Insper

Computer Vision

Class 8: The Scale Space and Convolutional Layers

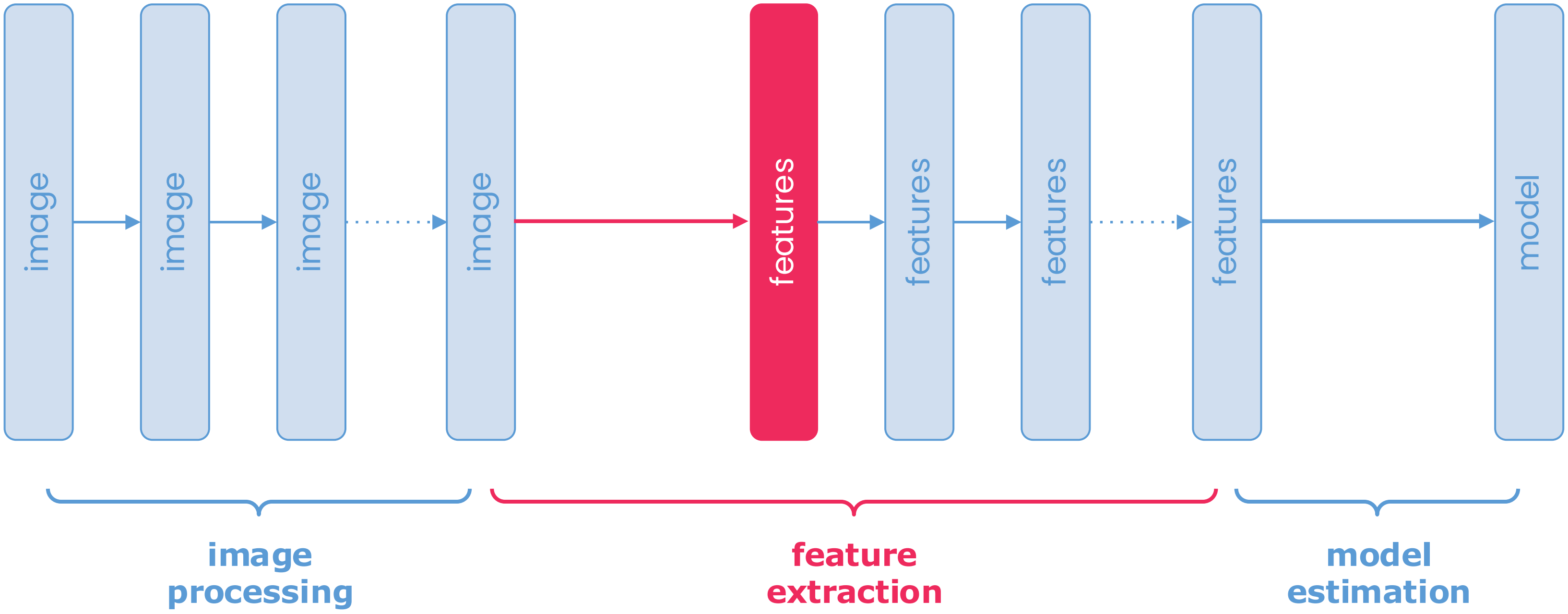


image
image
image
image
features
features
features
features
model
image processing
feature extraction
model estimation

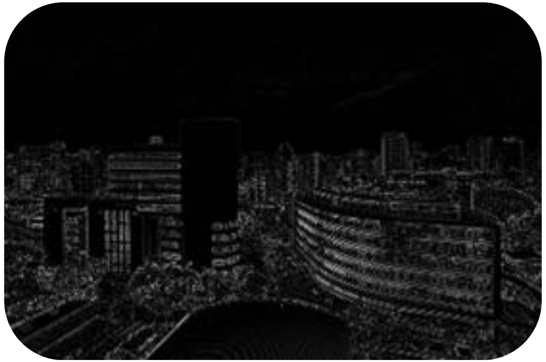
Laplace filter:
convolution with the Laplace kernel

normalized absolute values

$$P$$

0	1	0
1	-4	1
0	1	0

$$L$$



$$\nabla^2 P = P * L$$

Edge detection is...

- ...robust to gray level reduction. *(mostly)*
- ...robust to brightness changes. *(within reason)*
- ...robust to contrast changes. *(within reason)*
- ...robust to occlusion. *(locally)*
- ...robust to translation.
- ...robust to rotation.
- ...robust to perspective distortion. *(locally)*
- ...robust to deformation. *(locally)*

Edge detection is not...

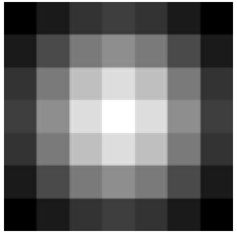
- ...robust to noise.
- ...robust to scale.

Edge detection is not...

- ...robust to noise, but we can mitigate that with image smoothing.
- ...robust to scale.

Image smoothing:
convolution with a kernel of weights

P



G

$S = P * G$

Laplacian of Gaussian:

convolution of a Gaussian kernel with a Laplacian kernel

$$LoG(x, y) = -\frac{1}{\pi\sigma^4}\left(1 - \frac{x^2+y^2}{2\sigma^2}\right) e^{-\frac{x^2+y^2}{2\sigma^2}}$$

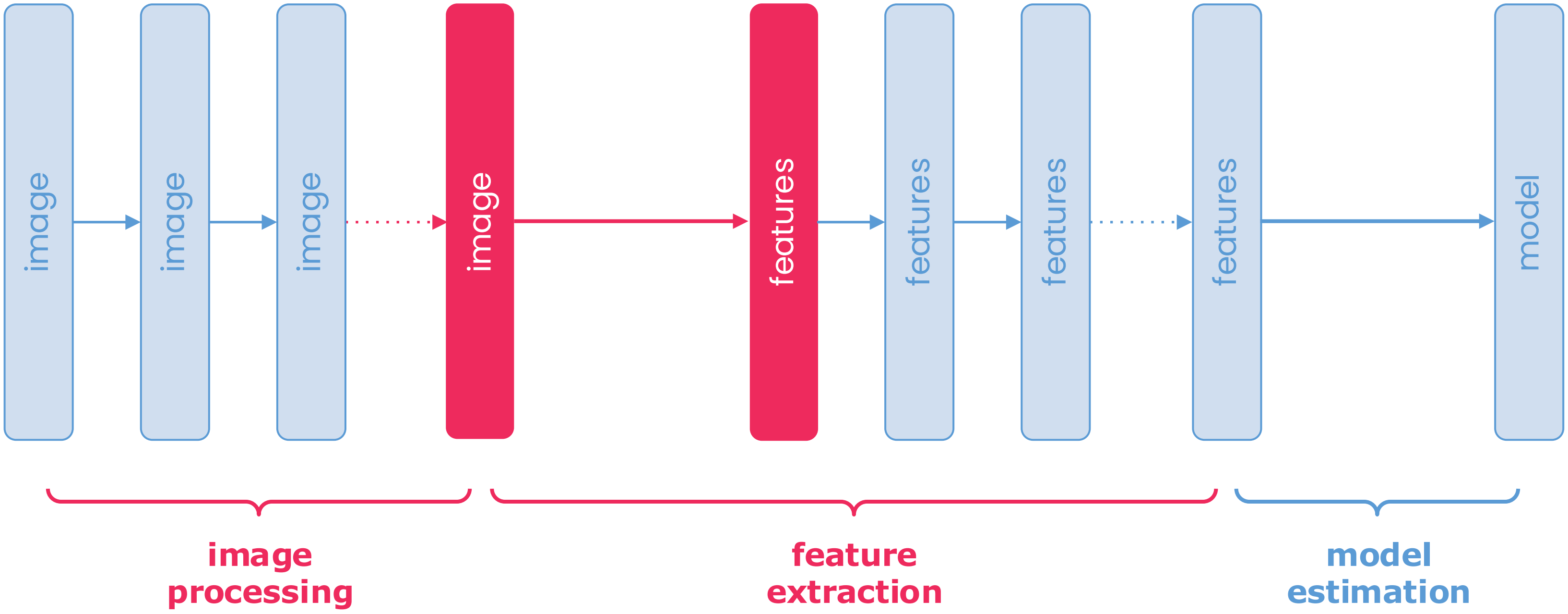


image
image
image
image
features
features
features
features
model
image processing
feature extraction
model estimation

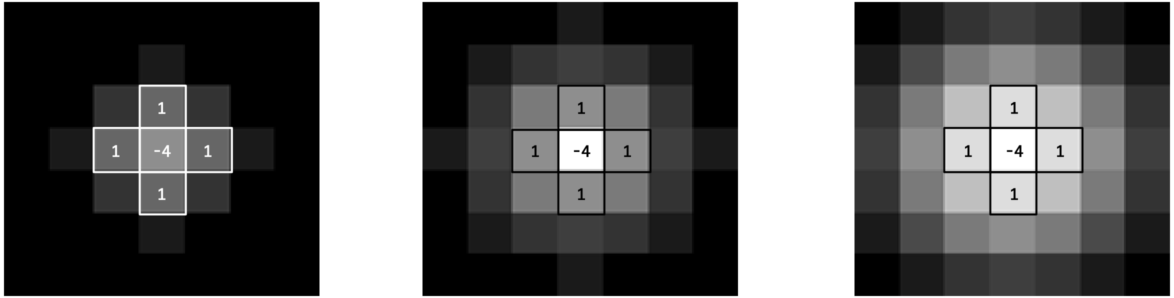
Edge detection is not…

- …robust to noise, but we can mitigate that with image smoothing.
- …robust to scale and… well… there’s not much we can do about it.

The Goldilocks problem:

detection of discontinuities depends on the scale

low absolute Laplacian | high absolute Laplacian | low absolute Laplacian



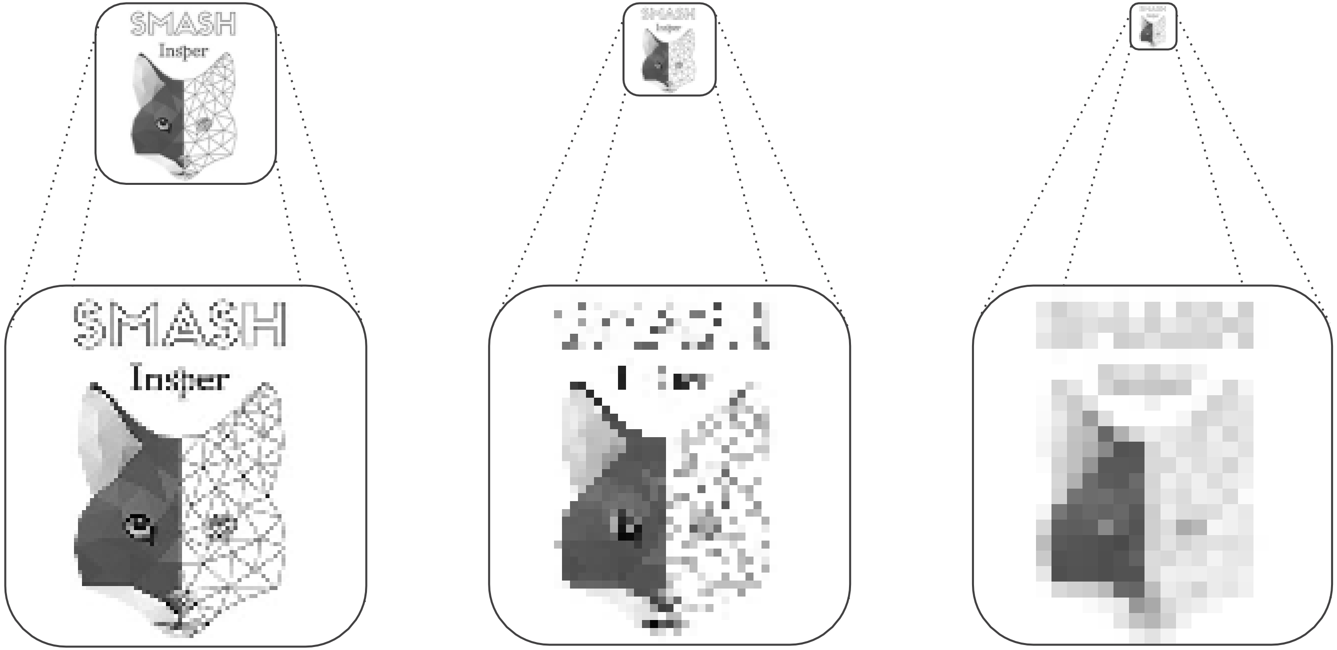
"too small" | "just right" | "too large"

But, before we start thinking about "super features" that are robust to everything...

…let's try something that is a little simpler.

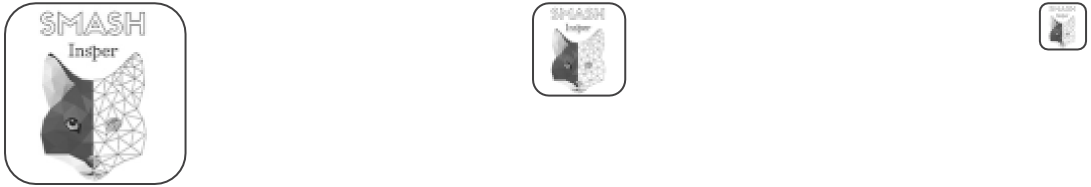
Multi-scale extraction:

features from multiple scales of the same image



The scale space:

increasing Gaussian smoothings as approximations of scale



S_σ $\qquad$ $S_{2\sigma}$ $\qquad$ $S_{3\sigma}$

Two approaches:

1. detect discontinuities in all scales and consolidate the detections; *(minimum, maximum, etc.)*

Simple example:

maximum of absolute Laplacians

S_σ

$S_{2\sigma}$

$S_{3\sigma}$

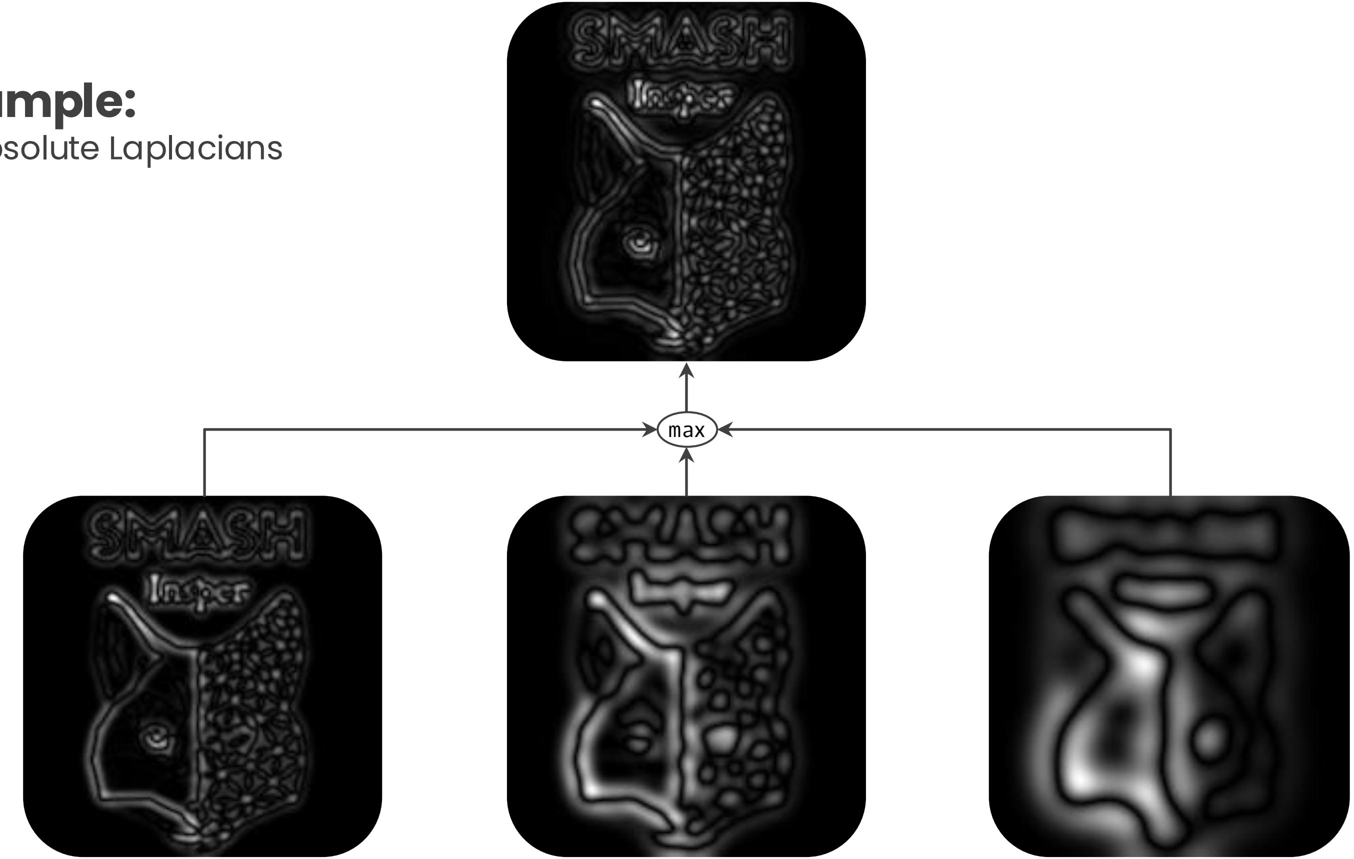
Simple example:
maximum of absolute Laplacians

$|\nabla^2 S_\sigma|$

$|\nabla^2 S_{2\sigma}|$

$|\nabla^2 S_{3\sigma}|$

Simple example:
maximum of absolute Laplacians



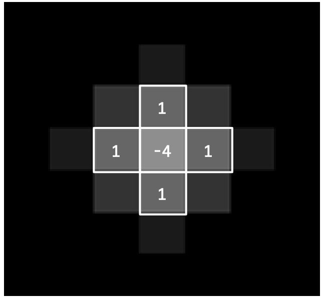
Two approaches:

1. detect discontinuities in all scales and consolidate the detections; *(minimum, maximum, etc.)*

1. detect discontinuities in both space (x, y) and scale (σ).

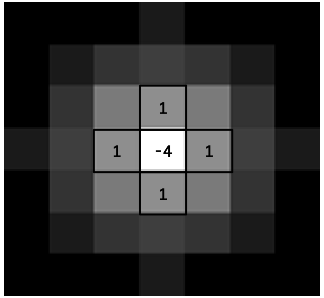
discontinuity in scale

low absolute Laplacian



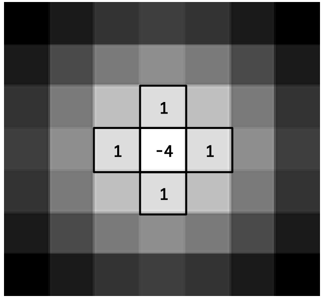
$S_{3\sigma}$

high absolute Laplacian



$S_{2\sigma}$

low absolute Laplacian



S_{σ}

Simple example:
absolute Laplacian of Laplacians

S_{σ}

$S_{2\sigma}$

$S_{3\sigma}$

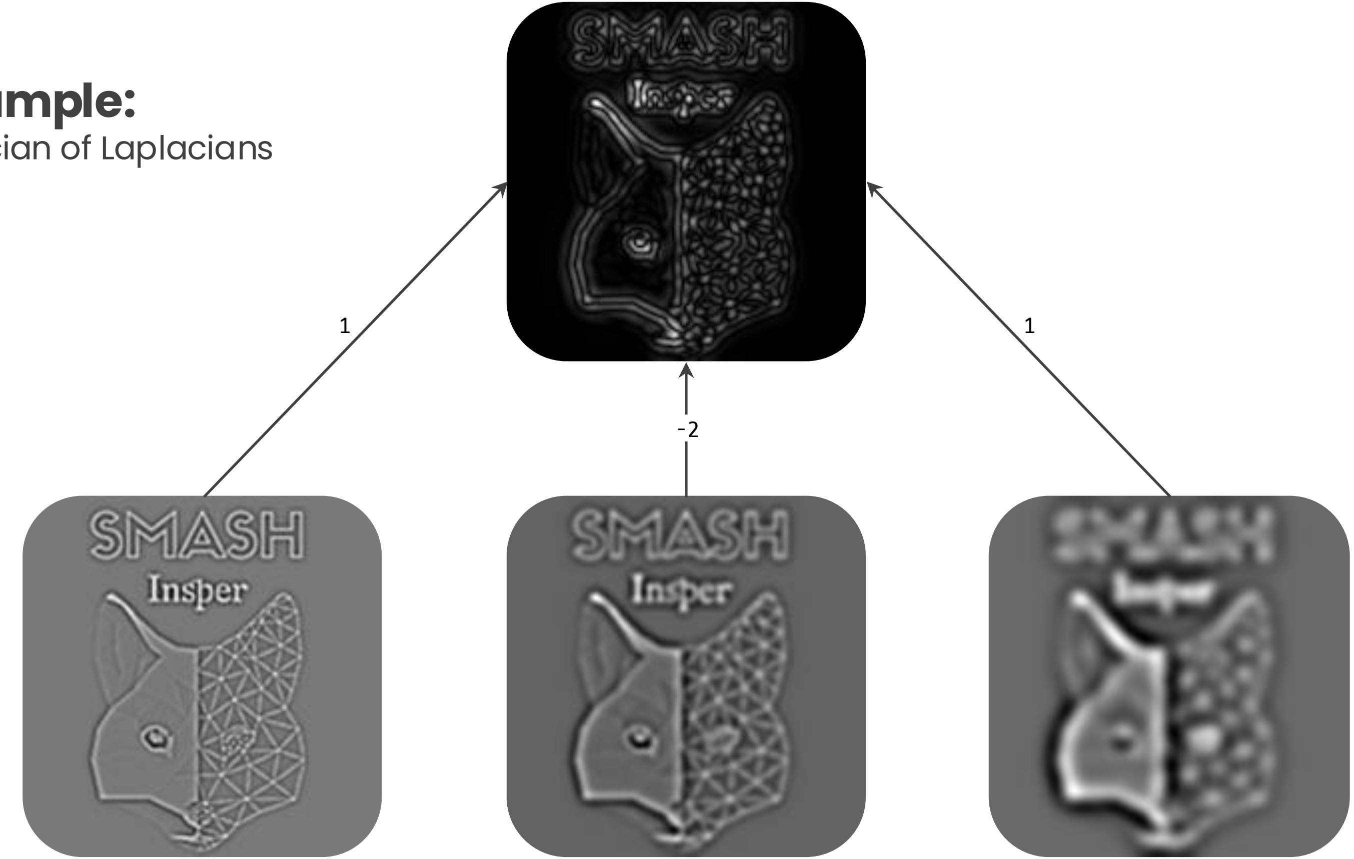
Simple example:
absolute Laplacian of Laplacians

$\nabla^2 S_\sigma$

$\nabla^2 S_{2\sigma}$

$\nabla^2 S_{3\sigma}$

Simple example:
absolute Laplacian of Laplacians



Scale space approaches can be applied to any feature detector...

...but the Laplacian in particular has a very interesting property.

Unsharp mask:
a simple approach to enhance details…

normalized signed values

$$S \quad - \quad P \quad = \quad U$$

Unsharp mask:
a simple approach to enhance details...

normalized signed values

$$S \quad - \quad U \quad = \quad P$$

Unsharp mask:
a simple approach to enhance details...

normalized signed values

$$S \quad - \quad 2U \quad = \quad P - U$$

Unsharp mask:

a simple approach to enhance details…

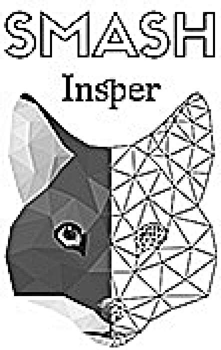
normalized signed values

$$S \quad - \quad 3U \quad = \quad P - 2U$$

Unsharp mask:

a simple approach to enhance details…

normalized signed values



$$S \quad - \quad 4U \quad = \quad P - 3U$$

Unsharp mask:
…that seems oddly similar to a Laplacian

P

0	1	0
1	-4	1
0	1	0

L

normalized signed values

$\nabla^2 P = P * L$

Difference of Gaussians:

a scaled approximation of the Laplacian of Gaussian

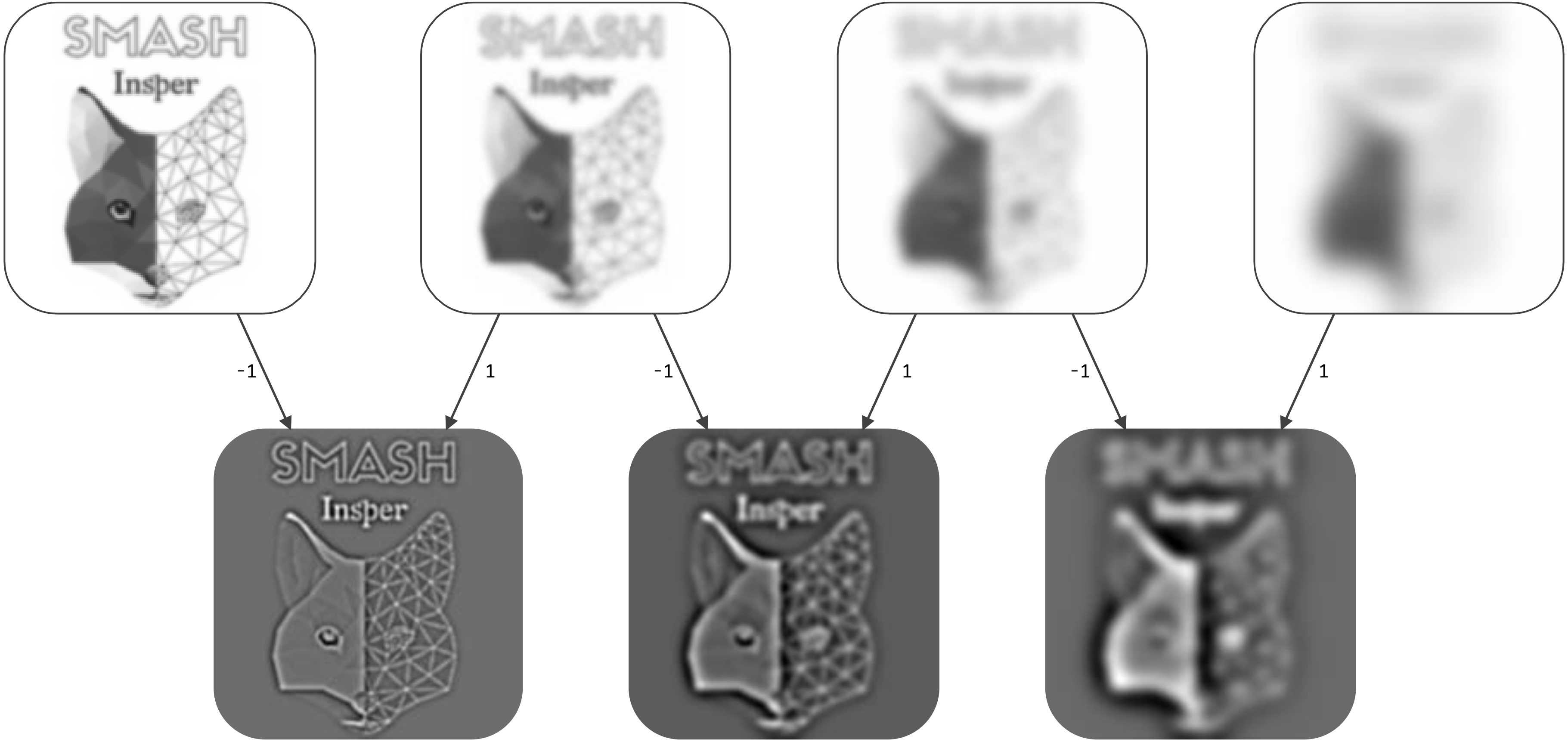
$$S_{k\sigma} - S_\sigma \approx (k-1) \cdot \sigma^2 \cdot \nabla^2 S_\sigma$$

S_σ

$S_{2\sigma}$

$S_{3\sigma}$

$S_{4\sigma}$


-1
1
-1
1
-1
1

$\nabla^2 S_\sigma$

$\nabla^2 S_{2\sigma}$

$\nabla^2 S_{3\sigma}$

The scale space gives Laplacians for free!

But, regardless of that, the main point is that robustness to scale is possible.

Convolution summary:

Convolution summary:

1. convolutions generalize many image processing and feature extraction algorithms;
    - gray level conversion is a convolution with a 1x1x3 kernel;
    - brightness adjustment is a convolution with a 1x1 kernel;
    - image smoothing is a convolution with a Gaussian kernel;
    - edge detection is a convolution with a Laplacian kernel;

Convolution summary:

1. convolutions generalize many image processing and feature extraction algorithms;
    - gray level conversion is a convolution with a 1x1x3 kernel;
    - brightness adjustment is a convolution with a 1x1 kernel;
    - image smoothing is a convolution with a Gaussian kernel;
    - edge detection is a convolution with a Laplacian kernel;

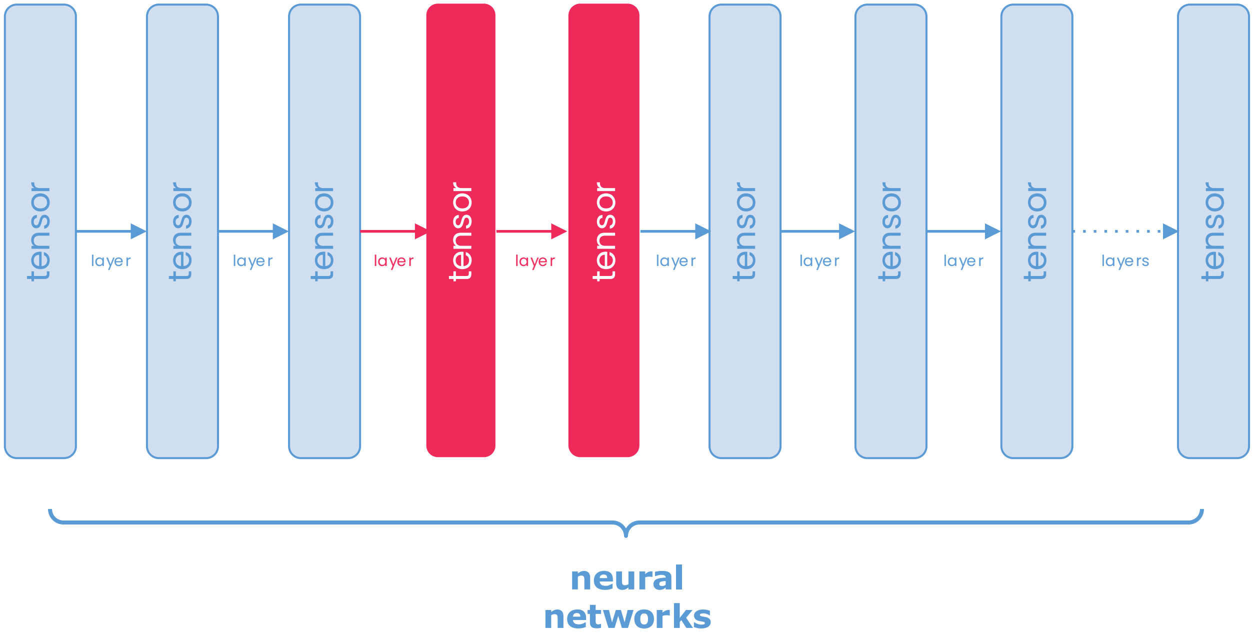
1. serial convolutions can be merged; *(Laplacian of Gaussian)*

Convolution summary:

1. convolutions generalize many image processing and feature extraction algorithms;
    - gray level conversion is a convolution with a 1x1x3 kernel;
    - brightness adjustment is a convolution with a 1x1 kernel;
    - image smoothing is a convolution with a Gaussian kernel;
    - edge detection is a convolution with a Laplacian kernel;

1. serial convolutions can be merged; *(Laplacian of Gaussian)*

1. but sometimes we need parallel convolutions. *(scale space)*

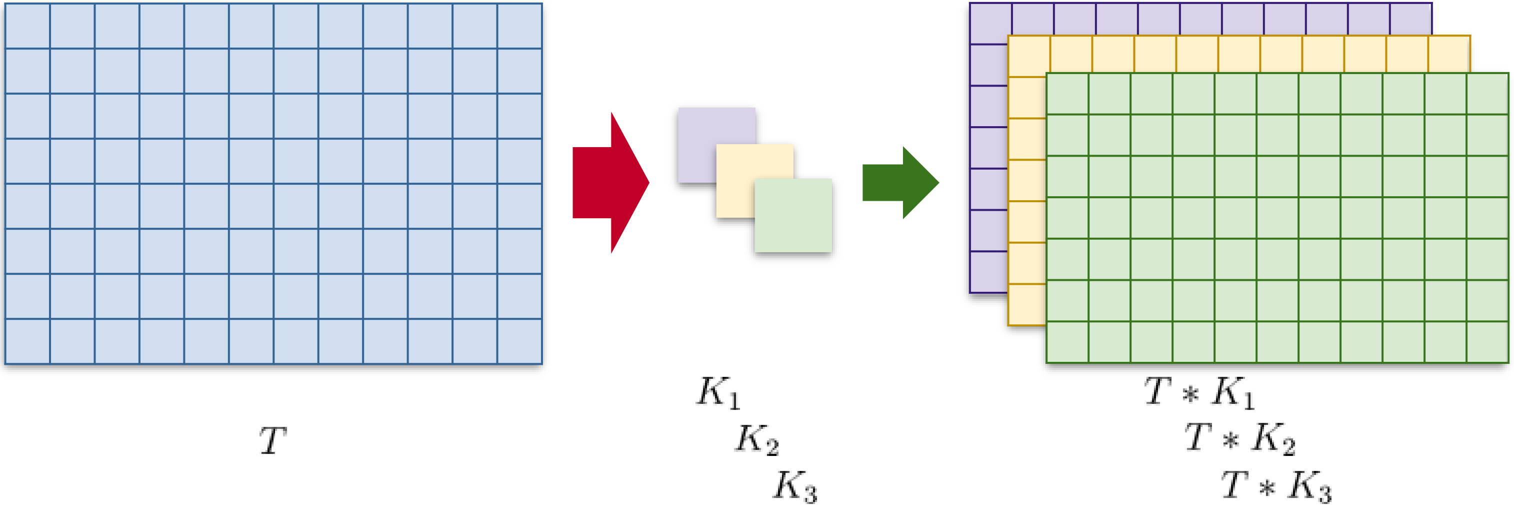

tensor
layer
tensor
layer
tensor
layer
tensor
layer
tensor
layer
tensor
layer
tensor
layer
tensor
layers
tensor
neural
networks

Neural network mysteries

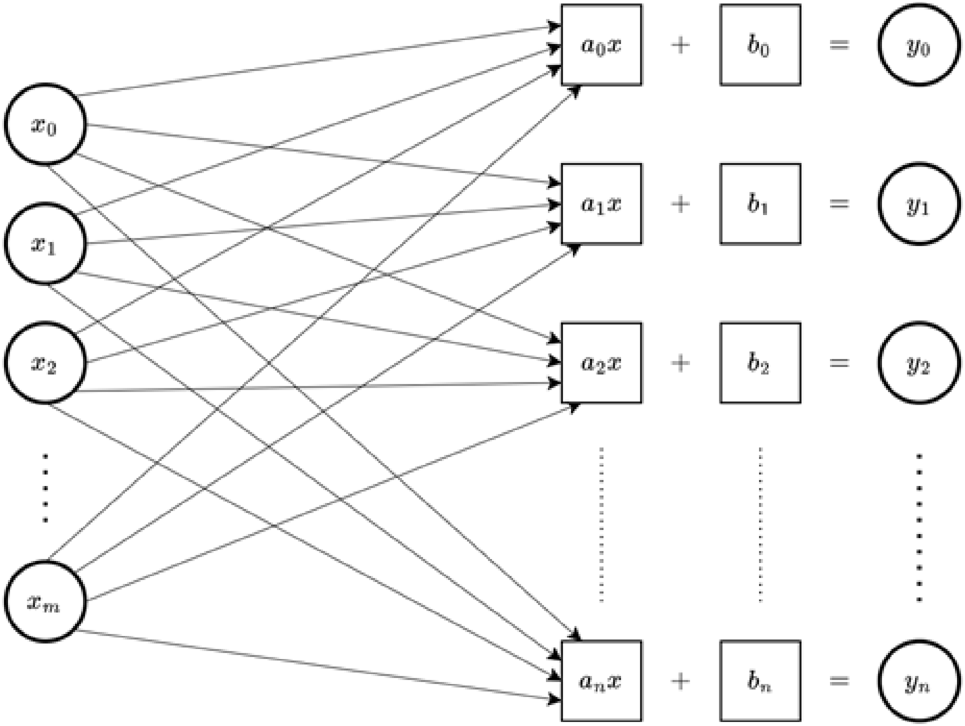
- How can we separate a dataset in training data and testing data?
- ~~What is a hidden layer?~~
- Does it matter if the input tensors are 3D, 2D, or 1D?
- ~~What is a dense layer?~~
- ~~How do we calculate the number of parameters in a dense layer?~~
- ~~How do we calculate the number of steps in a training process?~~
- ~~How do we calculate the number of steps in a testing process?~~
- Does the number of layers matter?
- ~~Does the size of a dense layer matter?~~
- What exactly `activation='relu'` does?
- ~~Can classic vision algorithms be represented by dense layers?~~
- ~~In a dense layer, are all the weights and biases always necessary?~~

Convolutional layer:

calculates the convolution of a tensor with one or more kernels



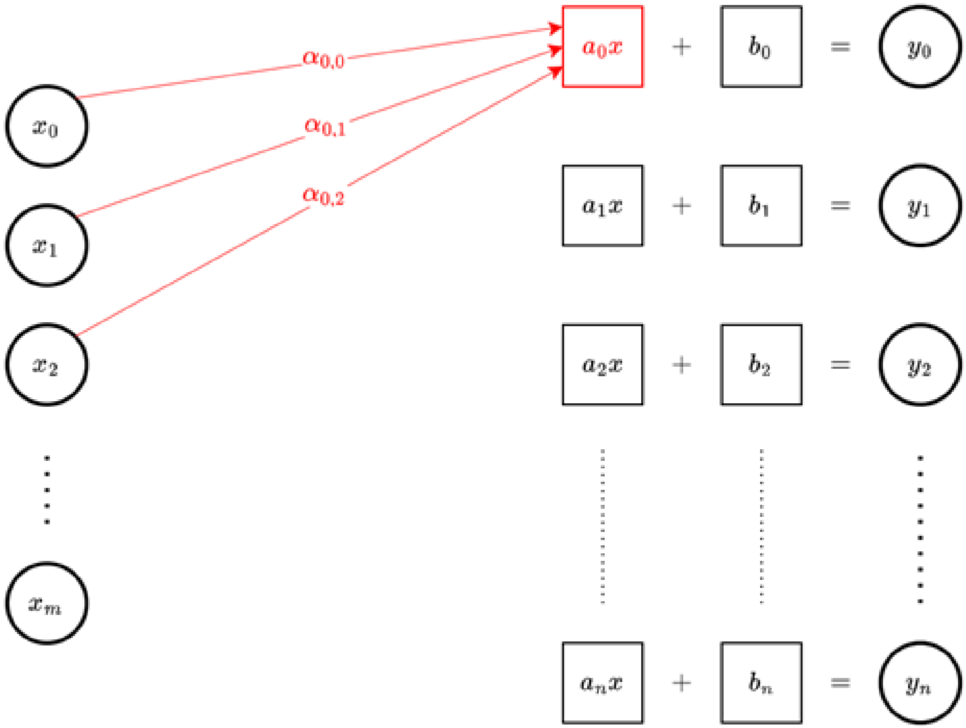
Convolutional layer:
can be seen as a dense layer...



Convolutional layer:

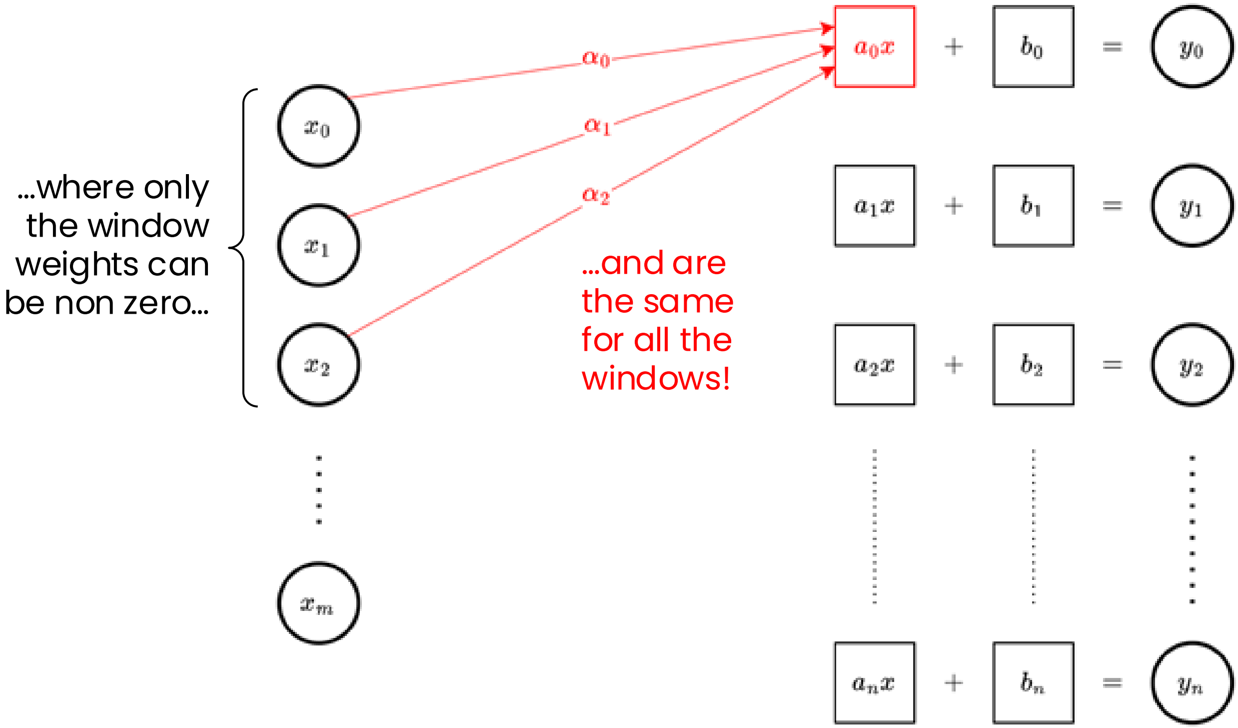
can be seen as a dense layer…

…where only the window weights can be non zero…



Convolutional layer:

can be seen as a dense layer…



Neural network mysteries

- How can we separate a dataset in training data and testing data?
- ~~What is a hidden layer?~~
- ~~Does it matter if the input tensors are 3D, 2D, or 1D?~~
- ~~What is a dense layer?~~
- ~~How do we calculate the number of parameters in a dense layer?~~
- ~~How do we calculate the number of steps in a training process?~~
- ~~How do we calculate the number of steps in a testing process?~~
- Does the number of layers matter?
- ~~Does the size of a dense layer matter?~~
- What exactly `activation='relu'` does?
- ~~Can classic vision algorithms be represented by dense layers?~~
- ~~In a dense layer, are all the weights and biases always necessary?~~
- How do we calculate the number of parameters in a convolutional layer?

Convolutional layer parameters

Convolutional layer parameters

- For a kernel of height h and width w...

Convolutional layer parameters

- For a kernel of height h and width w...
- ...with number of channels c...

Convolutional layer parameters

- For a kernel of height h and width w...
- ...with number of channels c...
- ...we have hwc weights...

Convolutional layer parameters

- For a kernel of height h and width w...
- ...with number of channels c...
- ...we have hwc weights...
- ...and c biases...

Convolutional layer parameters

- For a kernel of height h and width w...
- ...with number of channels c...
- ...we have hwc weights...
- ...and c biases...
- ...for a total of $hwc+c$ parameters.

Neural network mysteries

- How can we separate a dataset in training data and testing data?
- ~~What is a hidden layer?~~
- ~~Does it matter if the input tensors are 3D, 2D, or 1D?~~
- ~~What is a dense layer?~~
- ~~How do we calculate the number of parameters in a dense layer?~~
- ~~How do we calculate the number of steps in a training process?~~
- ~~How do we calculate the number of steps in a testing process?~~
- Does the number of layers matter?
- ~~Does the size of a dense layer matter?~~
- What exactly `activation='relu'` does?
- ~~Can classic vision algorithms be represented by dense layers?~~
- ~~In a dense layer, are all the weights and biases always necessary?~~
- ~~How do we calculate the number of parameters in a convolutional layer?~~

handout

Toolkit

- **Language:** Python
- **Library:** Keras
- **Platform:** Google Colab

Instructions

1. Organize in groups of 2 or 3 members. No more, no less.

1. Make a copy of the notebook, read it, and do the activities.

1. Clean the notebook, save as `ipynb`, and submit via form.

Next class:

- another type of features and an important classic framework.

Credits

This material was based on the work of other professors, listed below.

- Fabio Miranda (fabiomiranda@insper.edu.br)
- Raul Ikeda (RaulIGS@insper.edu.br)
- Fabio Ayres (FabioJA@insper.edu.br)
- Igor Montagner (IgorSM1@insper.edu.br)
- Andrew Kurauchi (AndrewTNK@insper.edu.br)
- Luciano Silva (LucianoS4@insper.edu.br)
- Tiago Sanches (tiagoss4@insper.edu.br)

Well, except for the errors. Any errors you might find are probably my fault.

Images

https://www.insper.edu.br/graduacao/organizacoes-estudantis/
https://www.insper.edu.br/campus/