Digital Image

(B4M33DZO, Summer 2024)

Lecture 3:

Convolution

https://cw.fel.cvut.cz/wiki/courses/b4m33dzo/start

Daniel Sýkora & Ondřej Drbohlav

Department of Cybernetics

Faculty of Electrical Engineering

Czech Technical University in Prague

Convolution

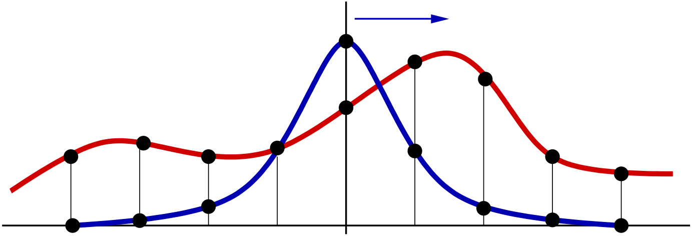
Sliding average of function f weighted by function g:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(x) \cdot g(t-x)\,\mathrm{d}x$$

Convolution

Sliding average of function f weighted by function g:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(x) \cdot g(t-x)\, \mathrm{d}x$$



Interesting Properties

Commutativity:

$$f * g = g * f$$

Interesting Properties

Commutativity:

$$f * g = g * f$$

Associativity:

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

Interesting Properties

Commutativity:

$$f * g = g * f$$

Associativity:

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

Distributivity:

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

Interesting Properties

Commutativity:

$$f * g = g * f$$

Associativity:

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

Distributivity:

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

Differentiation:

$$(f * g)' = f' * g = f * g'$$

Interesting Properties

Commutativity:

$$f * g = g * f$$

Associativity:

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

Distributivity:

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

Differentiation:

$$(f * g)' = f' * g = f * g'$$

Convolution theorem:

$$f * g = \mathcal{F}^{-1}\{\mathcal{F}\{f\} \cdot \mathcal{F}\{g\}\}$$

Convolution (in 2D)

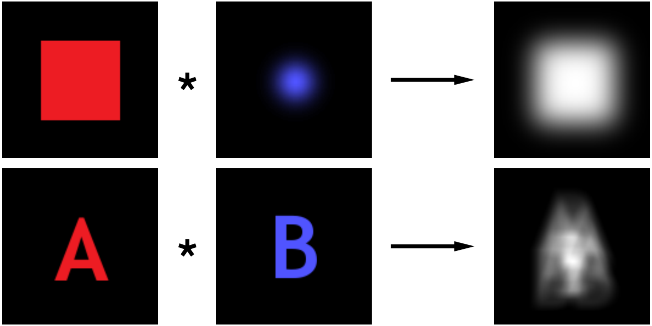
Extension of 1D case (image F, convolution kernel G):

$$(F * G)(s,t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(x,y) \cdot G(s-x, t-y)\, \mathrm{d}x\mathrm{d}y$$

Convolution (in 2D)

Extension of 1D case (image F, convolution kernel G):

$$(F * G)(s,t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(x,y)\cdot G(s-x,t-y)\,\mathrm{d}x\mathrm{d}y$$



Convolution (in 2D)

Discrete case (image F, convolution kernel G):

$$(\mathrm{F} * \mathrm{G})[s,t] = \sum_x \sum_y \mathrm{F}[x][y] \cdot \mathrm{G}[s-x][t-y]$$

Convolution (in 2D)

Discrete case (image F, convolution kernel G):

$$(\mathrm{F} * \mathrm{G})[s,t] = \sum_{x} \sum_{y} \mathrm{F}[x][y] \cdot \mathrm{G}[s-x][t-y]$$

Computational complexity: $\mathcal{O}(|\mathrm{F}| \cdot |\mathrm{G}|)$

Convolution (in 2D)

Discrete case (image F, convolution kernel G):

$$(\mathrm{F} * \mathrm{G})[s,t] = \sum_x \sum_y \mathrm{F}[x][y] \cdot \mathrm{G}[s-x][t-y]$$

Computational complexity: $\mathcal{O}(|\mathrm{F}| \cdot |\mathrm{G}|)$

Problem: intractable for large $|\mathrm{G}|$ even on GPU!

How to compute it faster?

Convolution (in 2D)

Discrete case (image F, convolution kernel G):

$$(\mathrm{F} * \mathrm{G})[s,t] = \sum_x \sum_y \mathrm{F}[x][y] \cdot \mathrm{G}[s-x][t-y]$$

Computational complexity: $\mathcal{O}(|\mathrm{F}| \cdot |\mathrm{G}|)$

Problem: intractable for large $|\mathrm{G}|$ even on GPU!

How to compute it faster?

method	accuracy	kernel	complexity
Fourier transform	exact	arbitrarily	$\mathcal{O}(\|\mathrm{F}\| \cdot \log \|\mathrm{F}\|)$
Separable kernel	exact	limited	$\mathcal{O}(\|\mathrm{F}\| \cdot \sqrt{\|\mathrm{G}\|})$
Integral image	high	limited	$\mathcal{O}(\|\mathrm{F}\|)$
Mip-mapping	low	arbitrarily	$\mathcal{O}(\|\mathrm{F}\|)$

Fourier Transform

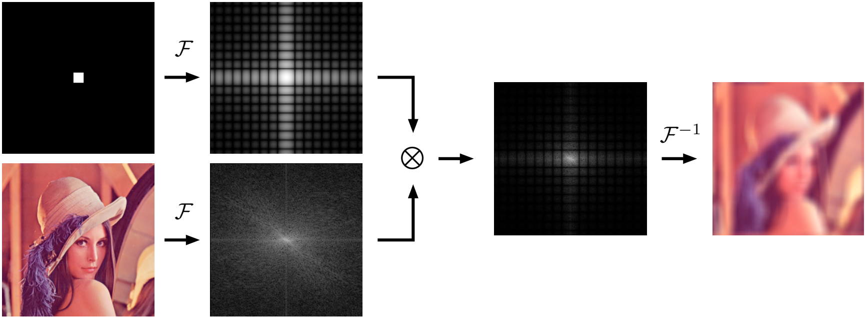
Convolution theorem:

$$\mathcal{F}\{F * G\} = \mathcal{F}\{F\} \cdot \mathcal{F}\{G\}$$

Fourier Transform

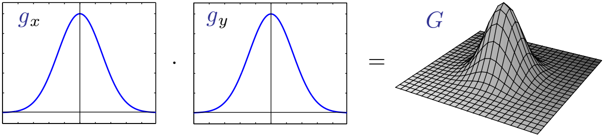
Convolution theorem:

$$\mathcal{F}\left\{F * G\right\} = \mathcal{F}\left\{F\right\} \cdot \mathcal{F}\left\{G\right\}$$



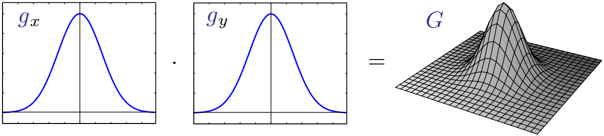
Separable Kernel

$$G(x, y) = g_x(x) \cdot g_y(y)$$



Separable Kernel

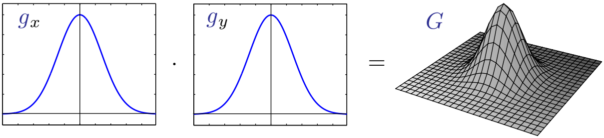
$$G(x,y) = g_x(x) \cdot g_y(y)$$



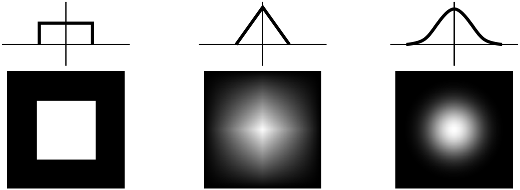
$$(F * G)(s,t) = \int_{-\infty}^{\infty} g_y(t-y) \cdot \left(\int_{-\infty}^{\infty} F(x,y) \cdot g_x(s-x)\,\mathrm{d}x \right) \mathrm{d}y$$

Separable Kernel

$$G(x,y) = g_x(x) \cdot g_y(y)$$

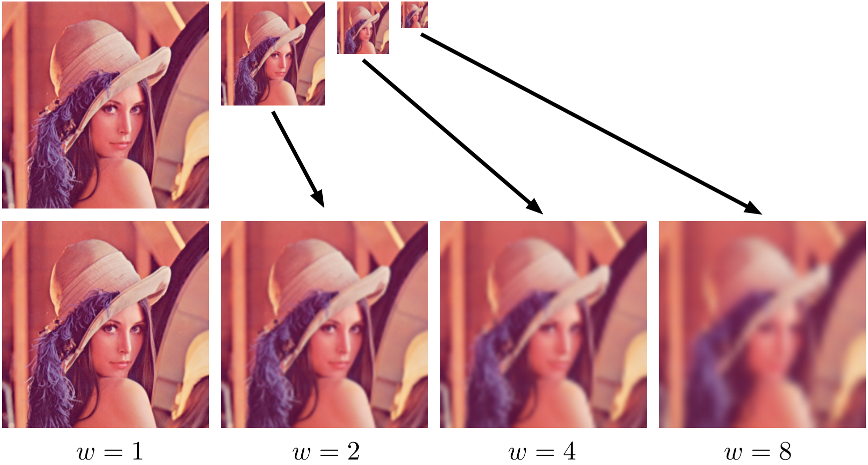


$$(F * G)(s,t) = \int_{-\infty}^{\infty} g_y(t-y) \cdot \left(\int_{-\infty}^{\infty} F(x,y) \cdot g_x(s-x)\,\mathrm{d}x \right) \mathrm{d}y$$



Mip-Mapping

Precalculate output for increasing kernel size:

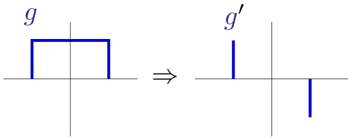


Integral Image

$$f * g = \int f * g'$$

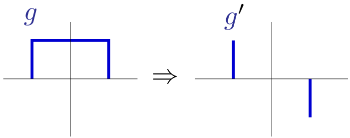
Integral Image

$$f * g = \int f * g'$$



Integral Image

$$f * g = \int f * g'$$



2	3	2	1
3	0	1	2
1	3	1	0
1	4	2	2

image I

Summed area table:

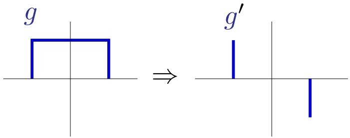
$$\mathrm{S}[x,y] = \sum_x \sum_y \mathrm{I}[x,y]$$

2	5	7	8
5	8	11	14
6	12	16	19
7	17	23	28

SAT S

Integral Image

$$f * g = \int f * g'$$



2	3	2	1
3	0	1	2
1	3	1	0
1	4	2	2

image I

Summed area table:

$$\mathrm{S}[x,y] = \sum_x \sum_y \mathrm{I}[x,y]$$

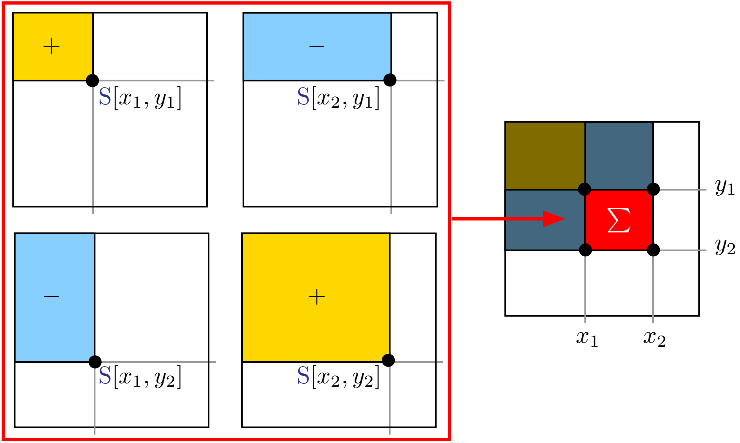
2	5	7	8
5	8	11	14
6	12	16	19
7	17	23	28

SAT S

$$\mathrm{S}[x,y] = \mathrm{I}[x,y] + \mathrm{S}[x-1,y] + \mathrm{S}[x,y-1] - \mathrm{S}[x-1,y-1]$$

Integral Image

$$\sum = \mathrm{S}[x_2, y_2] - \mathrm{S}[x_1, y_2] - \mathrm{S}[x_2, y_1] + \mathrm{S}[x_1, y_1]$$



Integral Image

Extension for more complicated kernels:

$$f * g = \int^{(n)} f * g^{(n)}$$

Integral Image

Extension for more complicated kernels:

$$f * g = \int^{(n)} f * g^{(n)}$$

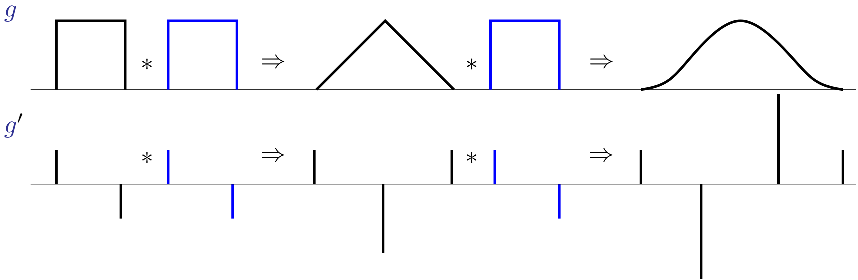
$$(\underbrace{g * g * \ldots * g}_{n})^{(n)} = \underbrace{g' * g' * \ldots * g'}_{n}$$

Integral Image

Extension for more complicated kernels:

$$f * g = \int^{(n)} f * g^{(n)}$$

$$\big(\underbrace{g * g * \ldots * g}_{n}\big)^{(n)} = \underbrace{g' * g' * \ldots * g'}_{n}$$



Integral Image

Examples of convolution kernels for integral image:

-1	1
1	-1

1	-2	1
-2	4	-2
1	-2	1

-1	3	-3	1
3	-9	9	-3
-3	9	-9	3
1	-3	3	-1