Integral and Discrete Transforms in Image Processing

Introduction & Some revision

David Svoboda

email: svoboda@fi.muni.cz
Centre for Biomedical Image Analysis
Faculty of Informatics, Masaryk University, Brno, CZ

CBIA

September 12, 2022

Outline

1. Introduction
   - Course rules
   - Course itinerary
   - Recommended reading

2. Some revision
   - Image transforms
   - Convolution
   - Impulse symbol
   - $\mathbb{C}$omplex numbers
   - Vector spaces

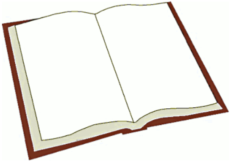
1 Introduction
- Course rules
- Course itinerary
- Recommended reading

2 Some revision
- Image transforms
- Convolution
- Impulse symbol
- $\mathbb{C}$omplex numbers
- Vector spaces

- 12 lectures & seminars
- Lecture + seminar = 2 + 2 hours per week.
- Final exam is written & spoken and is focused on your skills rather than knowledge.
- Basic knowledge of English and math (calculus, statistics, algebra) is highly recommended.
- Digital Image Processing (PV131) is highly recommended.
- Seminars take place in PC labs using Python.
- The experience from seminars will be useful for completing a small project written in Python, MATLAB®, C/C++, Java (or the preferred language).
- At the end of each lecture you can find a list of questions you should be able to answer if you want to pass the final exam.

1. Introduction & Revision
2. Fourier Transform, Spherical Harmonics, Hilbert Transform
3. Principle Component Analysis (PCA), Discrete Cosine Transform (DCT)
4. Singular Value Decomposition (SVD), Independent Component Analysis (ICA)
5. Image Resampling, Texture filtering
6. Z-transform
7. Wavelet Transform
8. Lifting Scheme
9. Recursive Filtering, Steerable Filters
10. Image Restoration
11. Image Compression Methods
12. Image Compression Standards

- Gonzalez, R. C., Woods, R. E., Digital image processing / 2nd ed., Upper Saddle River: Prentice Hall, 2002, pages 793, ISBN 0201180758
- Bracewell, R. N., Fourier transform and its applications / 2nd ed. New York: McGraw-Hill, pages 474, ISBN 0070070156
- Jähne, B., Digital image processing / 6th rev. and ext. ed., Berlin: Springer, 2005, pages 607, ISBN 3540240357
- selected papers



1 Introduction
- Course rules
- Course itinerary
- Recommended reading

2 Some revision
- Image transforms
- Convolution
- Impulse symbol
- $\mathbb{C}$omplex numbers
- Vector spaces

Image transform $\mathcal{T}$ is a function that converts the image from one vector space to another (or the same) vector space.

$$h = \mathcal{T}(f)$$

- $f(\mathbf{x})$ or $f(\mathbf{m})$ ... input image/signal
- $h(\mathbf{y})$ or $h(\mathbf{n})$ ... output image/signal
- $\mathcal{T}$ ... transform
- $\mathbf{x}, \mathbf{y}$ ... positions in continuous signal
- $\mathbf{m}, \mathbf{n}$ ... indices addressing the positions in discrete sequences/vectors

1D convolution

- Discrete: given two 1D signals $f(i)$ and $g(i)$:

$$(f * g)(i) \equiv \sum_k f(k)g(i-k)$$

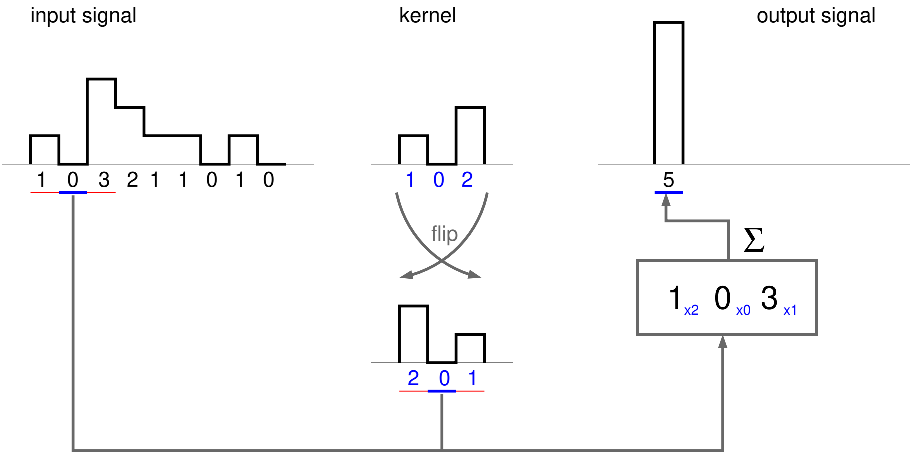
- Continuous: given two 1D signals $f(x)$ and $g(x)$:

$$(f * g)(x) \equiv \int_{-\infty}^{\infty} f(x')g(x-x')dx'$$

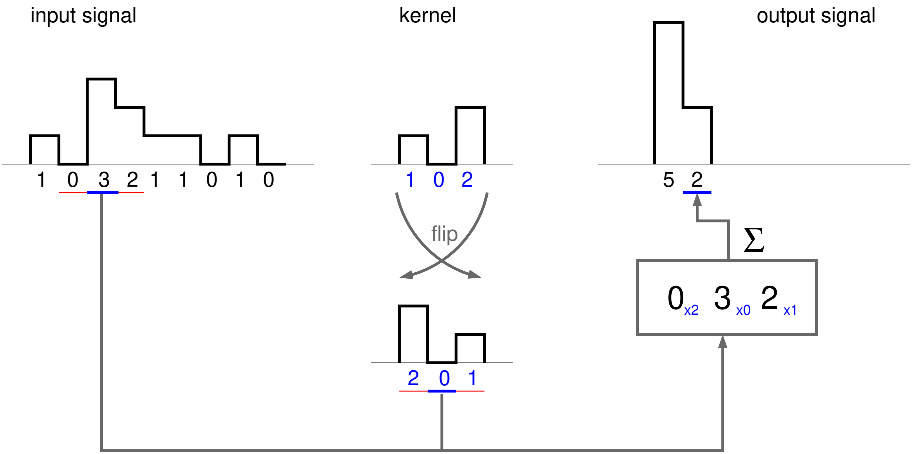
Notice: 'g' is called *a convolution kernel (mask)*

Convolution
Example

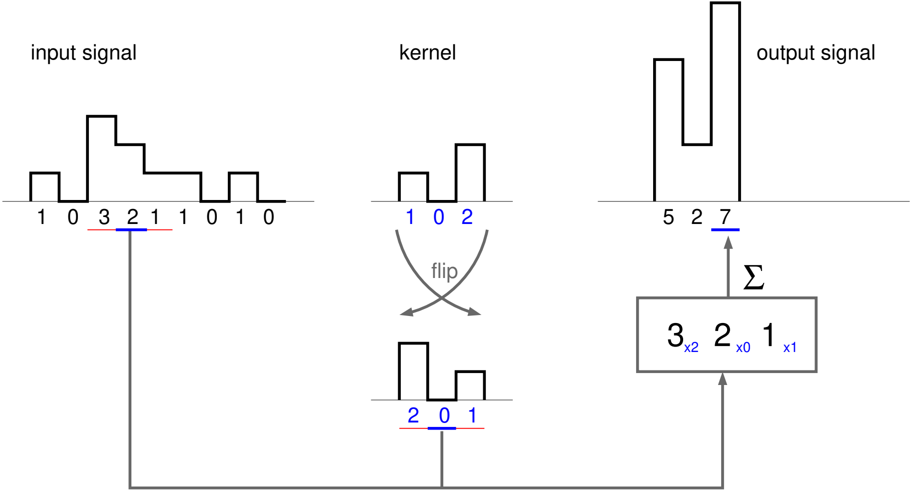
1D discrete convolution



1D discrete convolution



1D discrete convolution



2D convolution

- Discrete: given two 2D signals $f(i,j)$ and $g(i,j)$:

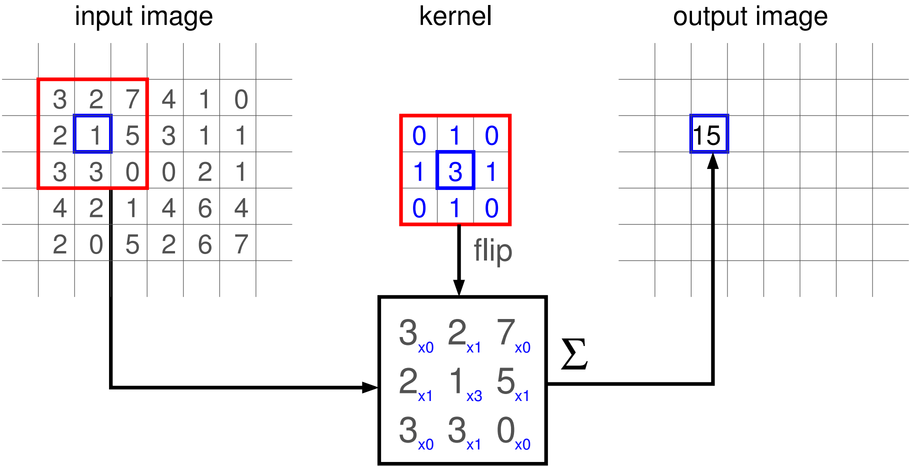
$$(f * g)(i,j) \equiv \sum_{k,l} f(k,l) g(i-k, j-l)$$

- Continuous: given two 2D signals $f(x,y)$ and $g(x,y)$:

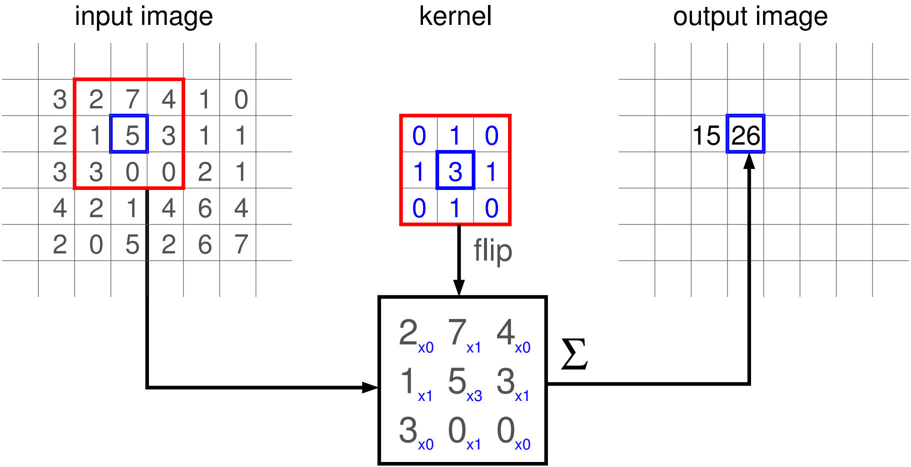
$$(f * g)(x,y) \equiv \int \int f(x',y') g(x-x', y-y') dx' dy'$$

Notice: If not necessary we will focus only on 1D discrete convolution.

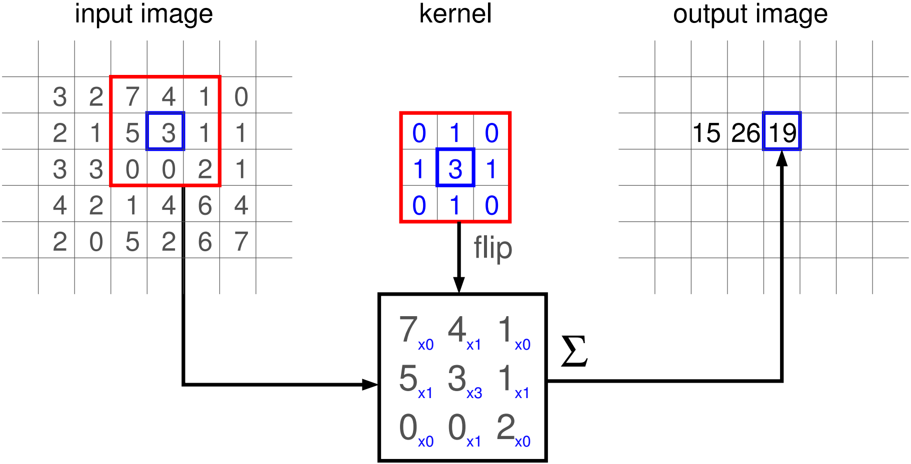
2D discrete convolution



2D discrete convolution



2D discrete convolution



Commutativity:

$$f * g = g * f$$

Distributivity:

$$f * (g + h) = f * g + f * h$$

Associativity:

$$(f * g) * h = f * (g * h)$$

Convolution theorem:

$$FT(f) \cdot FT(g) = FT(f * g)$$
$$FT(f) * FT(g) = FT(f \cdot g)$$

Separability:

2D kernel g is separable $\Leftrightarrow rank(g) = 1$

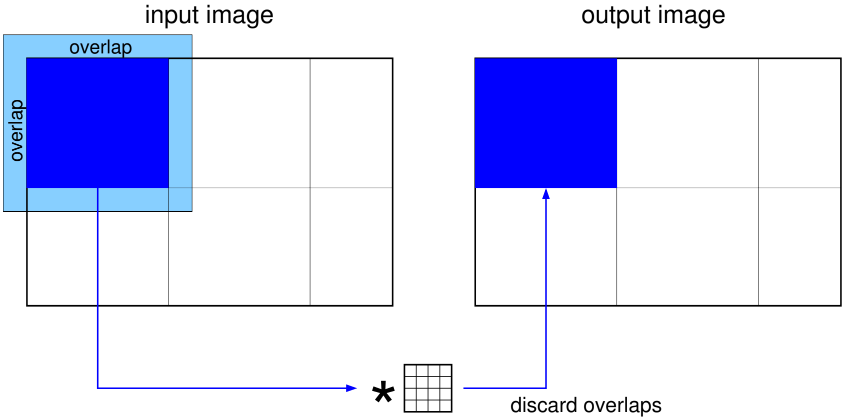
Notice: Expression '$FT()$' stands for Fourier transform.

Conditions

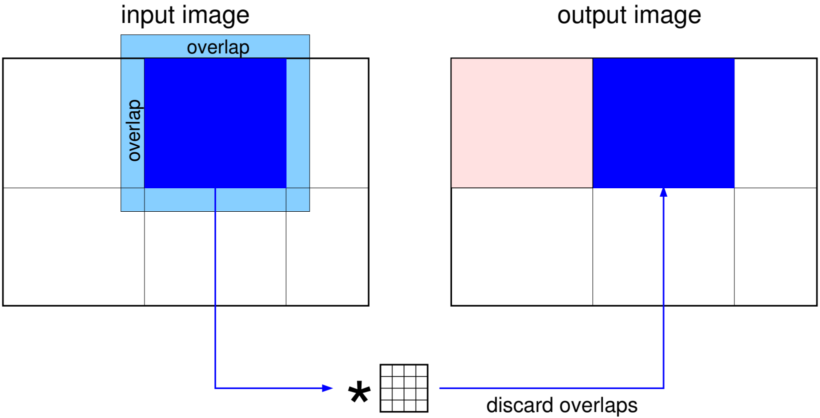
- input image $f : M \times M \rightarrow \{0, \dots, 2^{\text{bitdepth}}\}$
- convolution kernel $g : N \times N \rightarrow \langle 0; 1 \rangle$

- Standard/Naive 2D discrete convolution
  - Time: $O(M^2 N^2)$
  - Space: none required
  - Usability: Very slow.
- 2D discrete convolution with separable kernel
  - Time: $O(M^2 N)$
  - Space $O(M^2)$
  - Usability: Not all PSFs are separable.
- Convolution theorem
  - Time: $O((M+N)^2 log(M+N))$
  - Space: $O((M+N)^2)$
  - Usability: Huge memory requirements.

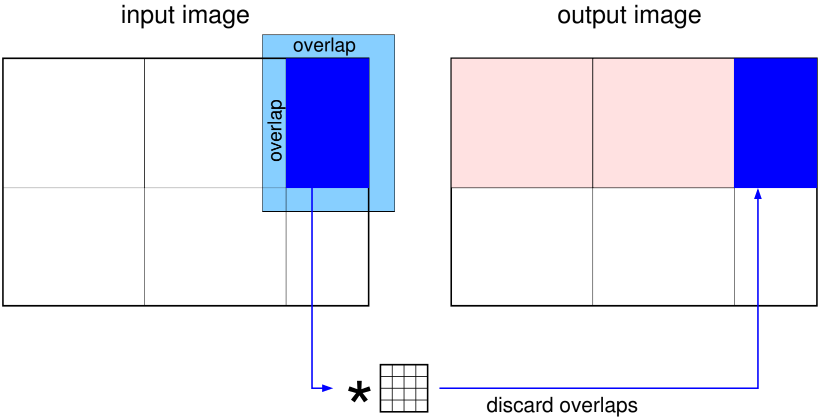
"Overlap & Save" scheme



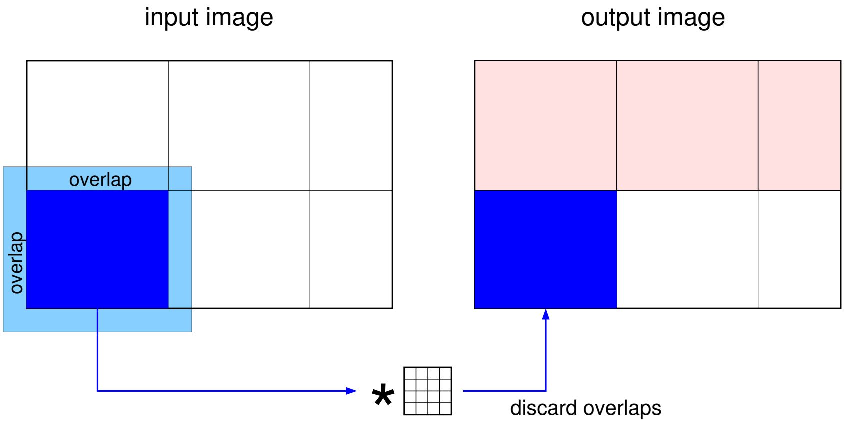
"Overlap & Save" scheme



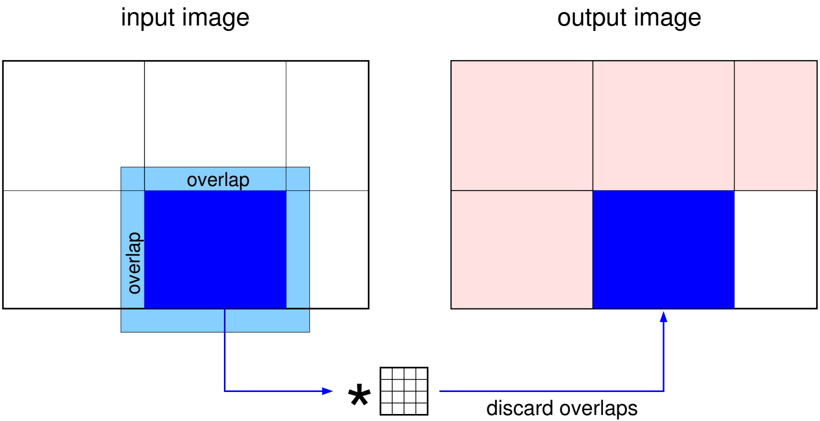
"Overlap & Save" scheme



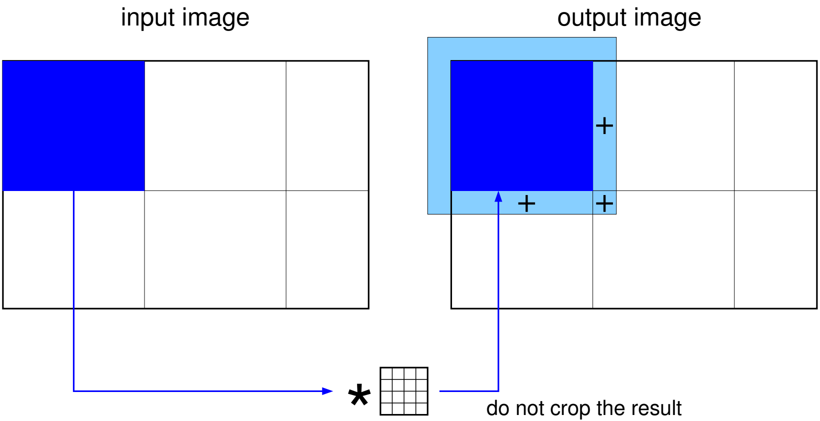
"Overlap & Save" scheme



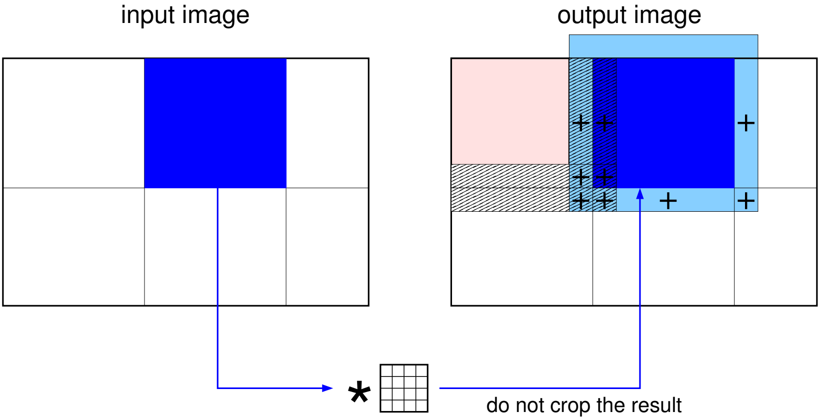
"Overlap & Save" scheme



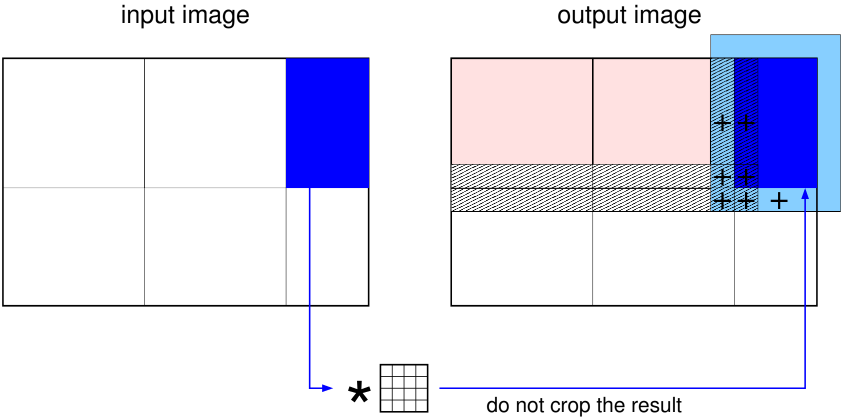
"Overlap & Add" scheme



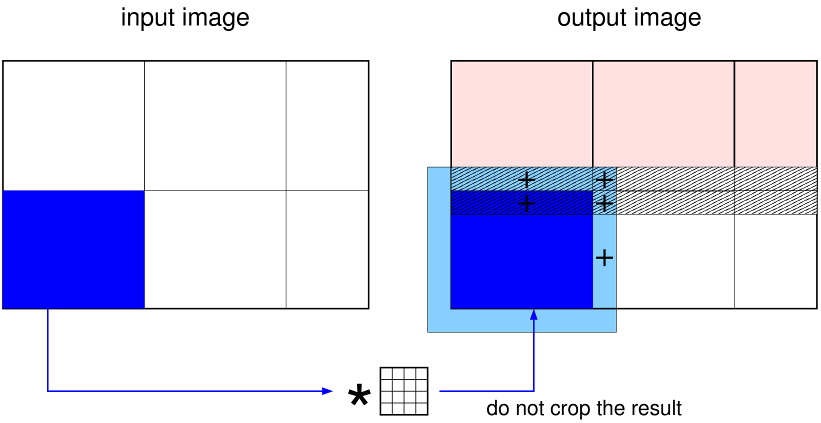
"Overlap & Add" scheme



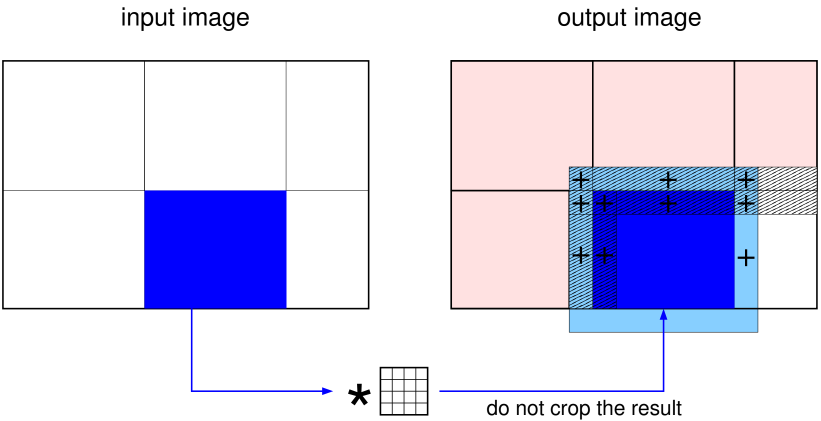
"Overlap & Add" scheme



"Overlap & Add" scheme



"Overlap & Add" scheme



Impulse symbol δ

Definition

Infinitely brief and infinitely strong unit-area impulse:

$$\delta(x) = 0 \quad x \neq 0$$

and

$$\int_{-\infty}^{\infty} \delta(x) dx = 1$$

- we call it Dirac delta function or impulse symbol
- it is NOT a function

Impulse symbol δ

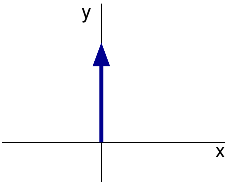
Properties

Given 1D function f and $a \in \mathbb{R}$:

$$\int_{-\infty}^{\infty} \delta(x) f(x) dx = f(0)$$

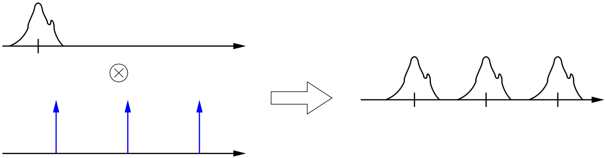
$$\int_{-\infty}^{\infty} \delta(x-a) f(x) dx = f(a)$$

$\delta(x)$ plot:



Convolution of any function f with:

- δ impulse shifts the origin of f to the nonzero response of δ
- δ impulses replicate the function f



- Gaussian shifts the origin of f to the position of the peak of the Gaussian and smooths

Kronecker delta (function)

Kronecker delta function . . . discrete counterpart to Dirac delta impulse.

$$\delta_{i,j} = \begin{cases} 1 & \text{if } (i = j) \\ 0 & \text{otherwise} \end{cases}$$

or

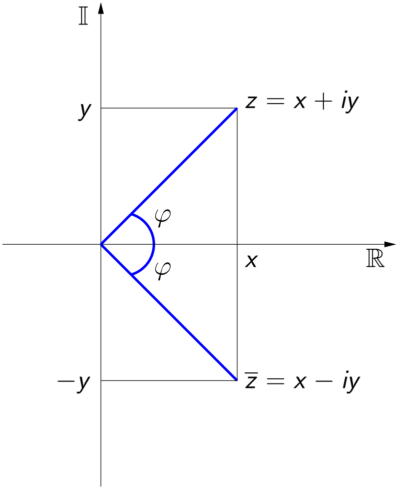
$$\delta(n) = \begin{cases} 1 & \text{if } (n = 0) \\ 0 & \text{otherwise} \end{cases}$$

$\mathbb{C}$omplex numbers

Any $z \in \mathbb{C}$ can be written in one of the following ways:

- $z = x + iy$
- $z = |z| (\cos \varphi + i \sin \varphi)$
- $z = |z| e^{i\varphi}$

where $x, y \in \mathbb{R}$ and $i^2 = -1$ is a constant, $|z|$ is a magnitude and φ is a phase of z

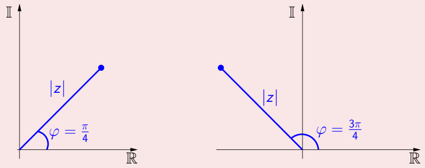


Properties:

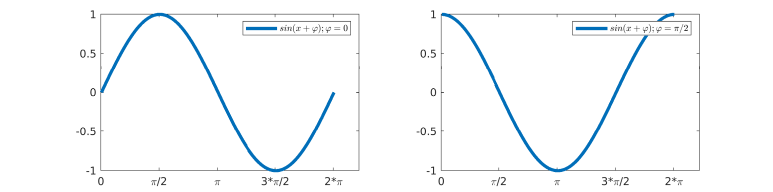
- conjugate complex number: $\overline{z} = x - iy$

Be aware of the difference!

- φ – phase (of complex number)

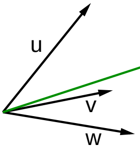


- φ – phase shift (of a function)

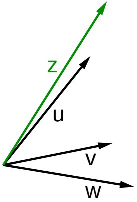


Let be given a Euclidean ($\mathbb{K} = \mathbb{R}$) or unitary ($\mathbb{K} = \mathbb{C}$) vector space $\mathbb{V} \subseteq \mathbb{K}^n$ and three vectors $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{V}$:

- Vector addition: $\mathbf{z} = \mathbf{u} + \mathbf{v} + \mathbf{w} \in \mathbb{V}$



- Linear combination of vectors: $\mathbf{z} = \frac{1}{2}\mathbf{u} + 3\mathbf{v} - 2\mathbf{w} \in \mathbb{V}$



Let be given Euclidean space $\mathbb{V} = \langle \mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_n} \rangle$, then each $\mathbf{v} \in \mathbb{V}$ can be written as:

$$\mathbf{v} = a_1\mathbf{u_1} + a_2\mathbf{u_2} + \cdots + a_n\mathbf{u_n}$$

where

- $(\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_n})$ is the basis of $\mathbb{V}$
- $\forall i = \{1, \ldots, n\} : a_i \in \mathbb{K}$
- vector $(a_1, a_2, \ldots, a_n)$ is unique.

Notes:

- two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{V}$ are orthogonal, if $\mathbf{u} \cdot \mathbf{v} = 0$
  ('$\cdot$' stands to inner product)
- basis $(\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_n})$ is orthonormal, if $\forall i, j = 1, \ldots, n : \mathbf{u_i} \cdot \mathbf{u_j} = \delta_{i,j}$
  ($\delta_{i,j}$ stands for Kronecker delta)

Vectors

Example

Given Cartesian coordinate system $\langle \mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3} \rangle$ and a vector $\mathbf{v} = (3.4, -2, 7)$, we can write:

$$\mathbf{v} = 3.4\mathbf{e_1} - 2\mathbf{e_2} + 7\mathbf{e_3}$$

where

$$\begin{aligned} \mathbf{e_1} &= (1, 0, 0) \\ \mathbf{e_2} &= (0, 1, 0) \\ \mathbf{e_3} &= (0, 0, 1) \end{aligned}$$

Question: How to find the (linear combination) coefficients when we do not project the vector $\mathbf{v}$ onto standard basis?

Given a vector $\mathbf{v} \in \mathbb{V}$ and "any" basis $(\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_n})$ in $\mathbb{V}$, we can write:

$$\mathbf{v} = a_1\mathbf{u_1} + a_2\mathbf{u_2} + \cdots + a_n\mathbf{u_n}$$

where

$$\forall i = \{1, \ldots, n\} : a_i = \frac{\mathbf{v} \cdot \mathbf{u_i}}{\mathbf{u_i} \cdot \mathbf{u_i}}$$

If the basis is orthonormal, it is sufficient to write: $a_i = \mathbf{v} \cdot \mathbf{u_i}$

Notice: Inner product $\mathbf{v} \cdot \mathbf{w}$ is a projection $\mathbf{v}$ onto $\mathbf{w}$. The result is a number.

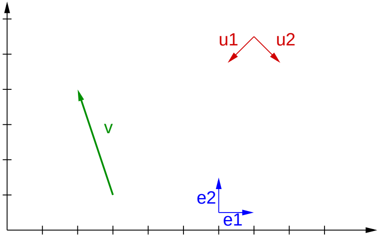
Vectors

Example

- standard basis: $\langle \mathbf{e_1}, \mathbf{e_2} \rangle = \langle (1,0), (0,1) \rangle$
  $\mathbf{v}_{\langle \mathbf{e_1}, \mathbf{e_2} \rangle} = (-1, 3)$
- another basis: $\langle \mathbf{u_1}, \mathbf{u_2} \rangle = \langle (-0.7, -0.7), (0.7, -0.7) \rangle$ $\quad (0.7 \doteq \frac{\sqrt{2}}{2})$

$$
\begin{aligned}
a_1 &= \frac{(-1,3)\cdot(-0.7,-0.7)}{(-0.7,-0.7)\cdot(-0.7,-0.7)} \doteq -1.42 \\
a_2 &= \frac{(-1,3)\cdot(0.7,-0.7)}{(0.7,-0.7)\cdot(0.7,-0.7)} \doteq -2.86
\end{aligned}
$$

$\mathbf{v}_{\langle \mathbf{u_1}, \mathbf{u_2} \rangle} = (-1.42, -2.86)$



Vectors

Example (cont'd)

Each orthonormal basis forms a square matrix:

$$A = \begin{bmatrix} \mathbf{u_1} \\ \mathbf{u_2} \end{bmatrix} = \begin{bmatrix} -0.7 & -0.7 \\ 0.7 & -0.7 \end{bmatrix}$$

The projection is therefore realized using matrix multiplication:

$$\mathbf{v}_{\langle \mathbf{u_1}, \mathbf{u_2} \rangle} = A\mathbf{v}_{\langle \mathbf{e_1}, \mathbf{e_2} \rangle}$$

Notice: Transform from one basis onto another one is a linear mapping.

Properties of transform matrices:

- A is unitary matrix, i.e. $A^{-1} = \overline{A}^T$.
- If $y = Ax$ is forward transform, then $x = A^{-1}y = \overline{A}^T y$ is inverse transform.

The following two sentences express the same:

- The vector $\mathbf{v}$ is projected into the basis $\langle \mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_n} \rangle$.
- The vector $\mathbf{v}$ is decomposed into a linear combination of basis vectors $\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_n}$

Decomposition in higher dimensions

Let's increase the dimensionality: 2-D $\rightarrow$ 3-D $\rightarrow \ldots \rightarrow$ N-D

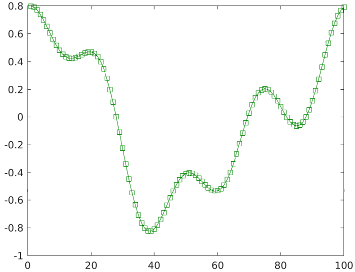
Let $\mathbb{V} \subseteq \mathbb{K}^N$ be a N-dimensional vector space ($\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$), then the following terms express the same:

- a vector $v \in \mathbb{V}$
  example: $v = (4, 2, 3, 6, -1, 5)$
- a point $p \in \mathbb{V}$
  example: $p = [4, 2, 3, 6, -1, 5]$
- a discrete 1D function f, where $|f| = N$
  example: $f = \{(0, 4), (1, 2), (2, 3), (3, 6), (4, -1), (5, 5)\}$

Notice: The addition of points/vectors equals to addition of functions. The same for other well known operations like subtraction, multiplication by scalar, or linear combination.

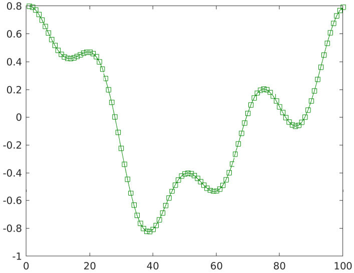
Decomposition in higher dimensions
Example

The discrete function below can be understood as a vector/point in 100-dimensional vector space:

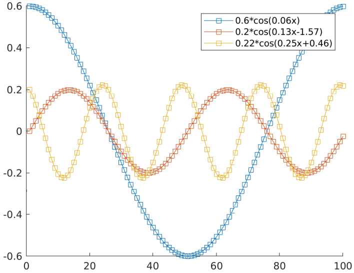


Notice: In higher dimensions, we use rather the concept of *functions*.

The discrete function below can be decomposed into a linear combination of some simple (well known) functions:



$\Rightarrow$

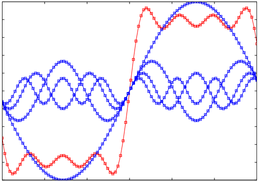


Notice: Each discrete function can be decomposed in this manner.

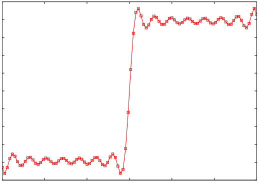
A step function (red color) is defined as an infinite sum of sine waves:

$$f_z(m) = \sum_{n=0}^{z} \frac{sin\left\{(2n+1)m\right\}}{2n+1}$$

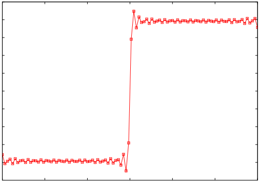
$f_3(m)$



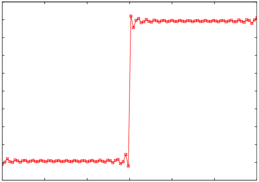
$f_{10}(m)$



$f_{20}(m)$



$f_{35}(m)$



Decomposition in higher dimensions

Let be a discrete 1D function f of N samples:

- f can be uniquely expressed as a linear combination of basis functions $\varphi_1, \varphi_2, \dots, \varphi_N$:

$$f(m) = \sum_{k=1}^{N} a_k \varphi_k(m)$$

where $a_k \in \mathbb{K}$ and $(\varphi_1, \varphi_2, \dots, \varphi_N)$ form the orthonormal basis

The coefficients of linear combination are found as:

$$\forall k = \{1, \dots, N\} : a_k = f \cdot \varphi_k$$

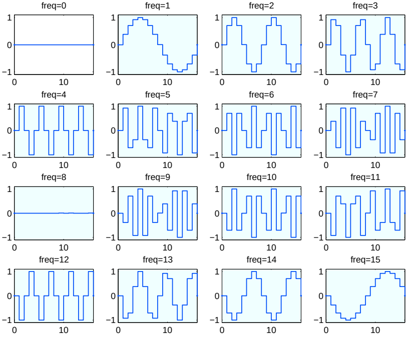
i.e. using the projection (inner/dot product)

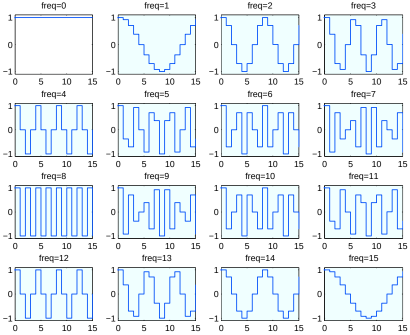
Notice: $f \cdot \varphi_k = \sum_m f(m)\overline{\varphi_k(m)}$

Common request:

- the basis is orthonormal, i.e. $\varphi_k \cdot \varphi_l = \delta_{k,l}$
- the basis functions for $N = 16$ are:

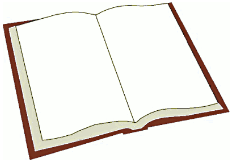
$$\varphi_k(m) = \frac{1}{\sqrt{N}} e^{\frac{-2\pi imk}{N}} = \frac{1}{\sqrt{N}} \left(\cos \frac{2\pi mk}{N} - i \sin \frac{2\pi mk}{N} \right)$$





Bibliography

- Gonzalez, R. C., Woods, R. E., Digital image processing / 2nd ed., Upper Saddle River: Prentice Hall, 2002, pages 793, ISBN 0201180758
- Bracewell, R. N., Fourier transform and its applications / 2nd ed. New York: McGraw-Hill, pages 474, ISBN 0070070156



You should know the answers . . .

- What happens if we convolve a function f with Gaussian located outside the origin?
- What is the result when convolving a function f with several δ impulses?
- Under which conditions is the convolution kernel separable?
- What is the *basis* and *vector space* generated by the given basis?
- What are the orthogonal vectors?
- What is the orthonormal basis?
- How can we simply convert a vector from one basis to another basis?
- What is the unitary/orthogonal matrix?
- What is the difference between *basis vector* and *basis function*?