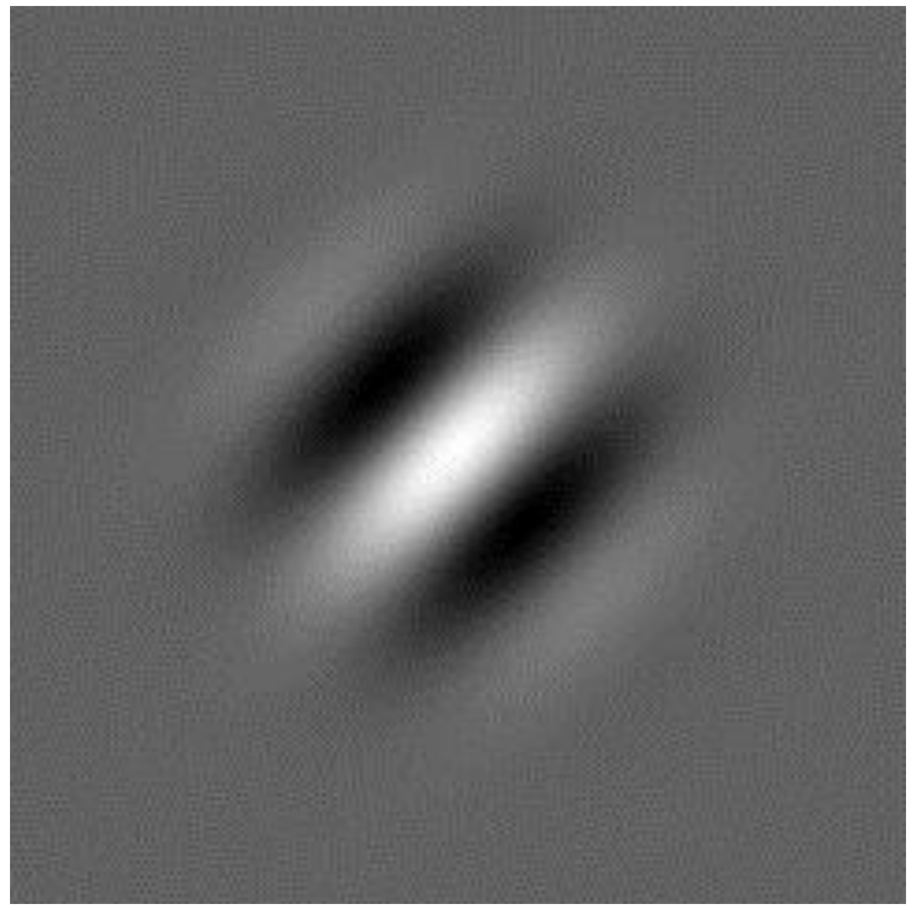
$$e^{-\frac{x^2+y^2}{2\sigma^2}} \cos(2\pi(k_x x + k_y y))$$



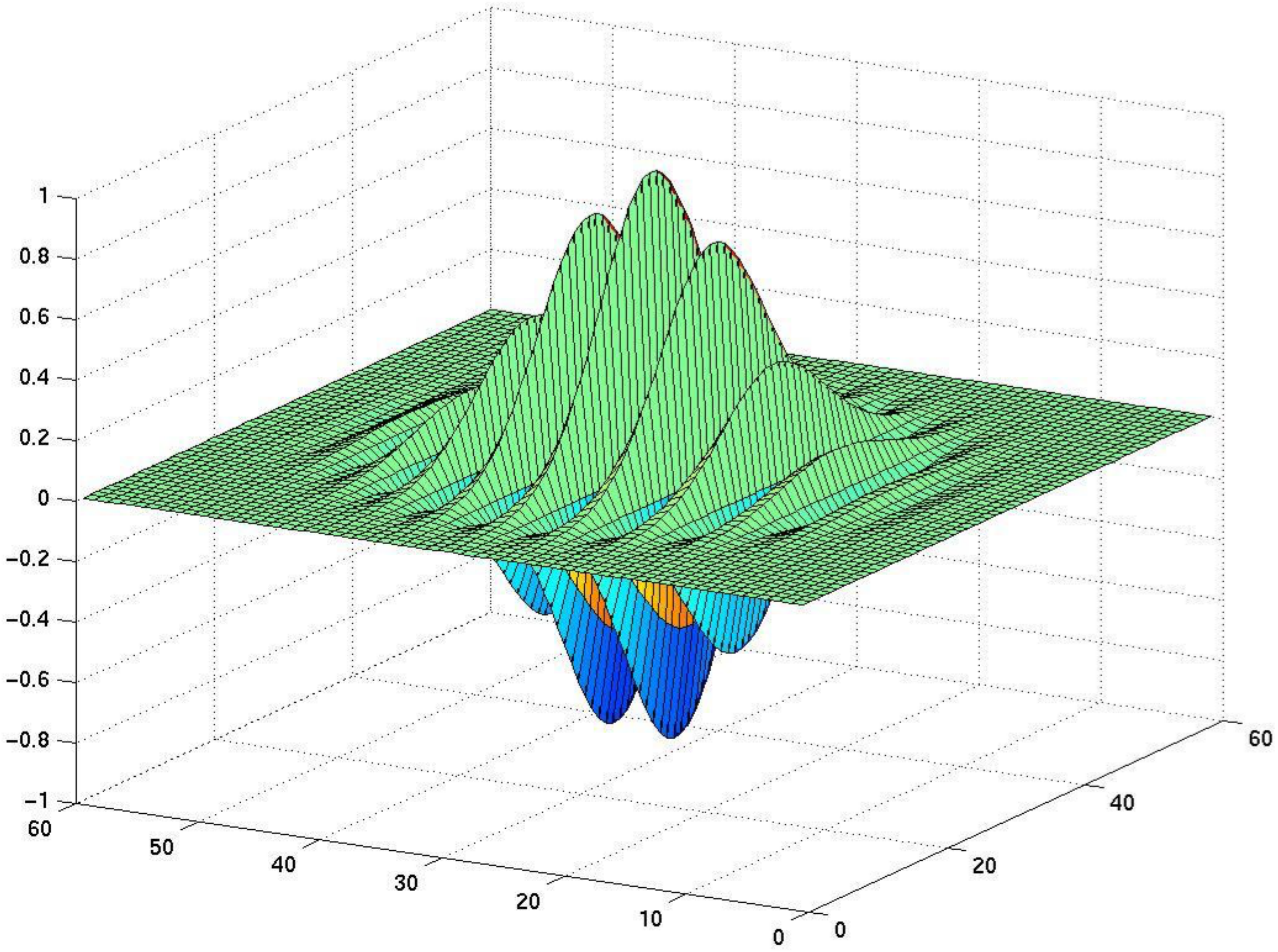
High frequency along axis

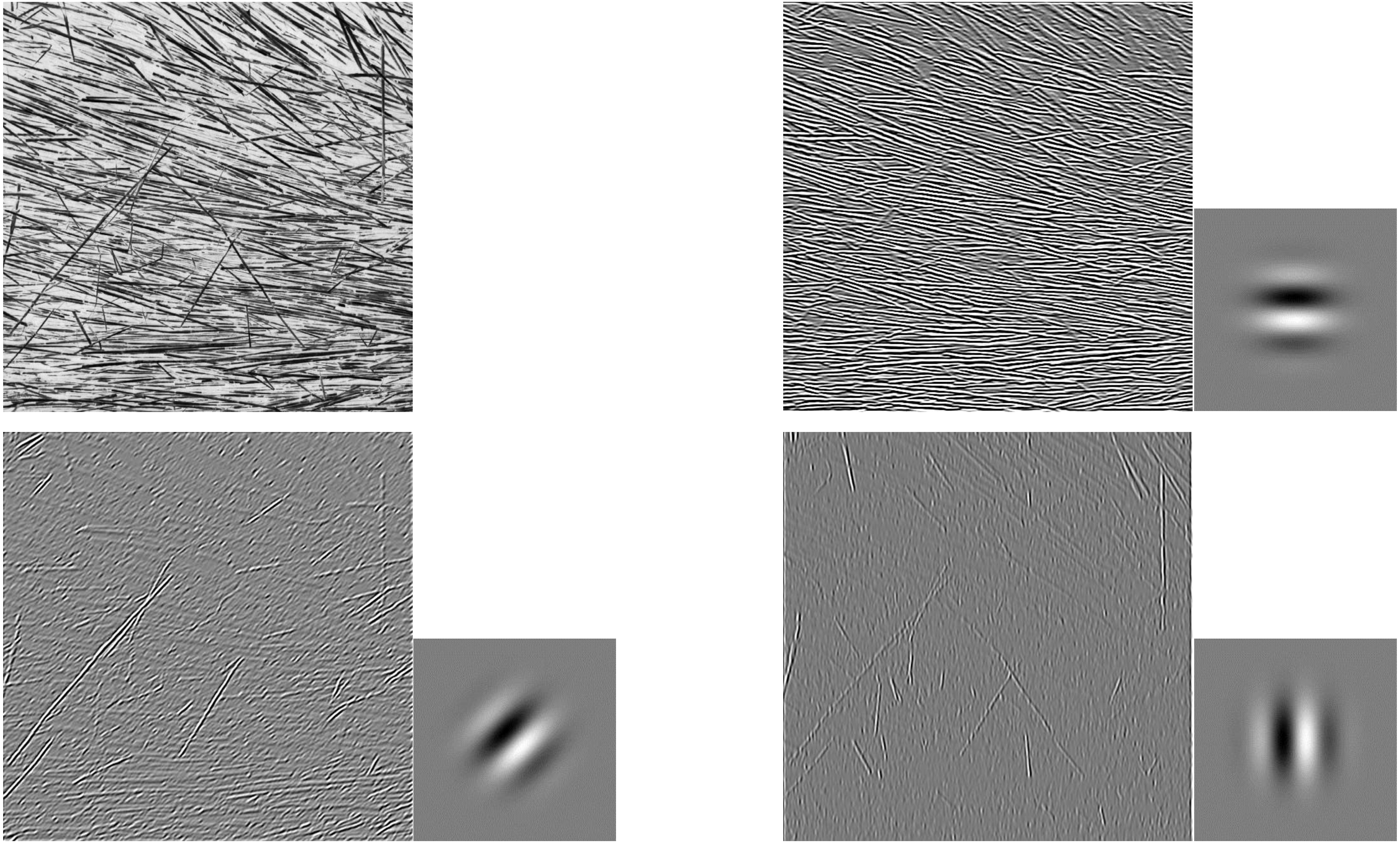


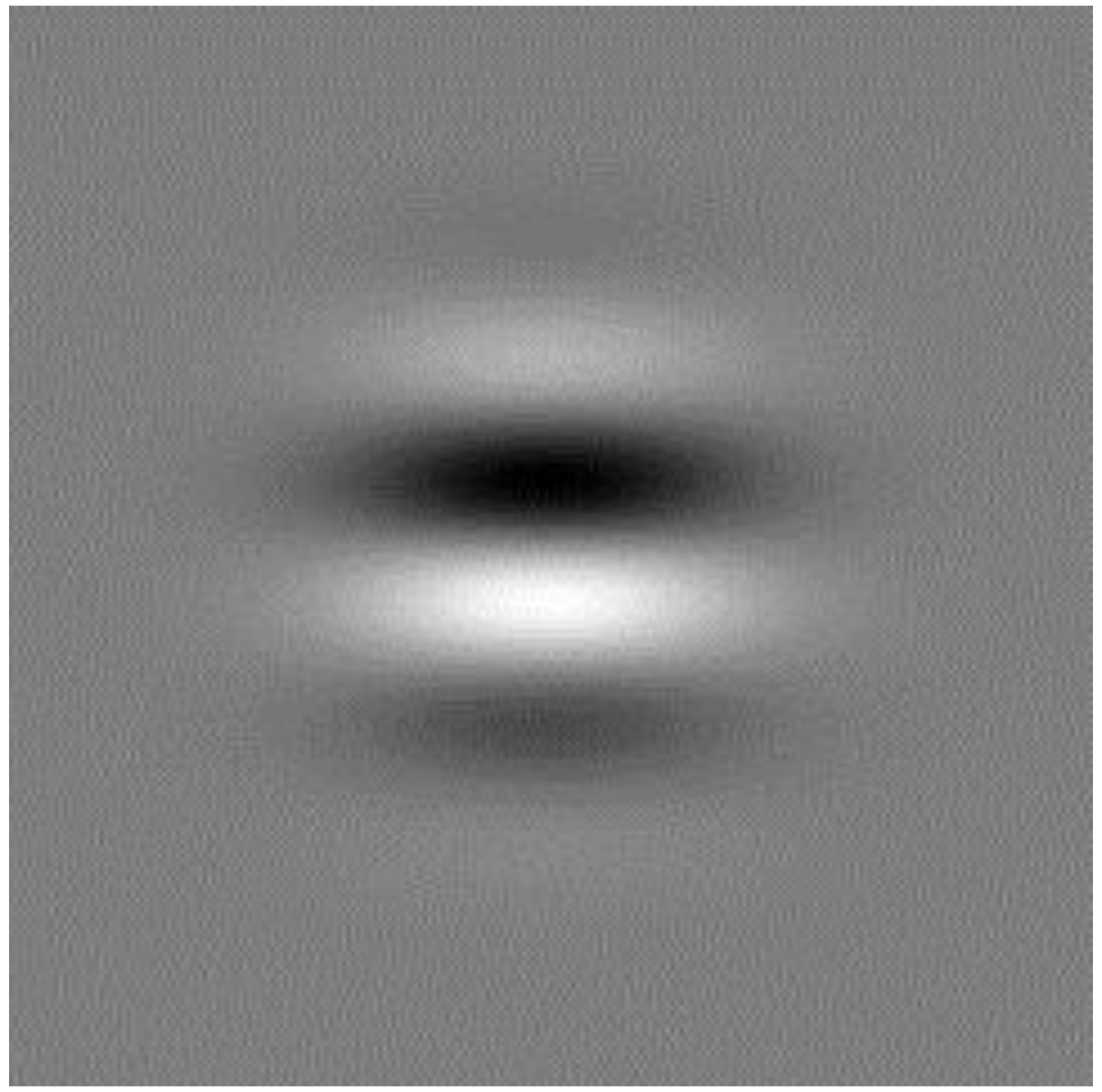
Lower frequency along diagonal



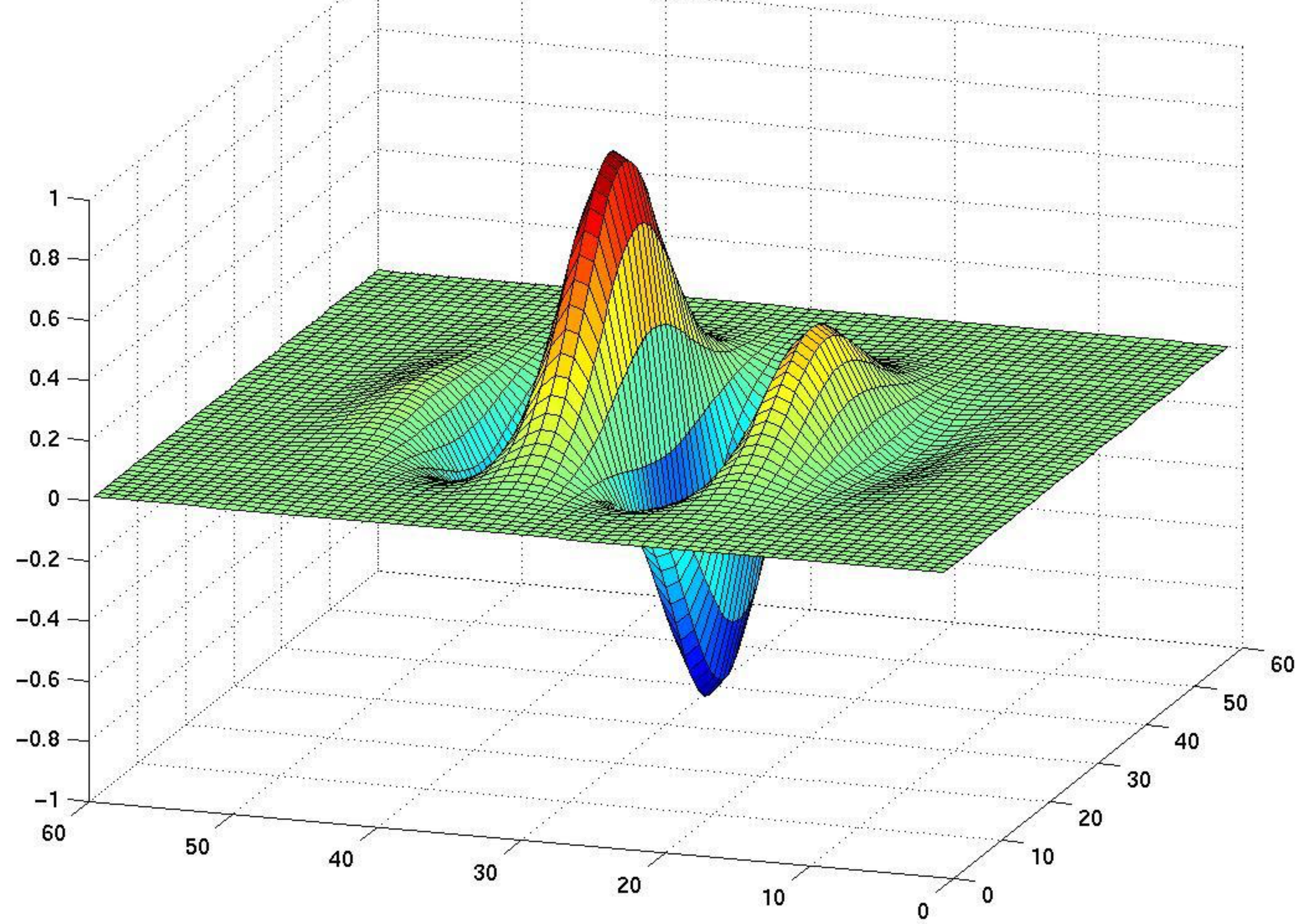
Even lower frequency

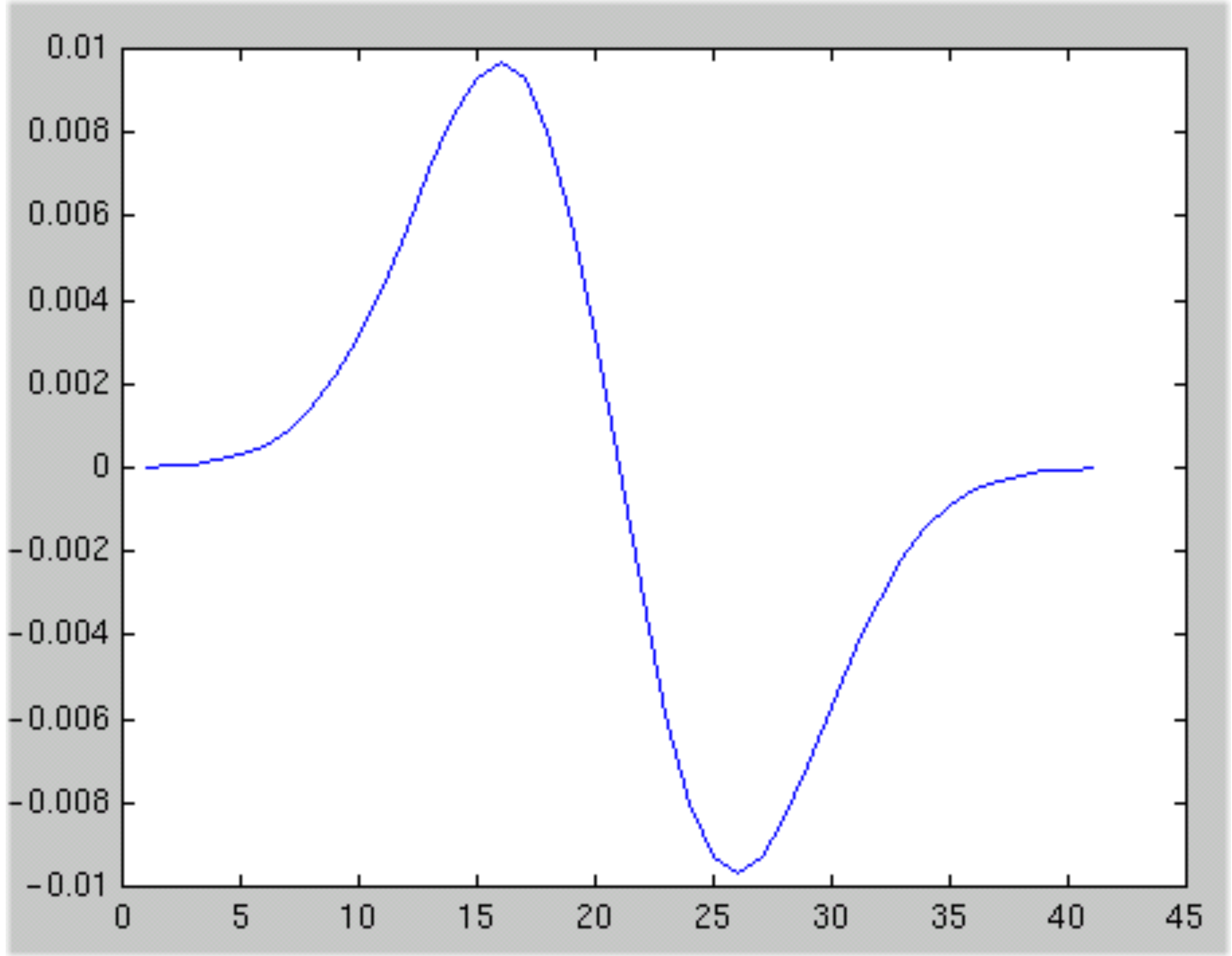




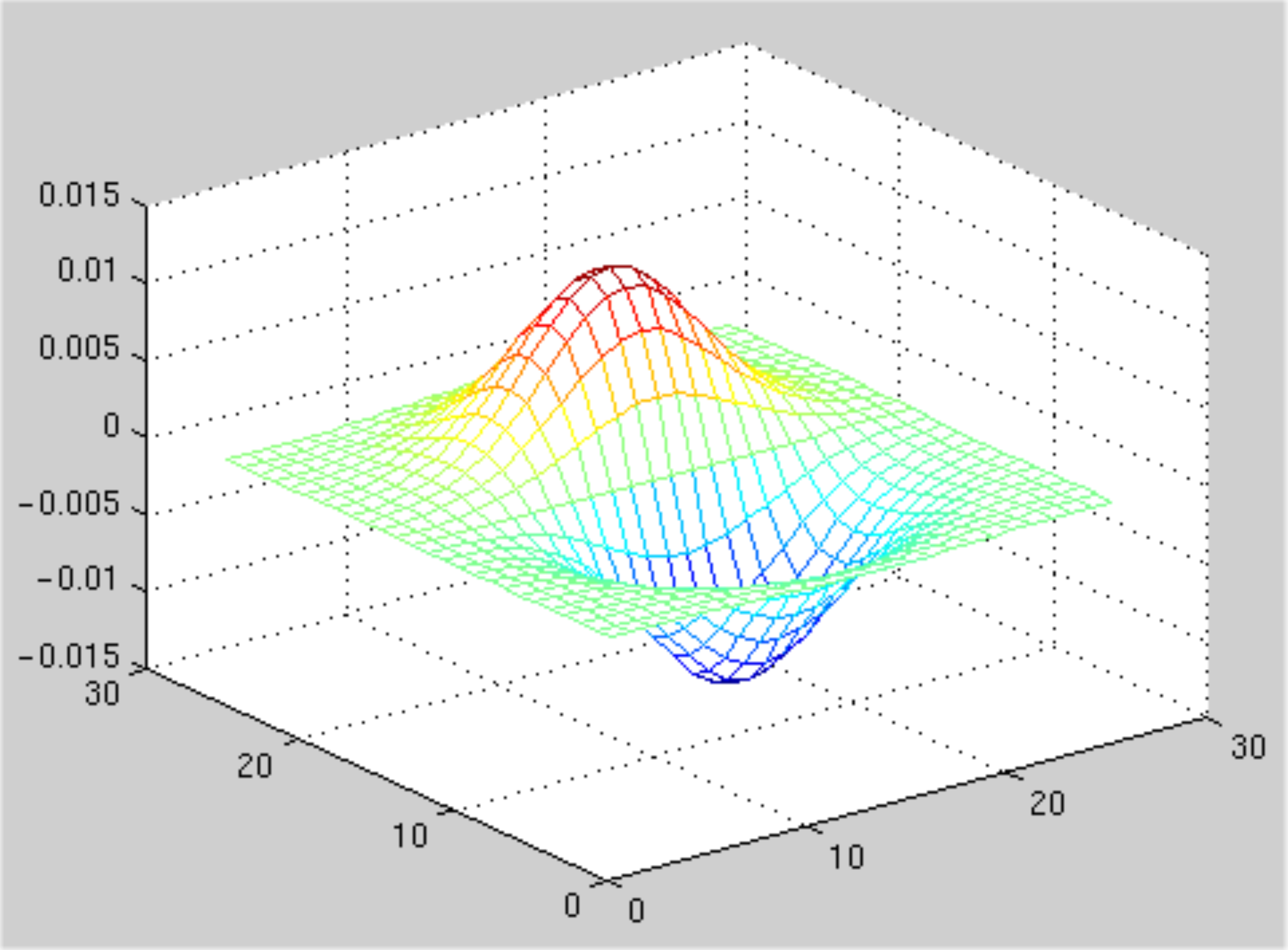


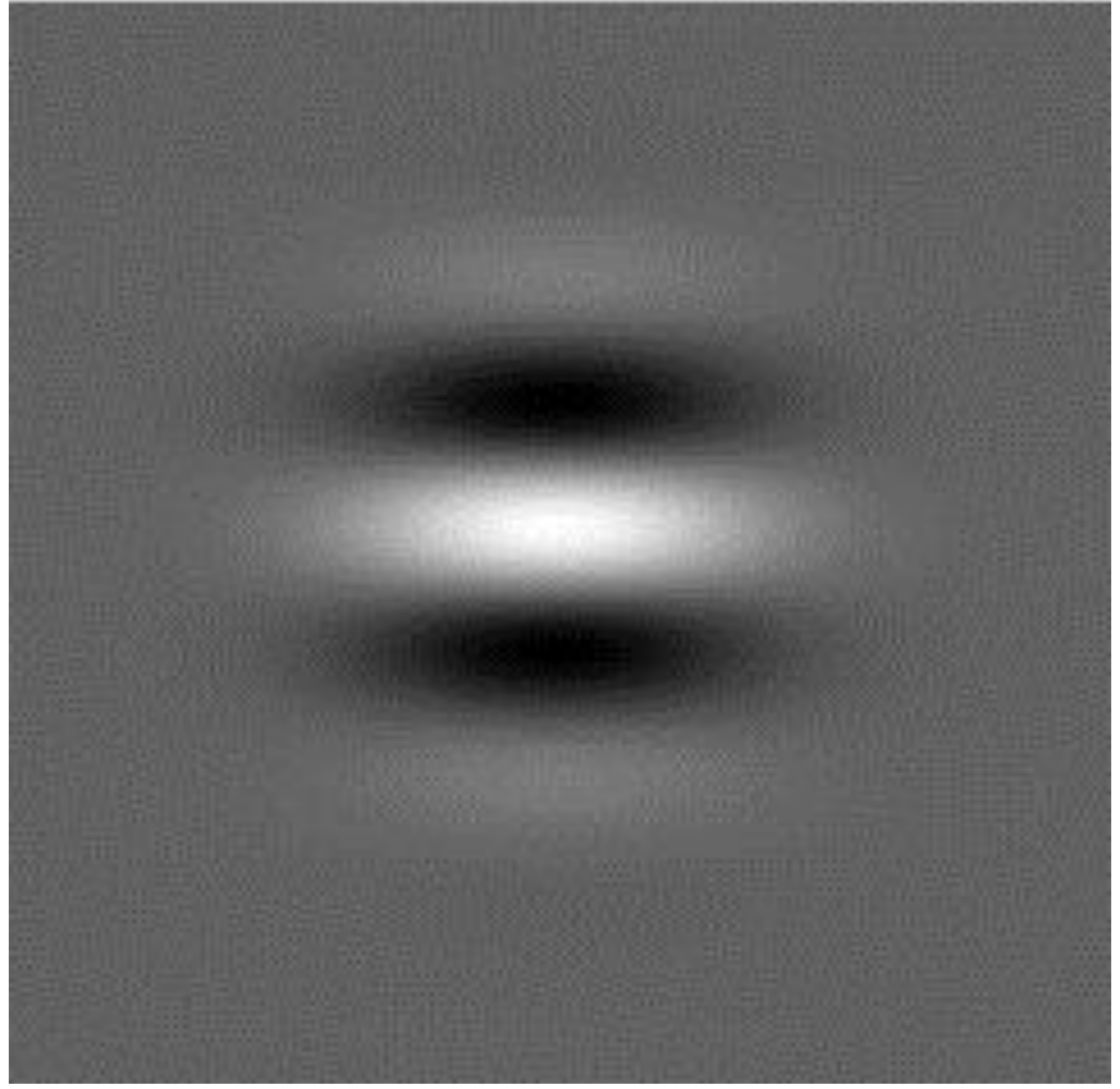
Odd Gabor Filter



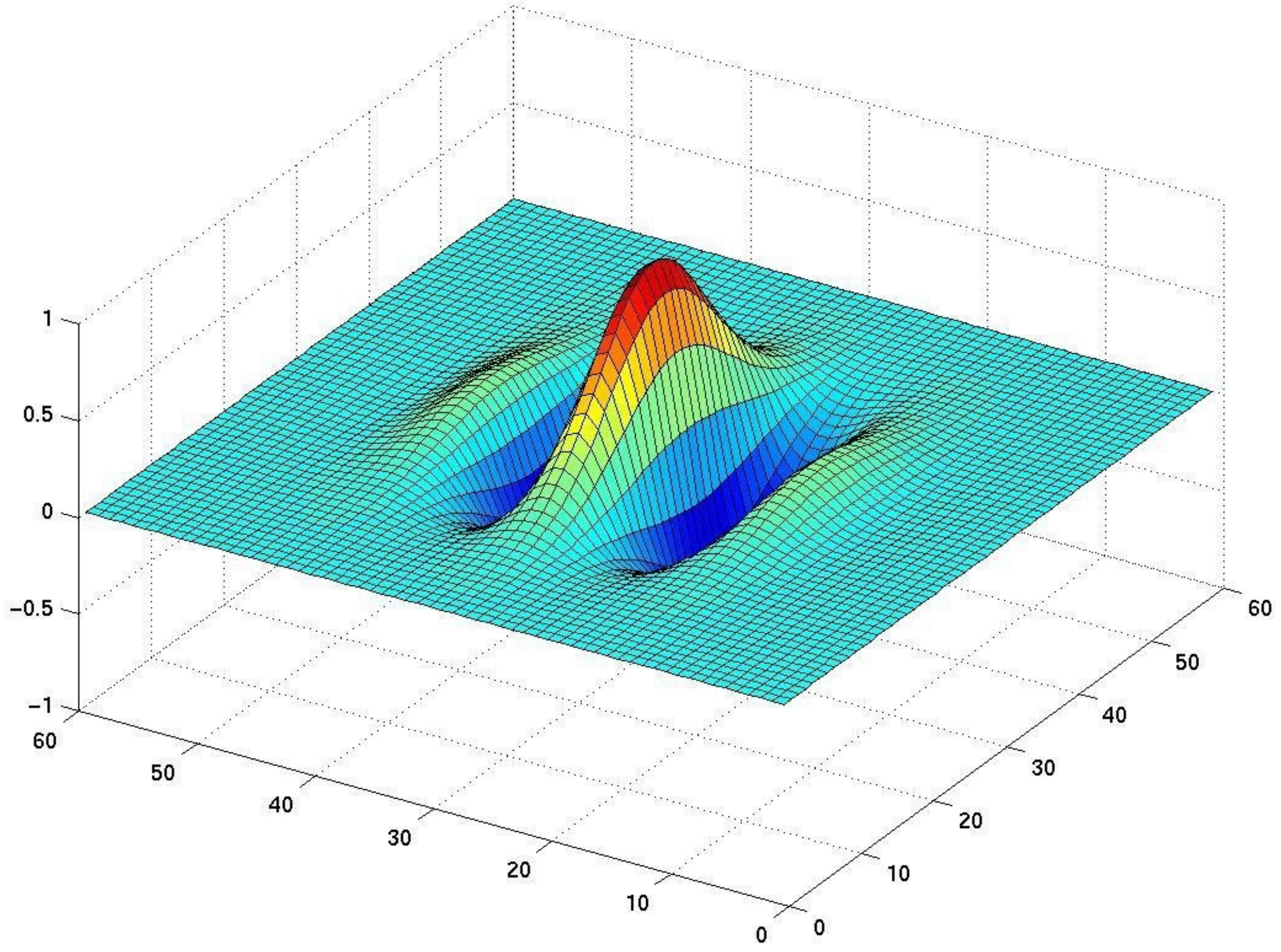


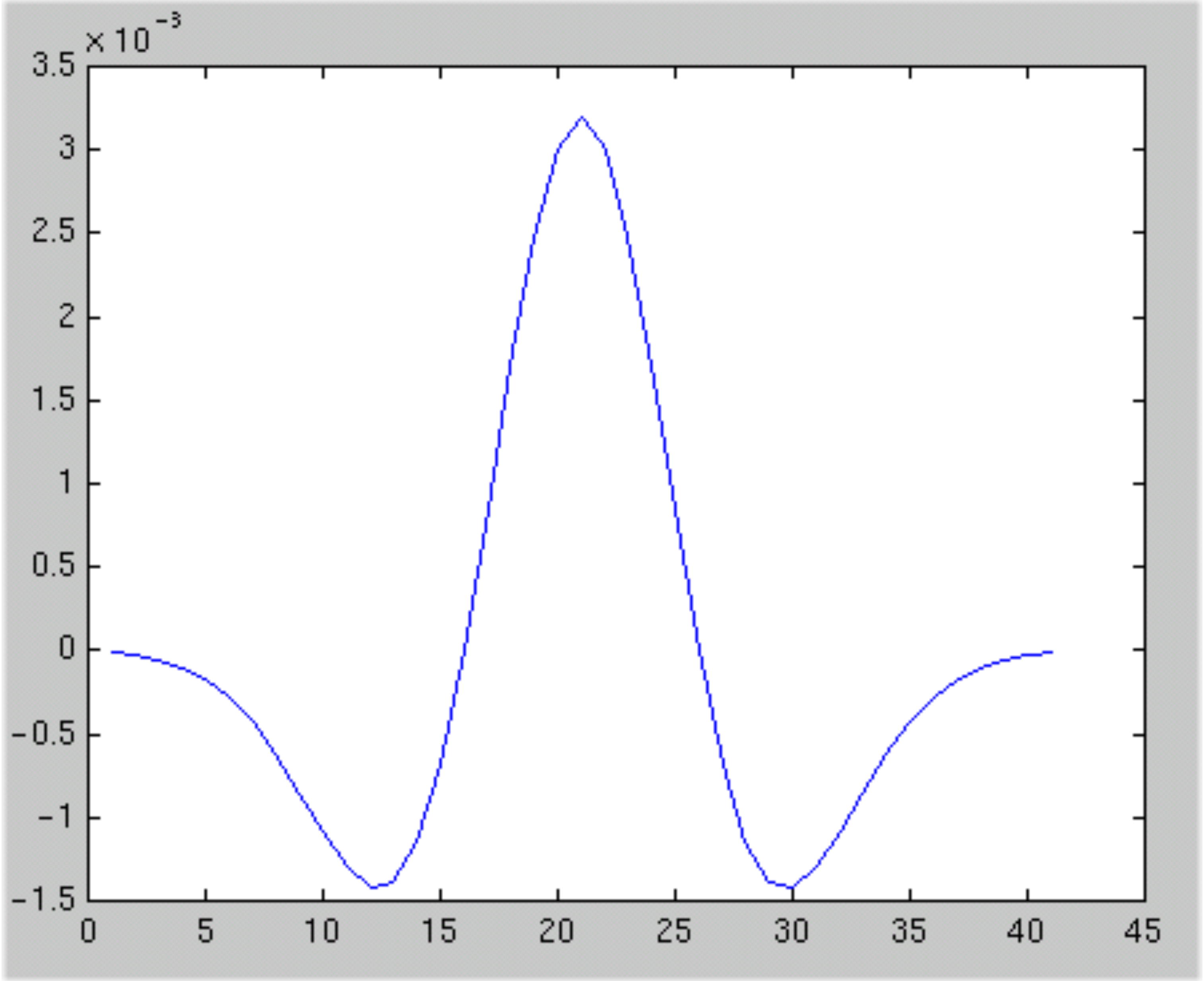
First Derivative



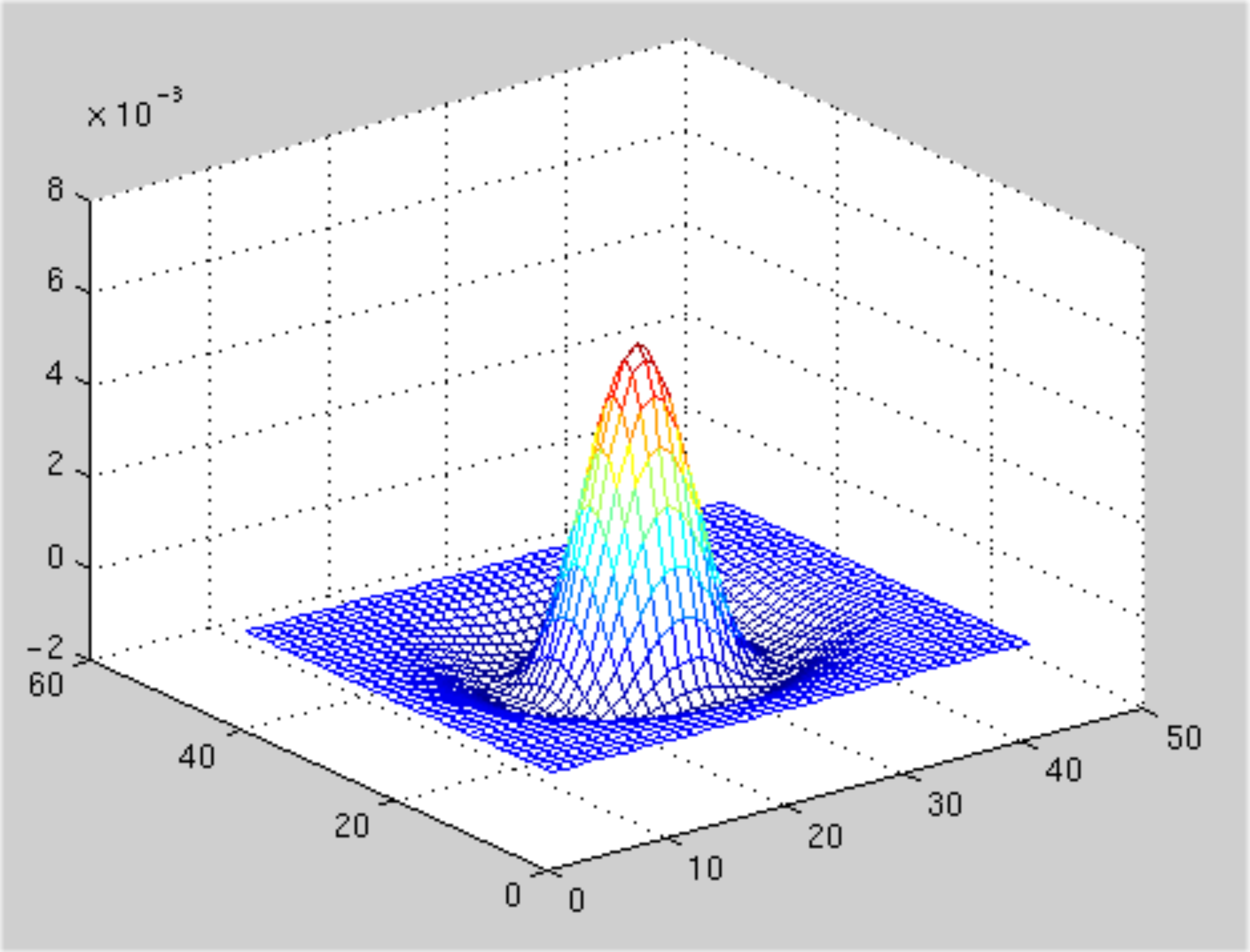


Even Gabor Filter

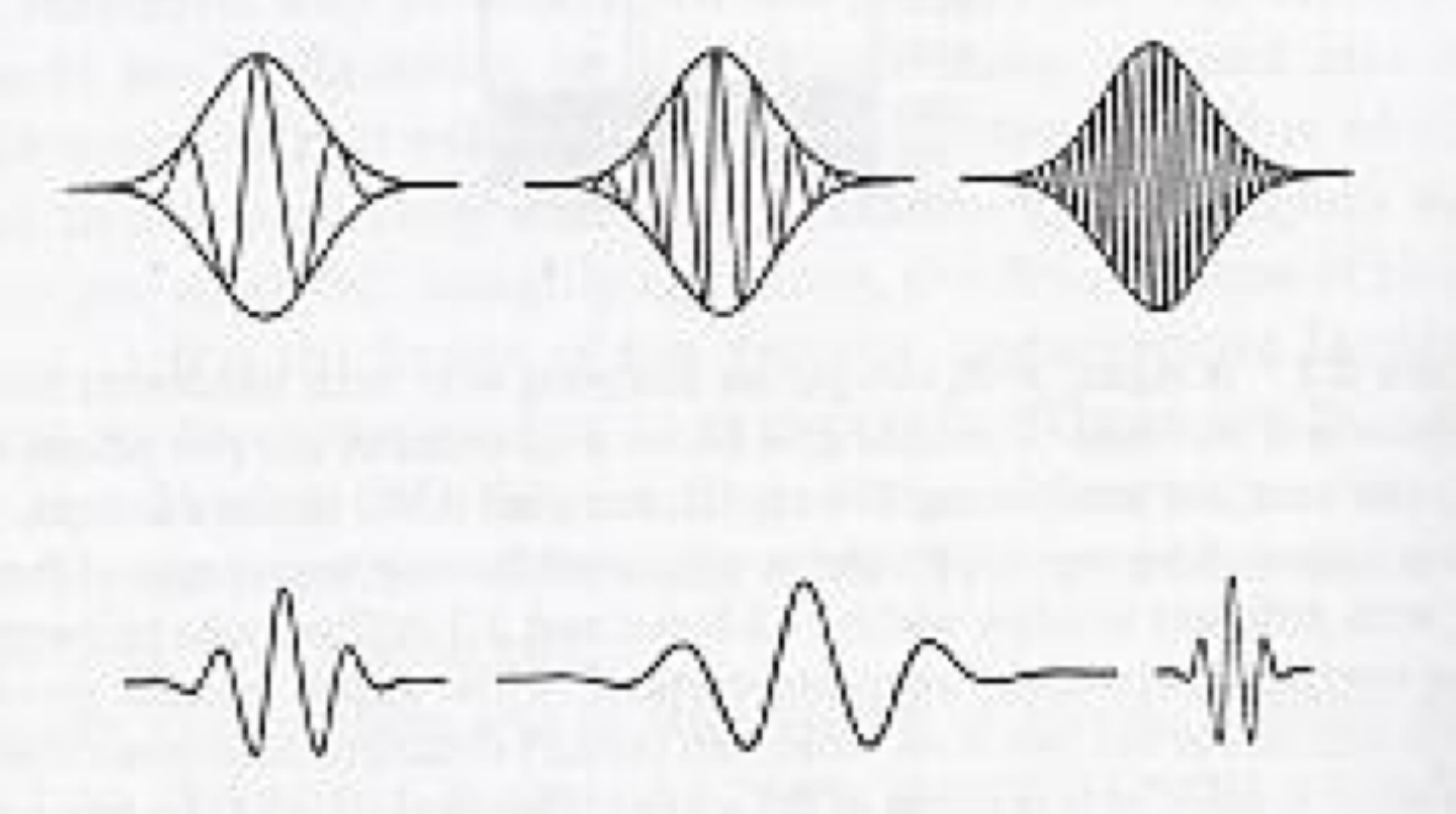




Laplacian



Matching Scale & Frequency



Example Results