COMPUTER VISION
MEI/1

University of Beira Interior, Departament of Informatics
Hugo Pedro Proença, hugomcp@di.ubi.pt, 2018/2019

Signals and Systems

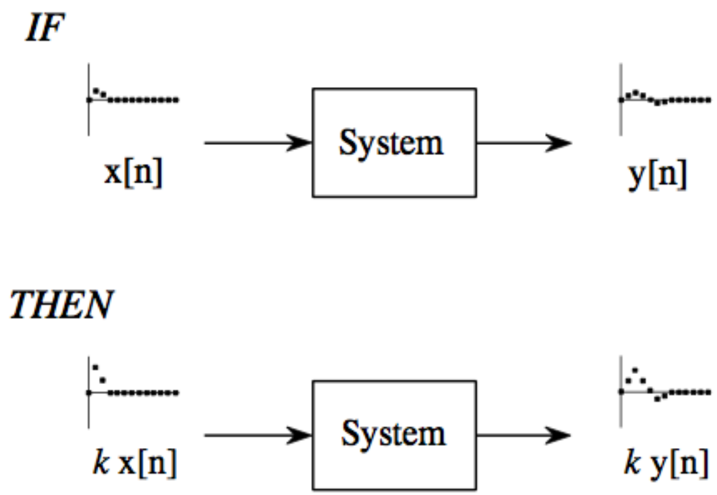
- What is a **signal**?
  - It can be regarded as a **description how** a **parameter varies** (dependent variable) **with respect to another** (independent variable);
  - E.g., the voltage of an electric charge varies with respect to time (1D signals) ;
  - E.g., the intensity of a pixel varies with respect its location in image (2D signals);
- Tipically, signals are denoted by **upper case letters**.
  - Discrete signals are denoted by **[]**:
    - E.g., X[n], Y[k]
  - Continuous signals are denoted by ()
    - E.g., X(i), Y(j)

Linear Systems

- A system is said to be **linear** if it complies two mathematical properties:
  - Homogeneity;
  - Additivity;
- There is a third property which is not a strict requirement for linearity, but it is mandatory for most pratical digital signal processing techniques:
  - Shift invariance

Linear Systems: Homogeneity

- Let f: R→R be a system, such that f(x)=y.
  - **If z=kx then f(z)=k f(x).**
- In practical terms a system is homogenous if an **amplitude change** in the **input** corresponds to an **identical amplitude** change in its **output**.



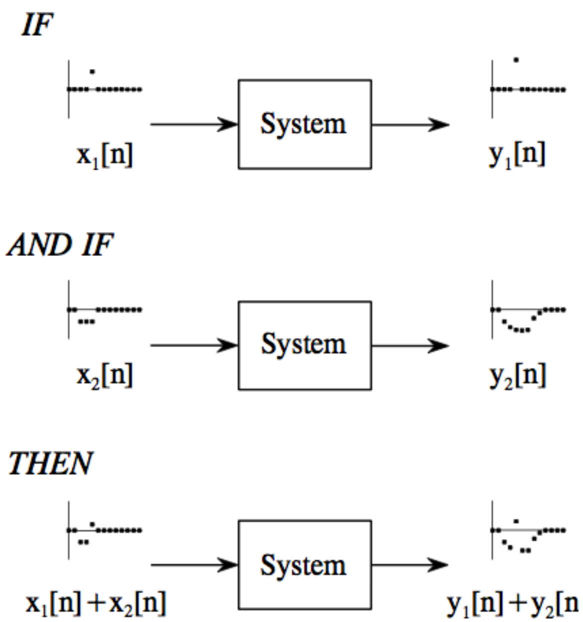
Linear Systems: Exercises

- Consider the following system $f: R^2 \rightarrow R$, such that:
  - $f(x,y)=2x-4y+2$
- Determine the homogeneity of “f”.

- Now, consider the following system $g: R \rightarrow R$, such that:
  - $g(x)= \exp(x)$
- Determine the homogeneity of “g”.

Linear Systems: Additivity

- Let f: R→R be one system, such that $f(x_1)=y$ and $f(x_2)=z$.
  - **If $x_3=x_1+x_2$ then $f(x_3)=f(x_1)+f(x_2)=y+z$**
- In practical terms a system is additive if **added signals** pass by the system **without interacting**.

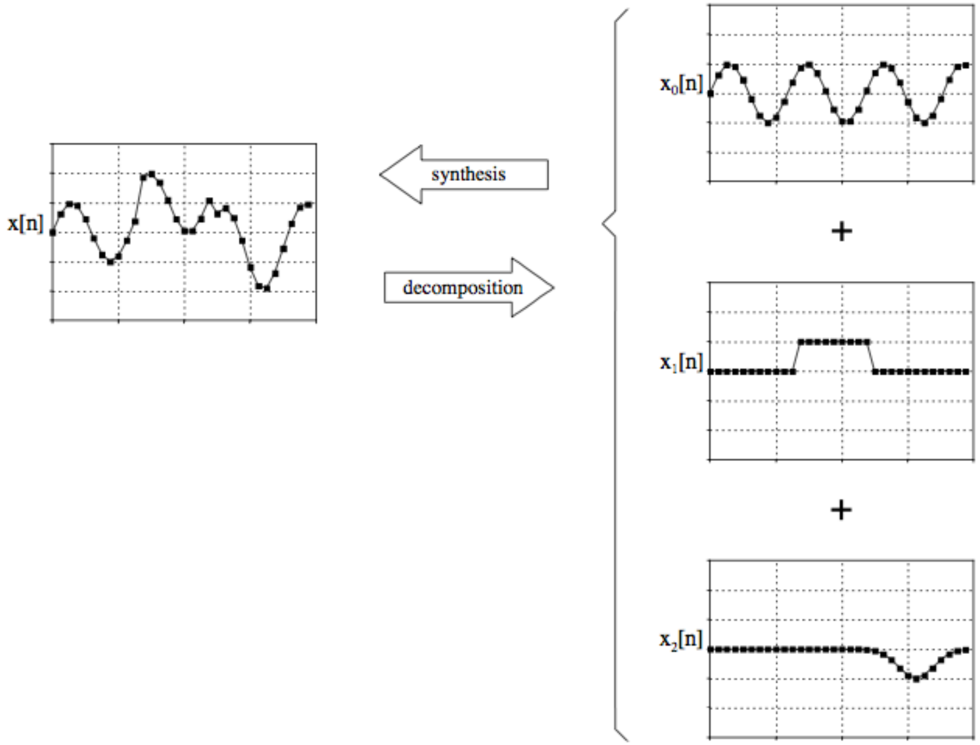


Linear Systems: Exercises

- Consider the following systems. Evaluate their additivity:
  - f: R→R, such that
    - $f(x) = x$;
  - g:R→R, such that
    - $g(x)=0$;
  - h:R^2→R, such that
    - $h(x,y)=xy$;
  - z:R^2→R, such that
    - $z(x,y)=x+3y$;

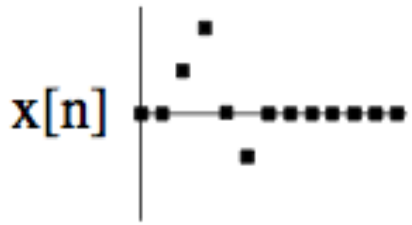
Superposition of Signals

- When we are working with linear systems, the only way signals can be combined is by scaling (multiplication of the signals by constants), followed by addition.
  - The process of combining several signals into a single one is called **synthesis**
  - The inverse process, broking a signal into its fundamental parts, its called **decomposition**.

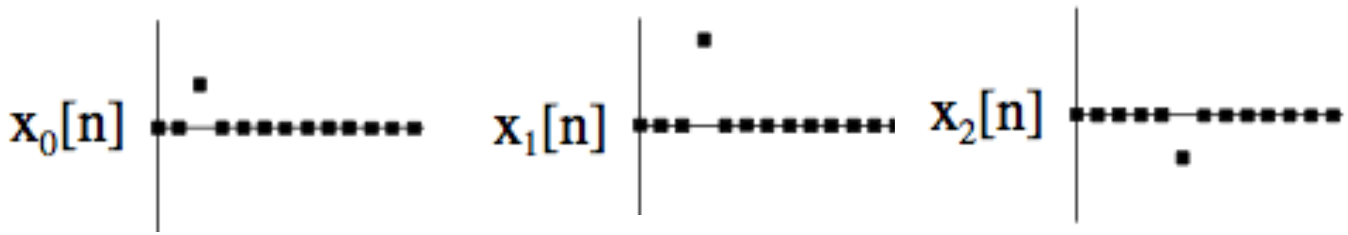


Superposition of Signals

- It's the **heart** of signal processing system.
  - It gives the overall strategy to understand how systems and signals are analyzed:
- Having one input signal:



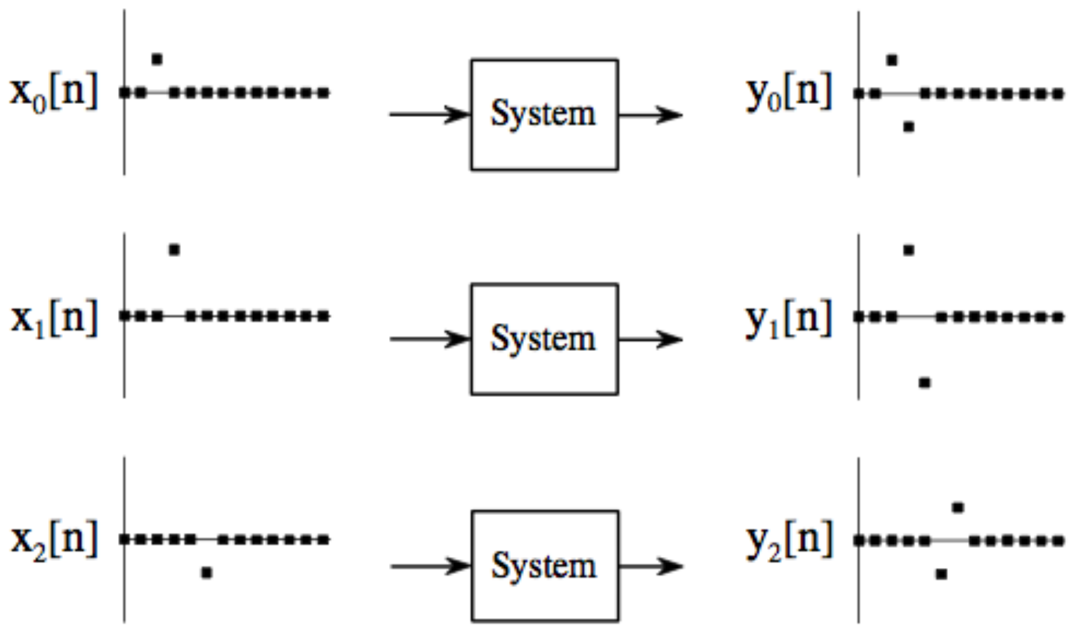
- We decompose it into simpler signals:



- ...remember that our goal is to **understand the system**!

Superposition of Signals

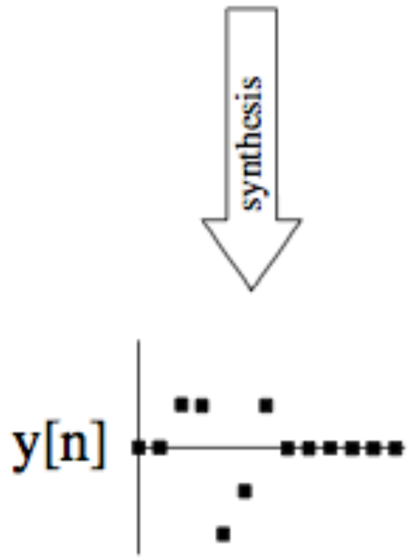
- Next, each input signal component passes individually through the system:



- These are the output signal components.

Superposition of Signals

- Instead of trying to understand how complicated signals pass through the system, all we need to know is how their **simplest components** are affected by the system.
- Finally, the output signal components are summed and we get the signal output, exactly equal as if the original signal was passed through the system.

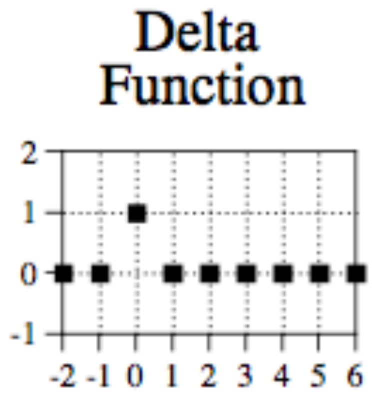


Signal Decomposition

- **Impulse Decomposition**
  - Decomposes the original signal “x” (length N) into N signals, where each component contains only one non-zero value:
    - $x_k(k)=x(k)$
    - $x_k(j)=0$, j<>k
- Impulse decomposition is important because it allows signals to be examined one sample at a time.
- By knowing how a system responds to an impulse, the system output can be calculated for any given input. This approach is called **convolution** and will be the subject of further discussions.
- **Exercise**: Consider the following signal, represented in time-domain:
  - [2,3,-4,1,0,5,2,4]
- Use impulse decomposition in the above signal and extract the resulting impulses.

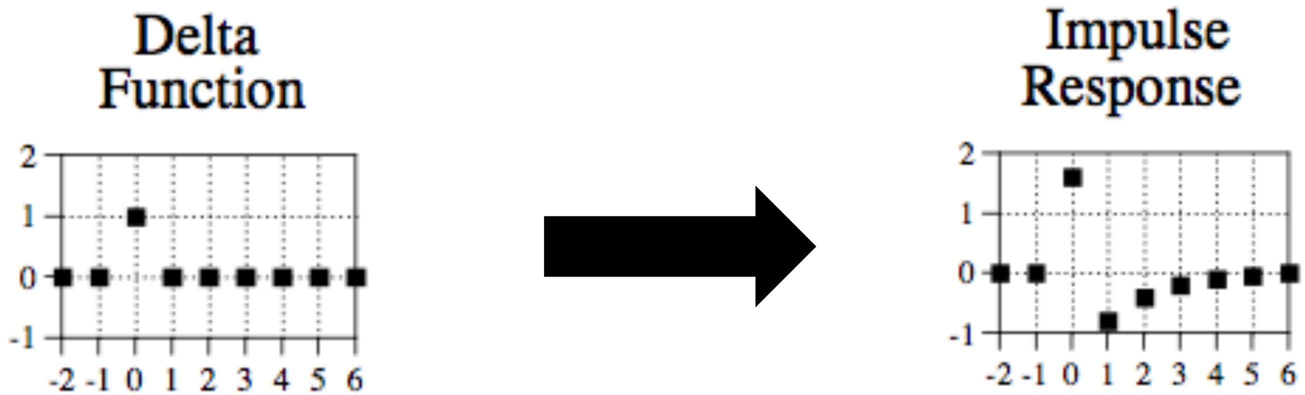
Signal Decomposition

- **Impulse Decomposition**
  - The notion of "**Delta function**" (d) is extremely important, when using impulse decomposition. A delta function has the central component equal to 1and the remaining ones equal to 0.
  - Let $f_k(x)$ be a signal resultant of input decomposition of f(x).
    - $f_k(x)$ = k d(x+t). Every input is s scaled and shifted version of the delta function



Signal decomposition

- **Impulse Decomposition**
  - According to the above discussion, the output signal can be found by adding the output of these scaled and shifted impulse responses.
  - In practical terms, if we know the response of a system to an impulse, we know everything about that system.

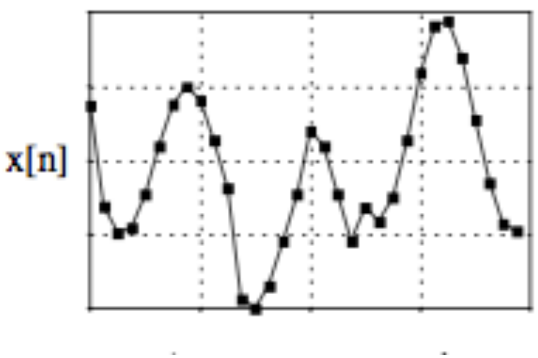


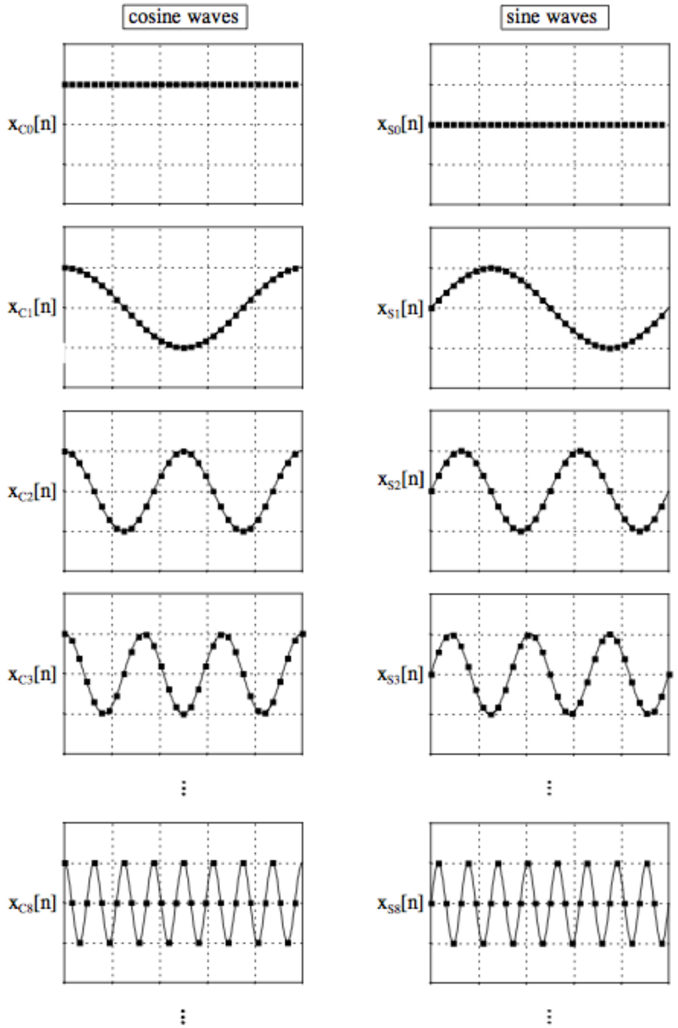
Signal Decomposition

❑ **Fourier Decomposition**

- ❑ It resulted from a very important finding, by J. Fourier
  - ❑ "Any periodic signal can be decomposed by a (potentially infinite) sum of simpler periodic signals".
- ❑ In practice, it decomposes any N length signal into N+2 signals, half of them sin waves and the remaining ones cosine waves.
  - ❑ The first cosine component has fundamental frequency 0. The second has fundamental frequency 1, …
  - ❑ Similar observations for the sin waves.
- ❑ Since the frequency of each component is fixed, the only thing that changes for different signals being decomposed is the amplitude of each of the sine and cosine waves.

Fourier Decomposition: Example





Signal Decomposition

- Exercise: Consider the following impulse response of a 1D signal in a system "f" (centered at index "0").
  - **[0, 0, -1, 0, 1, 0, 0]**
- Determine:
  - f([1,2,4,0,-1,1,0,2,3,1,0])

- In the general signal processing domain, the impulse response of a system is called "**filter kernel**" or "**convolution kernel**".
- In image processing, it is called point spread function.

Convolution

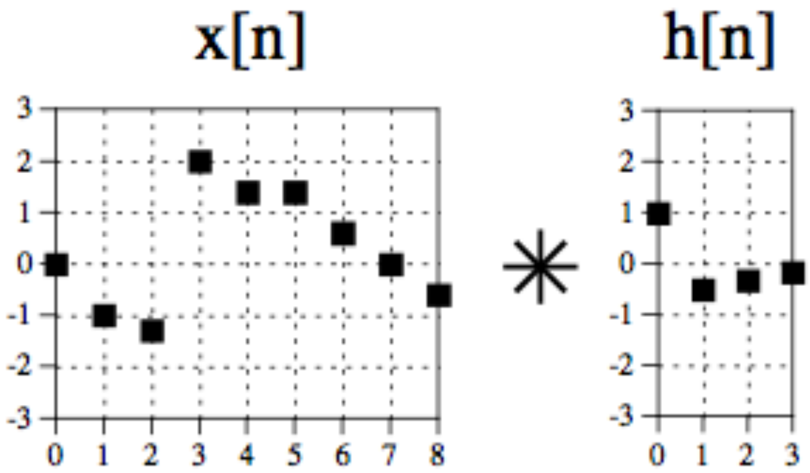
- It is a mathematical operation that describes the relationship between three signals:
  - One **input** signal;
  - One **impulse response;**
  - Yielding the **output** signal

- As it combines addition (+) with multiplication (x), it is usually denoted by "*****".
  - Y[k]=H[k]*X[k]

$$y[i] = \sum_{j=0}^{M-1} h[j]\,x[i-j]$$

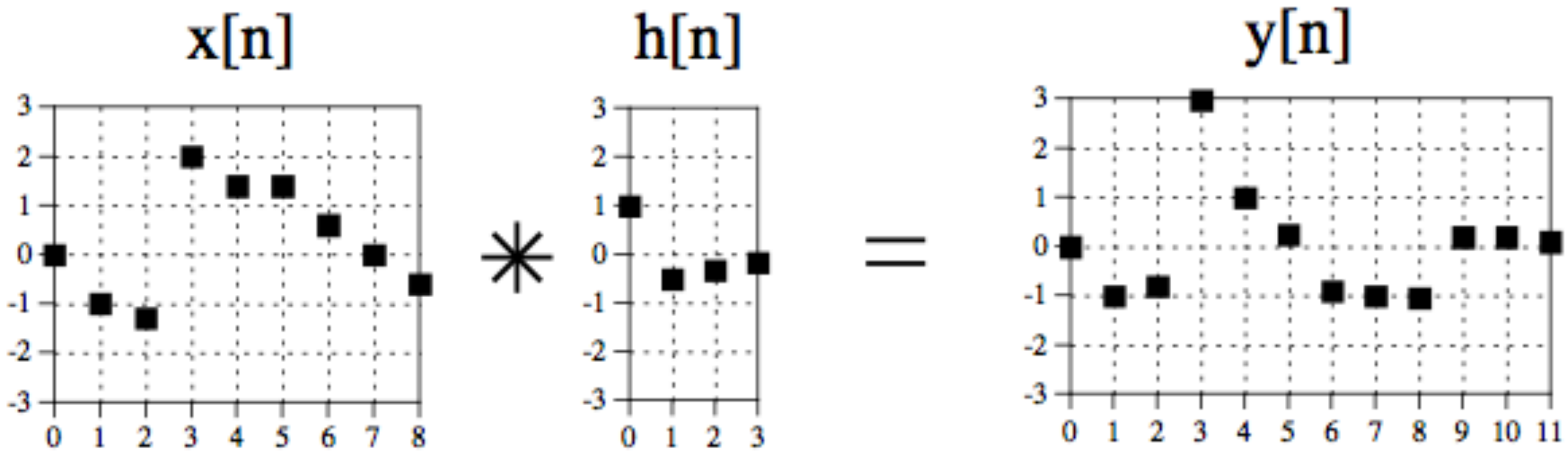
Convolution: Exercise

❑Obtain the result of the convolution of the following signals:



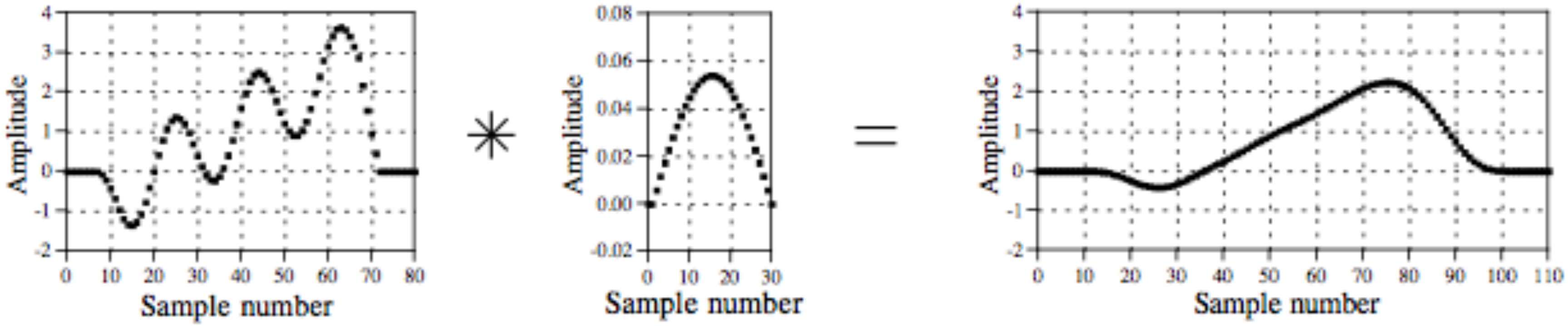
Convolution: Exercise

- Obtain the result of the convolution of the following signals:



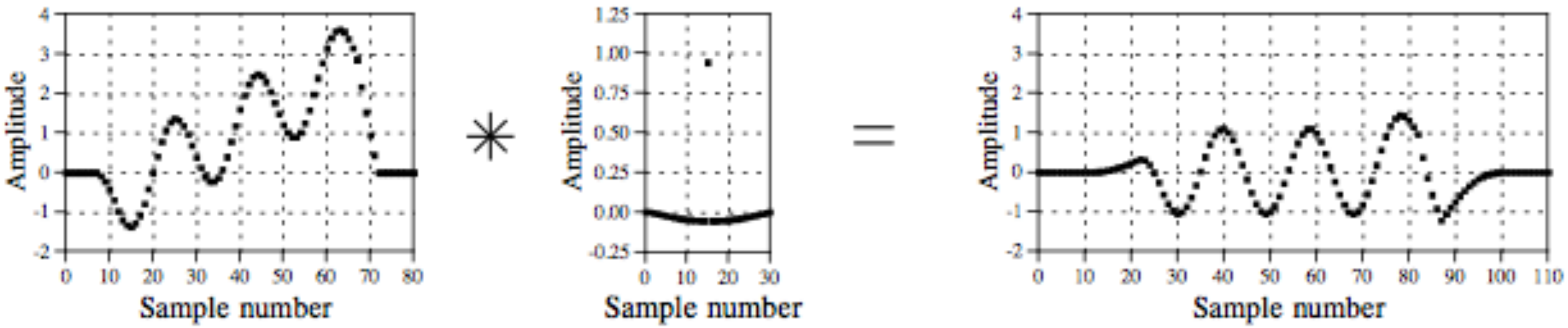
Convolution: Examples

❑**Low-pass filtering:**



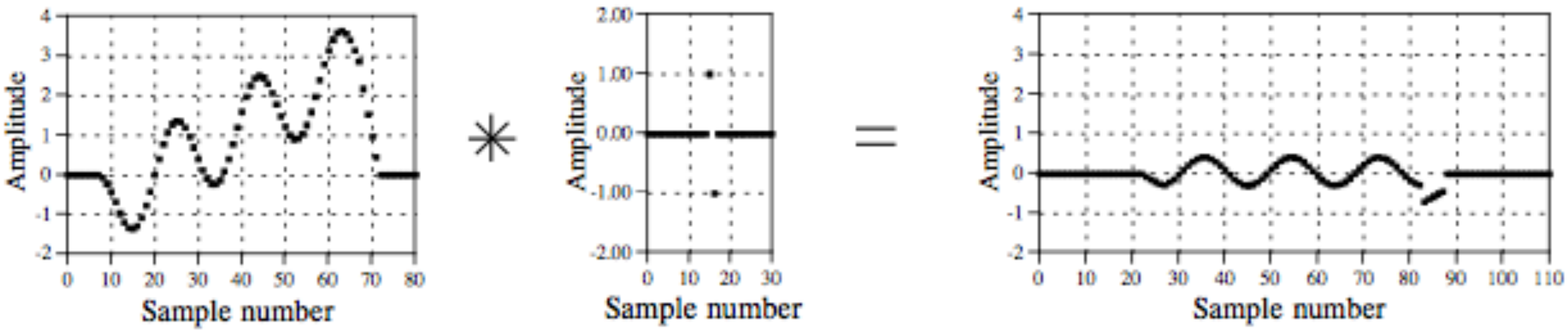
Convolution: Examples

❑**High-pass filtering:**



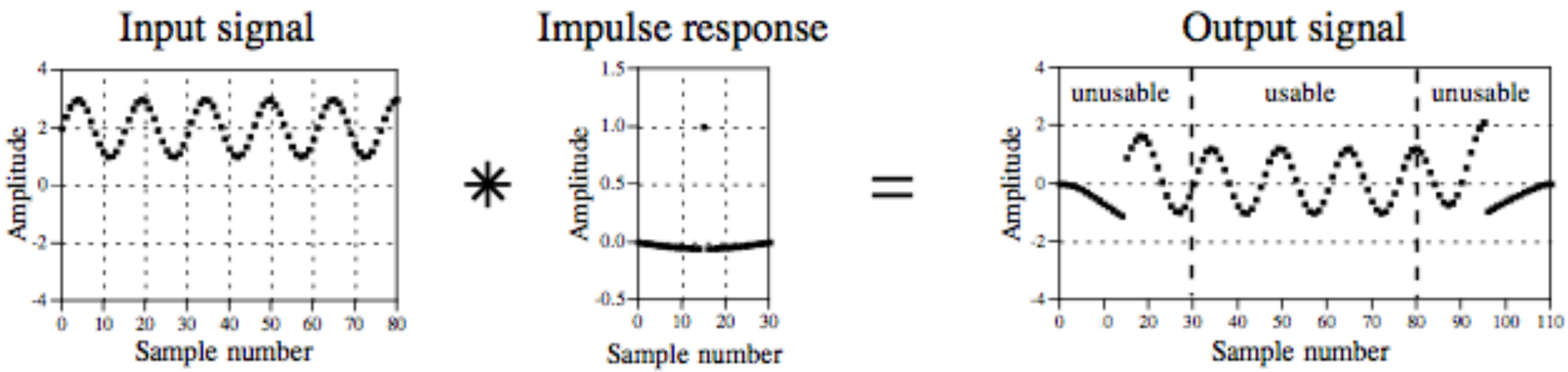
Convolution: Examples

❑**Discrete derivative:**



Convolution: Caution!!

- When an input signal is convolved with an impulse response of length “M”, then the first and last “M-1” components are not fully reliable.
  - Why is this?



Frequency Domain

- Any signal can be represented by a linear combination of basis-functions.
- In case of 2D images, we have the following function:

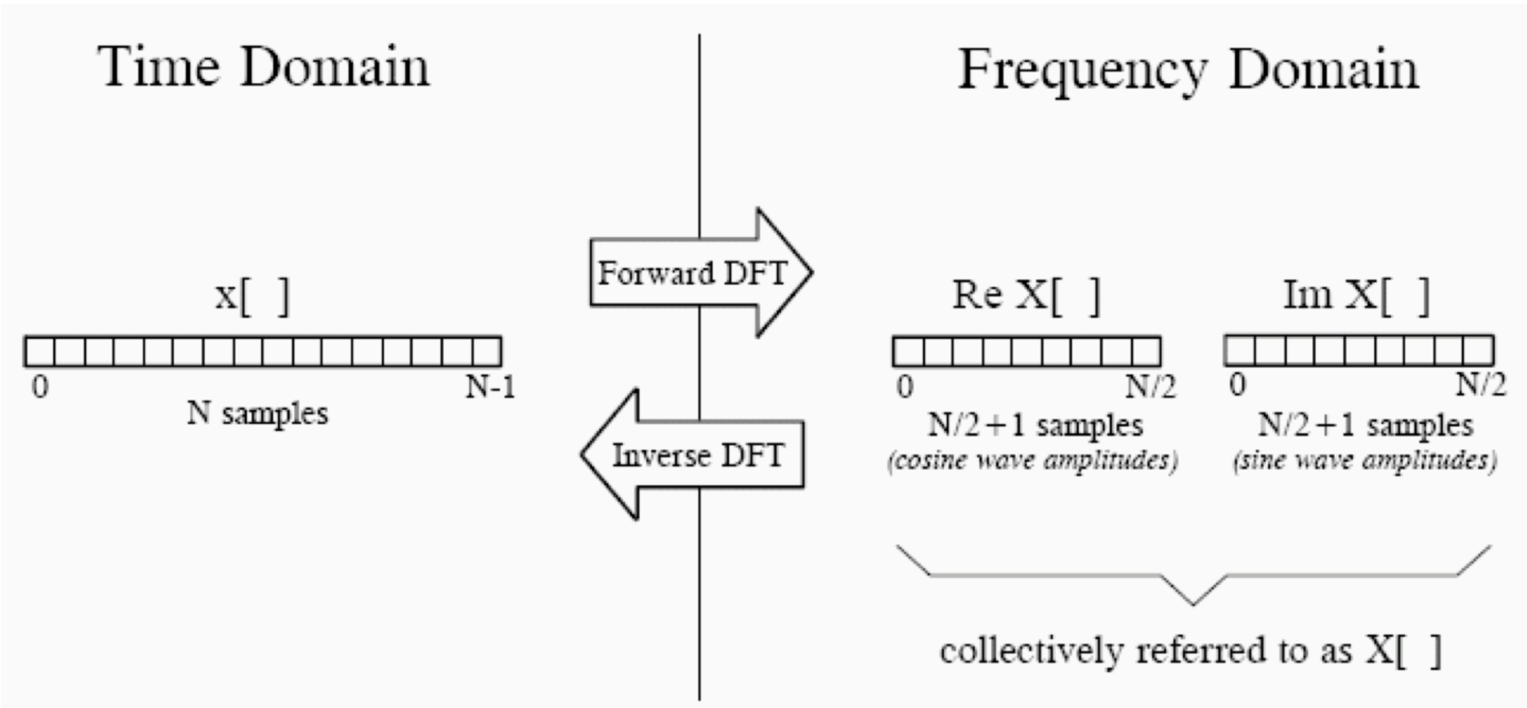
$$f(x, y) = \sum_k a_k \Psi_k(x, y)$$

- Here, a_k are the contributions of each basis-function to the original image.
- Basis functions are exponentials, complex and expressed in terms of harmonic functions ("*sin*" and "*cos*"):

$$\Psi_k(x, y) = \exp(i(\mu_k x + \nu_k y)) \qquad \exp(i\theta) = \cos(\theta) + i\sin(\theta)$$

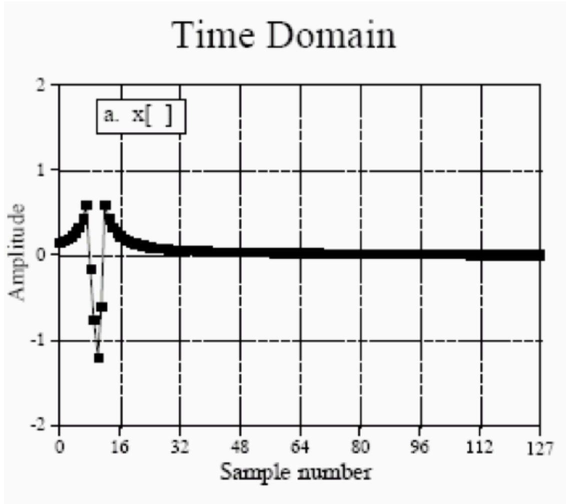
Discrete Fourier Transform (DFT)

- We can build the following correspondence between any signal represented in the time (space) and frequency domains:

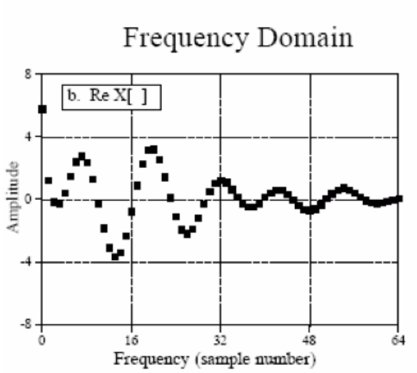


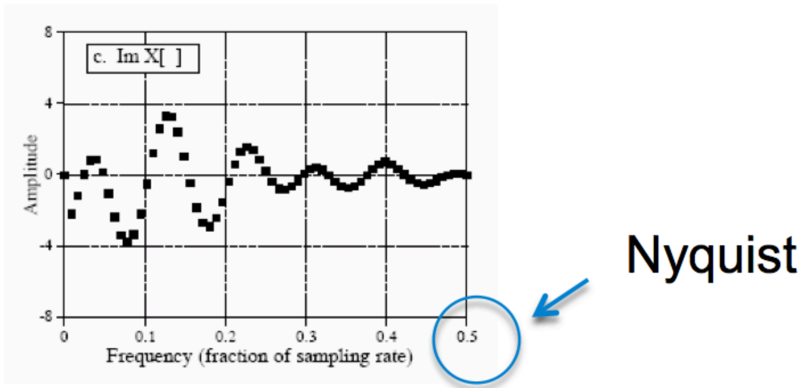
Discrete Fourier Transform (DFT)

- Suppose we have the following signal, represented in the time-domain:



- By using the DFT algorithm, we are able to express it in the following way:

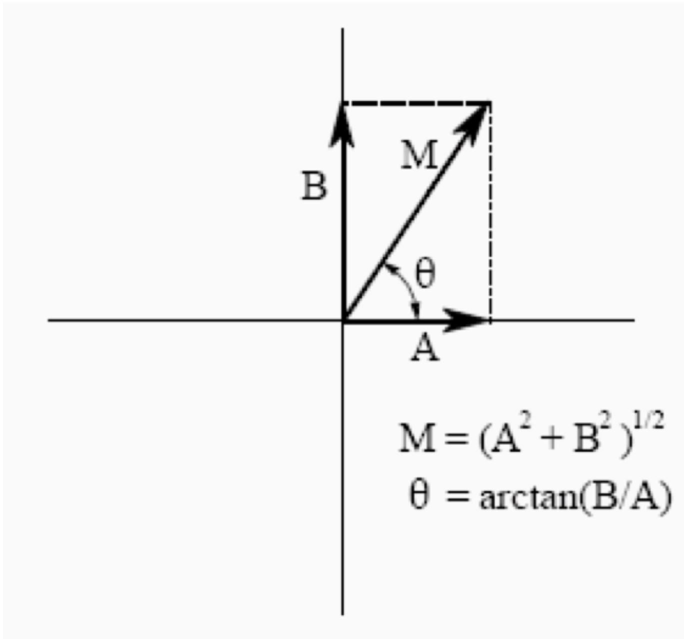




Discrete Fourier Transform (DFT)

- Often, it is more understandable to express the output of the DFT in Polar coordinates (**magnitude** + **phase**) , rather than in the original real and imaginary components:

  - Mag(X[k])
    - sqrt($Re(X[k])^2 + Im(X[k])^2$)
  - Phase(X[k])
    - arc tan(Im(X[k])/Re(X[k]))



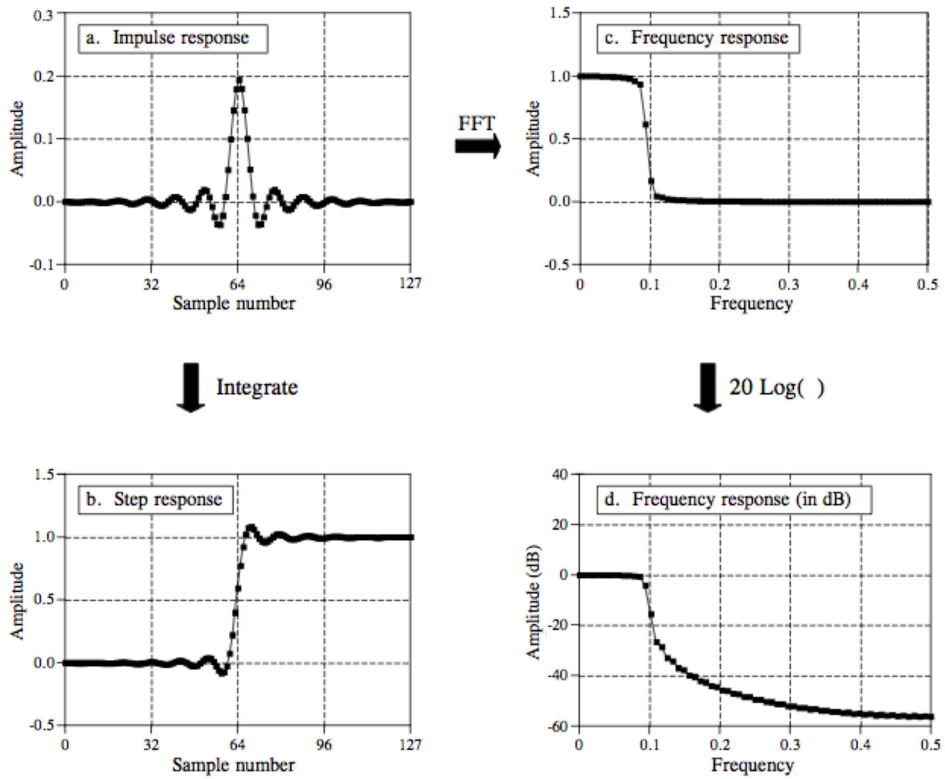
Convolution Theorem

- The convolution of two signals in a given domain (either spatial or frequency) corresponds to the point-by-point multiplication in the complementary domain.
  - **H(x)=f(x)*g(x) ←→ H(x)=F(x) x G(x)**
- This is extremely important in modern DSP and in practical terms, enabled the existence of most state-of-the-art technologies and devices:
  - TV, radio, computer,…;

Filters

- According to the convolution theorem, the convolution in time|frequency domain corresponds to multiplication in frequency|time domain.

  - Each filter has an **impulse response**, a **step response** and a **frequency response**:



Filters

- **Impulse Response**
  - Output of the system to an impulse;

- **Step Response**
  - Output of the system when the input is a step;
    - It can be obtained without passing any signal thorugh the system.
    - By integrating (running sum in discrete mathematics) the impulse response.

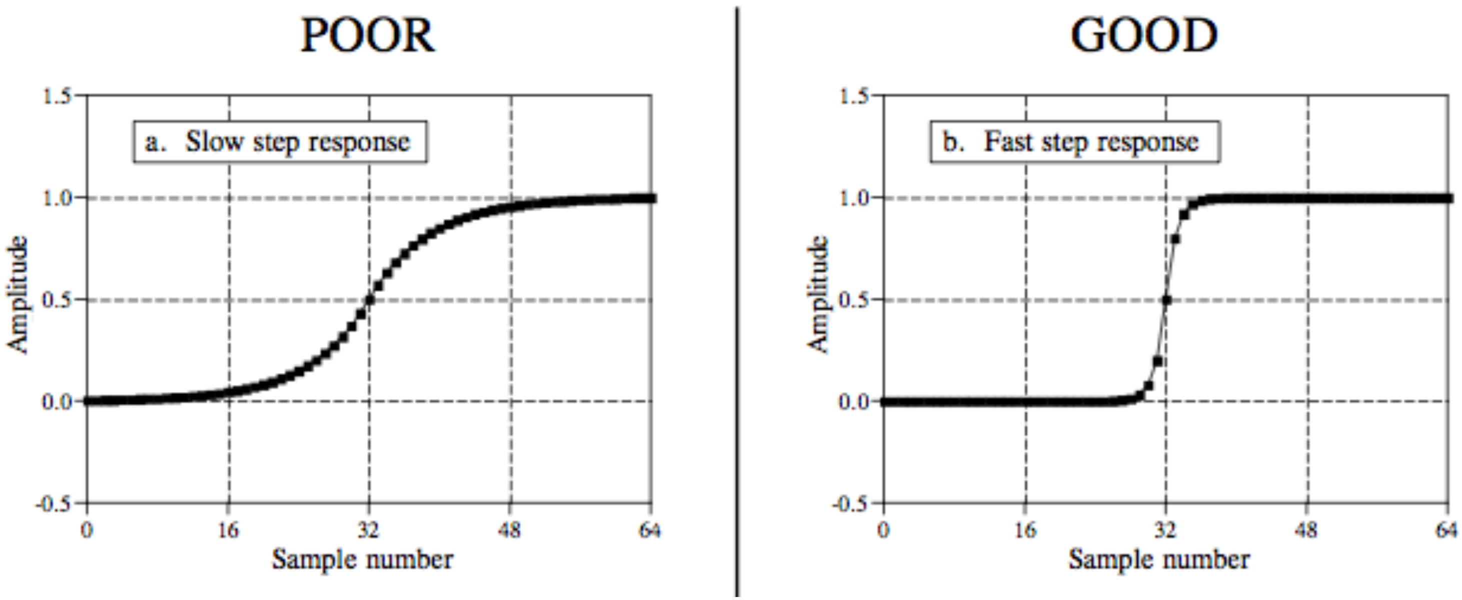
- **Frequency Response**
  - It can plotted in liner or logarithmic scales (decibels).
  - Corresponds to the **Fourier Transform** of the Inpulse Response

Filters

- The step response is often used to measure how well a filter performs in the time domain, mostly in terms of:
  - **Transition speed**. In order to discriminate components of a signal, the duration of the step should be shorter that the spacing of events. Thus, the transition speed should be as fast as possible.
    - Usually expressed by the proportion of samples between a low a high amplitude levels (10 and 90%).

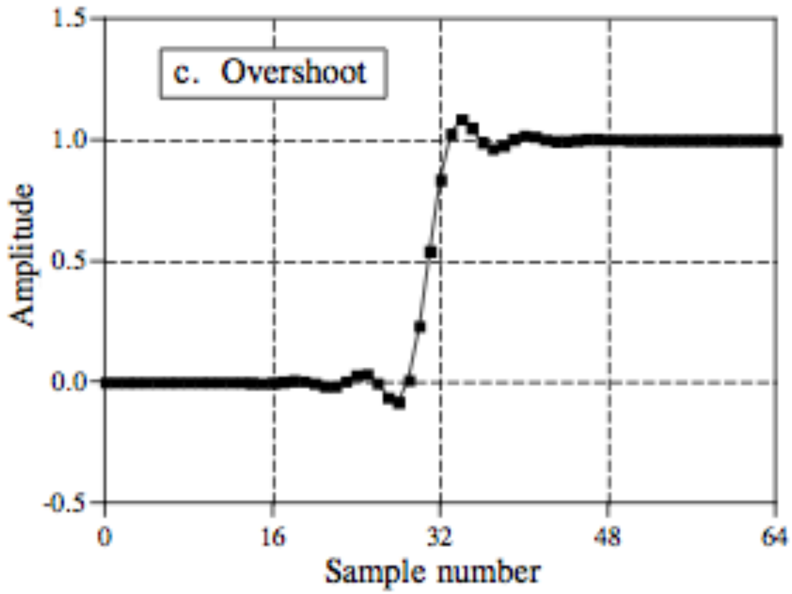
Filters: Examples

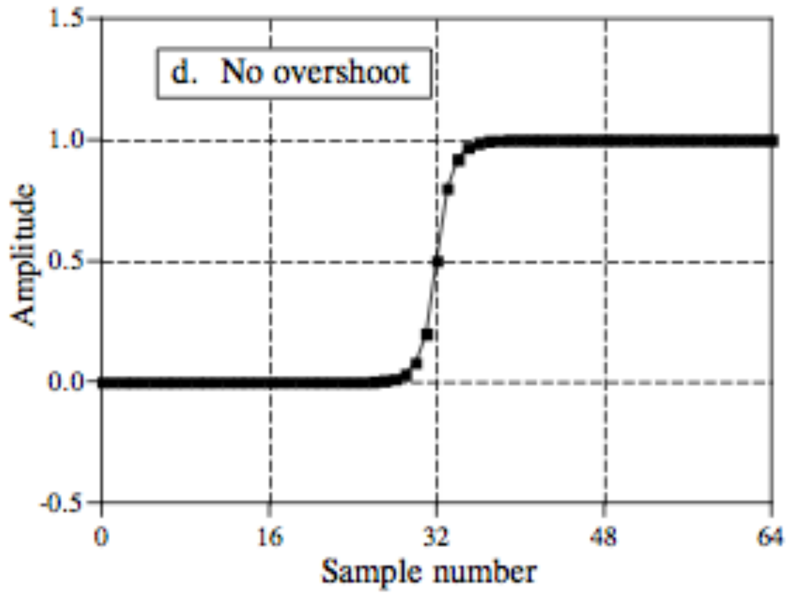
❑Transition Speed



Filters: Examples

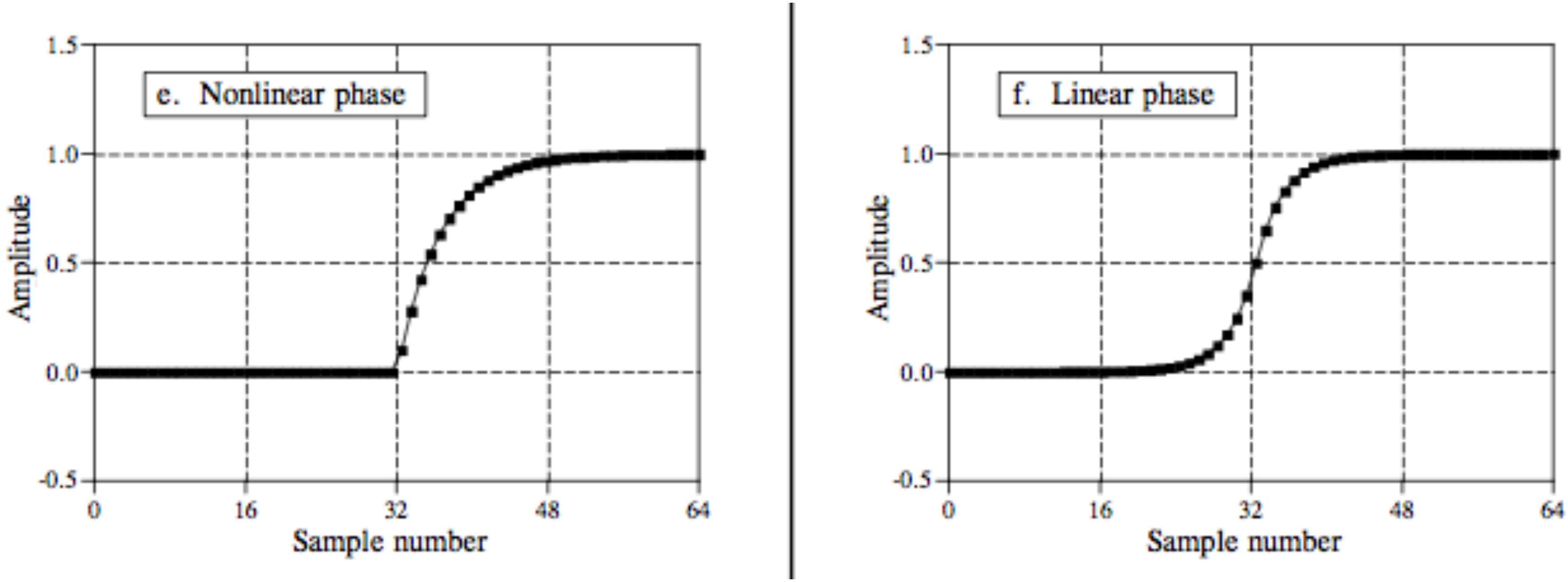
- **Overshoot.**
  - It corresponds to inverse variations to the major variation of step response.
  - It changes the signal amplitude non-homogeneously.





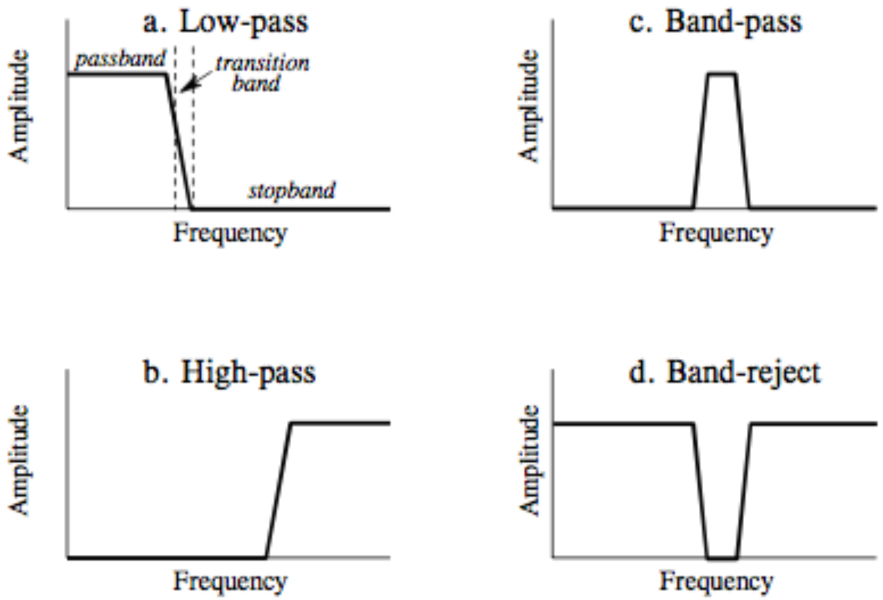
Filters: Examples

- Linear Phase.
  - Usually it is desired that the upper half of the step response is symmetrical to the lower half.



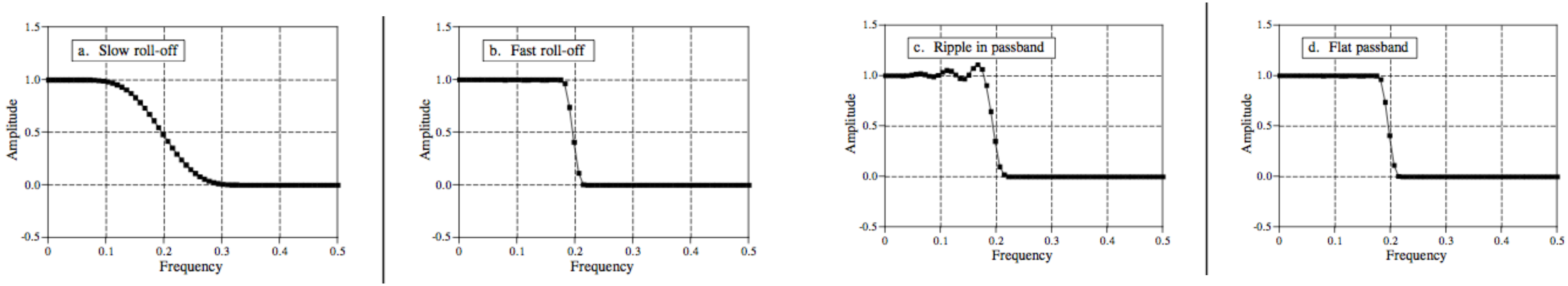
Filters: Summary

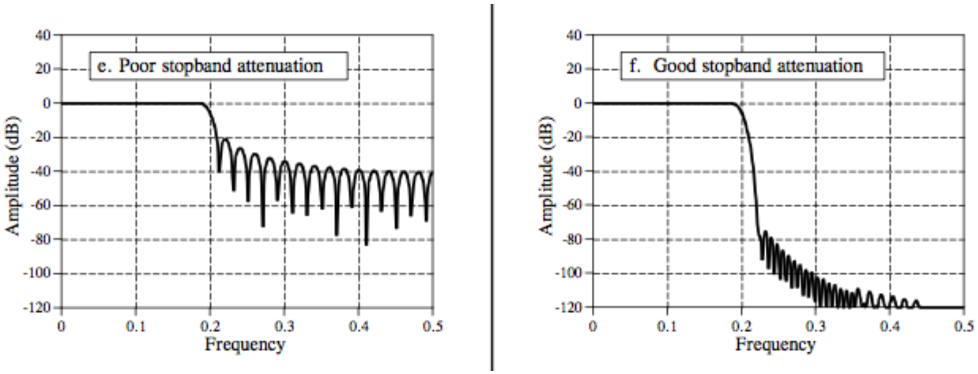
- When analyzing a system in terms of its frequency response, the most important factor is to observe the amount of frequencies that are blocked or passing through the system.
  - The **pass band** refers to the range of frequencies that pass trough the system
  - The **stop band** gives the frequencies that are blocked
  - The **transition band** is on the boundary



Filters

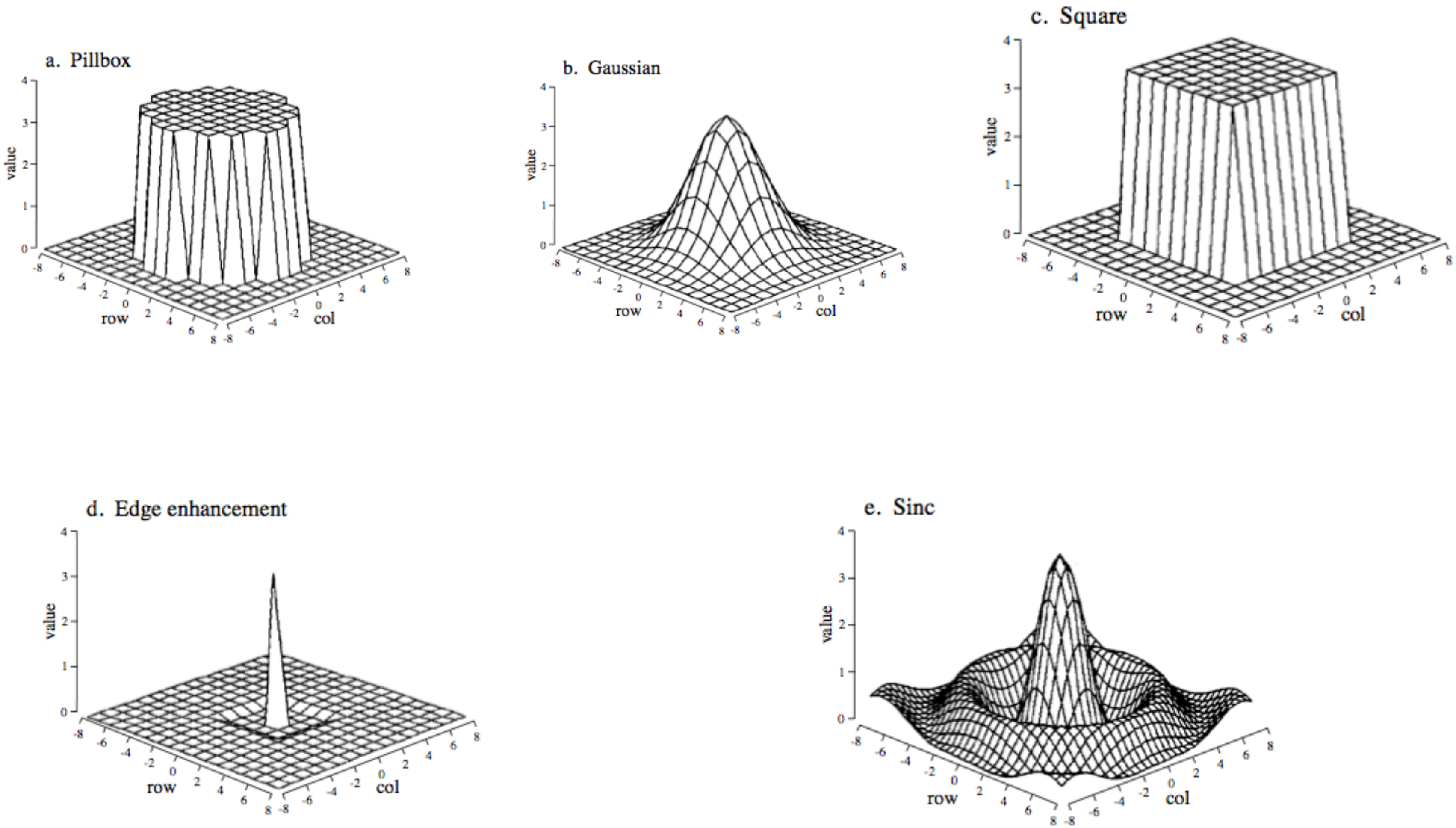
- We are usually interested in filters that have a fast **roll-off** (short transiction band) and without **ripples**. Finally, in order to actually block frequencies, we want to keep good **stop band attenuation** (expressed in logarithmic scale).





A Bel (Alexander Bell) expresses that the power is changed one order of magnitude. As such, decibel values of -10dB, 0dB, 10dB mean power ratios of 0.1, 1 and 10. **Amplitude is the square root of power**. As such, 20dB mean that amplitude changes one order of magnitude

Filters: Examples



Multiscale Analysis

- Usually, features occur in signals at different locations, scales (rotations, …).
- A widely used strategy is to build a **data pyramid**:
  - Different versions of the same data, each one represented at different scale
  - According to the Nyquist Theorem, only two samples are required to reconstruct a signal with 1 cycle.
  - By blurring each scale with different Gaussian kernels, there is useless information at each scale.
  - That useless information is simply removed from the signal.

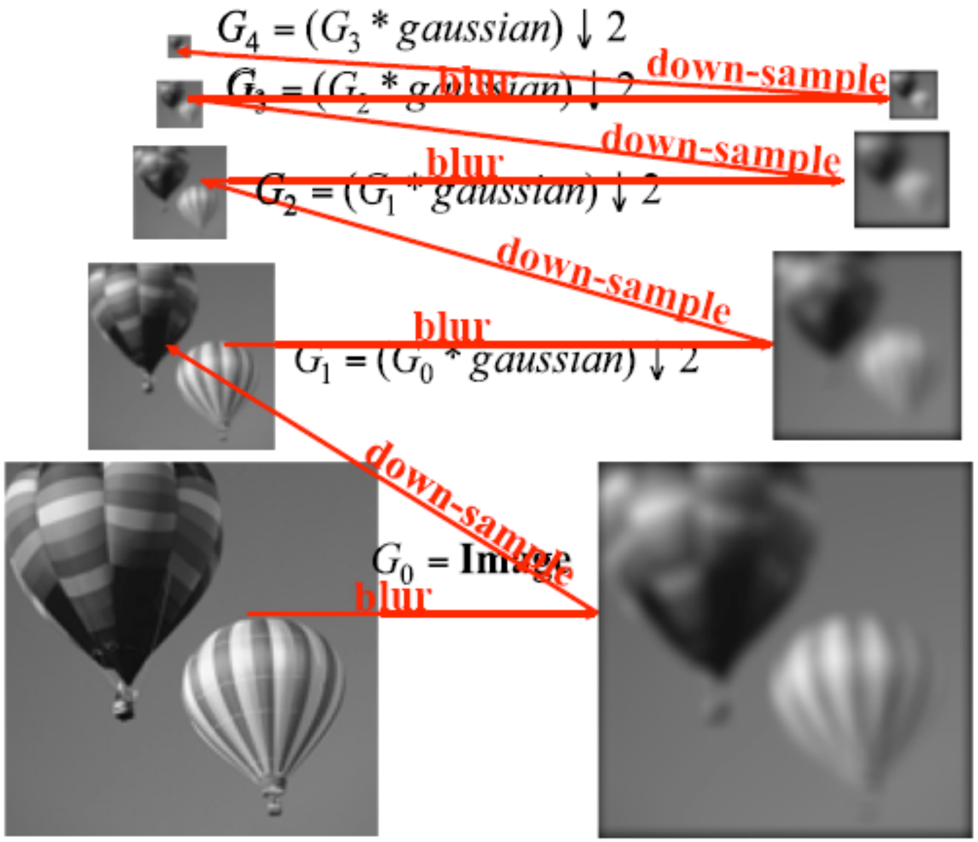


Image Scale-Space

- It is a theory for representing signals at multiple scales.
  - One parameter (t) family of smoothed images. “t” is defined as the scale parameter
  - All structures smaller that sqrt(t) are smoothed away from the “t”-level of the scale-space.
- Let f(x,y) be an image, represented in the spatial domain:
  - f: N x N → N
- Let g(x,y,t) be a family of Gaussian kernels, such that:
  - g:N x N x N → N

$$g(x, y; t) = \frac{1}{2\pi t} e^{-(x^2+y^2)/2t}$$

Image Scale-Space

- The scale-space representation at level "t" is given by:
  - L(x,y,t) = f(x,y) * g(x,y,t)
- As a limit, for t=0, g(x,y) becomes an impulse function.
  - As such, L(x,y,0)=f(x,y)
- Example:
  - L(x,y,0)

Image Scale-Space: Example

L(x,y,1)

L(x,y,4)

L(x,y,16)

L(x,y,64)

L(x,y,256)

Image Scale-Space: Interest Points

- Let $L_x = \vartheta_x / L(x,y,t)$ and $L_y = \vartheta_y / L(x,y,t)$
- An edge at scale "t" is a local maximum of the gradient magnitude:
  - $L_v = sqrt(L_x + L_y)$
- Similarly, a **BLOB** (Binary Large Object) at a given scale "t" **corresponds to the extreme values** of the determinant of the **Hessian Matrix**:
- $det(H(L(x,y,t))) = L_{xx}L_{yy} - L_{xy}^2$

Image Scale-Space: Interest Points

- Formally, the Hessian matrix is the square matrix of the second order partial derivatives of a function.
- Let $f(x_1, x_2, \ldots x_n)$ be a real-valued function. Its Hessian matrix is given by:

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1\,\partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1\,\partial x_n} \\ \frac{\partial^2 f}{\partial x_2\,\partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2\,\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n\,\partial x_1} & \frac{\partial^2 f}{\partial x_n\,\partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Hessian Matrix

- Exercise: Consider the following image.
  - a) Obtain its Hessian matrix.
  - b) Locate its edges
  - c) Locate its BLOBs

9	14	23	43	23	26	0
12	16	18	98	187	26	1
12	87	19	47	21	18	17
128	128	200	173	77	76	14
171	38	1	23	2	187	14
72	23	29	93	99	28	200