(Advanced) deep learning

Convolutional neural networks - 2

2024/12/4

Jon Sporring,
Department of Computer Science

UNIVERSITY OF COPENHAGEN

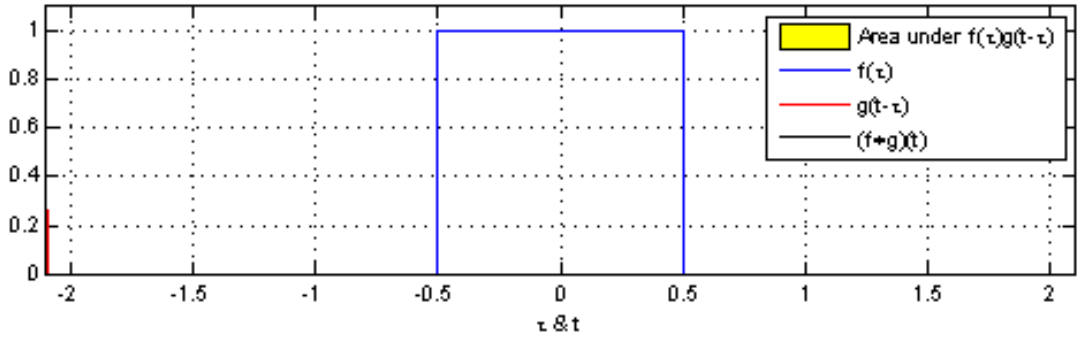
Convolution 1D

$$\mathrm{f, k}: \mathbb{R} \to \mathbb{R}$$

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

$$\mathrm{f, k} \in \mathbb{R}^M$$

$$\mathrm{g}(i) = (f * k)(i) = \sum_{i=1}^{M} f(m) k(i - m)$$



By Convolution_of_box_signal_with_itself.gif: Brian Ambergderivative work: Tinos (talk) - Convolution_of_box_signal_with_itself.gif, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=11003835

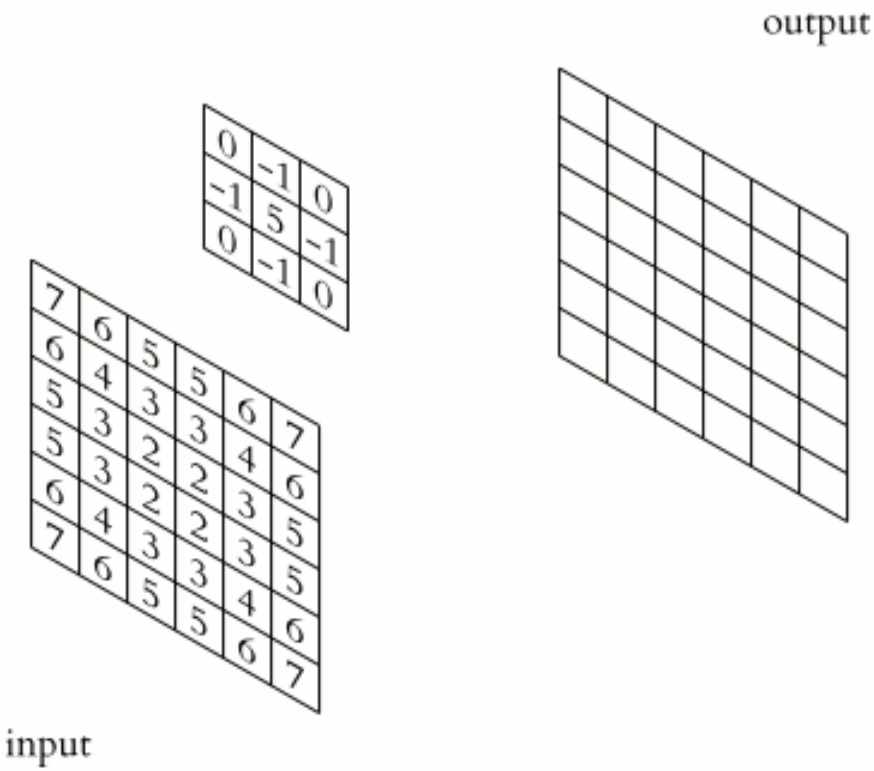
Convolution 2D

$$\mathrm{f}, \mathrm{k}\colon\ \mathbb{R}^2 \to \mathbb{R}^2$$

$$(\mathrm{f} * \mathrm{k})(x, y) = \iint_{-\infty}^{\infty} f(\alpha, \beta)\, k(x - \alpha, y - \beta)\, d\alpha\, d\beta$$

$$\mathrm{f}, \mathrm{k} \in \mathbb{R}^{M \times N}$$

$$(\mathrm{f} * \mathrm{k})(i, j) = \sum_{m=1}^{M} \sum_{n=1}^{N} f(m, n) k(i - m, j - n)$$



By Michael Plotke - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=24288958

Unet: Convolutional network – encoding and decoding

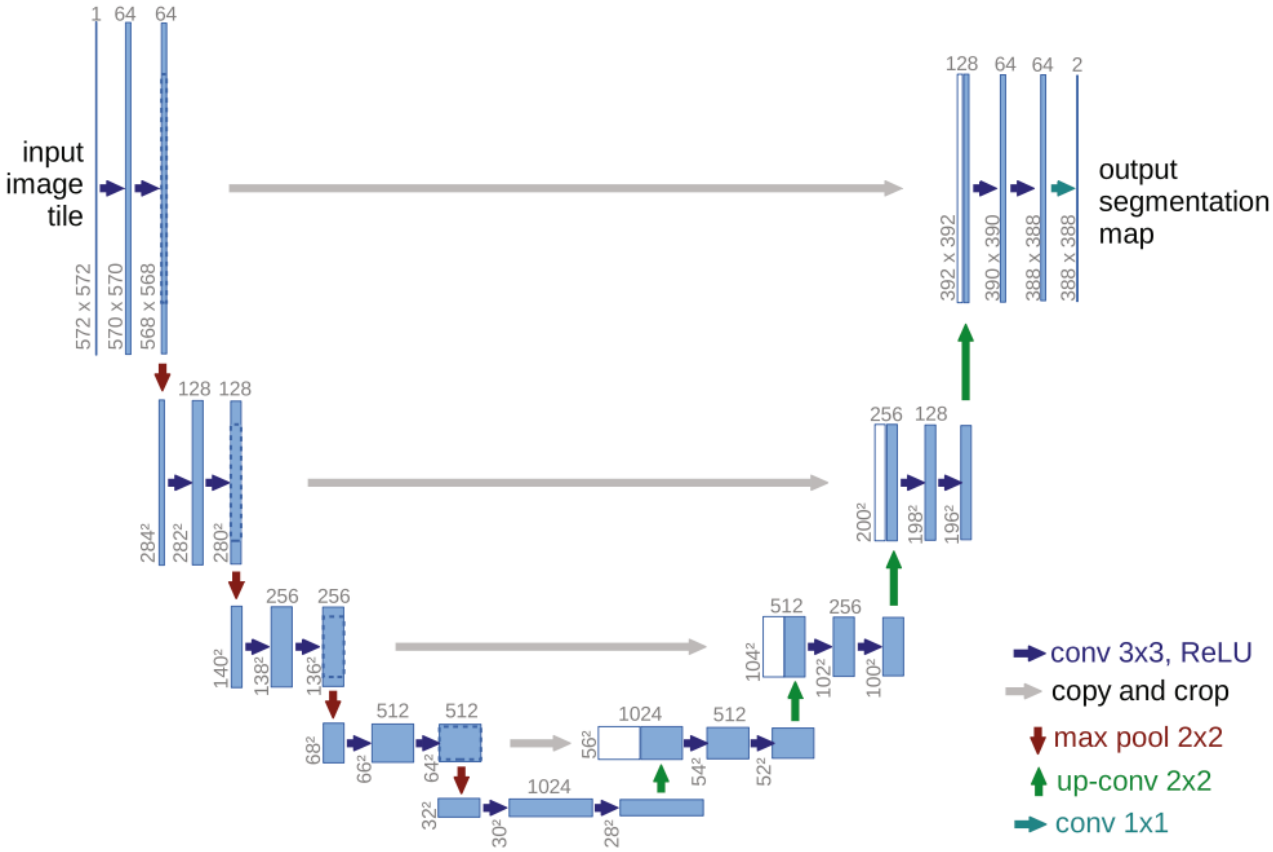
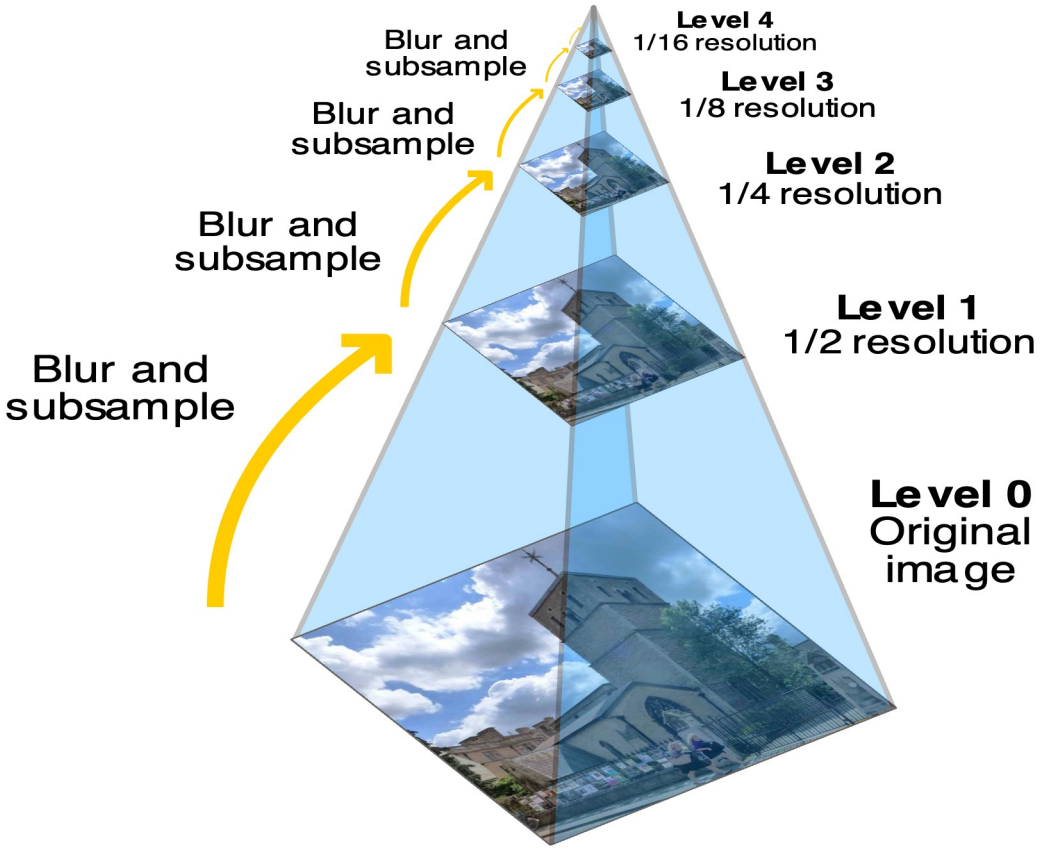
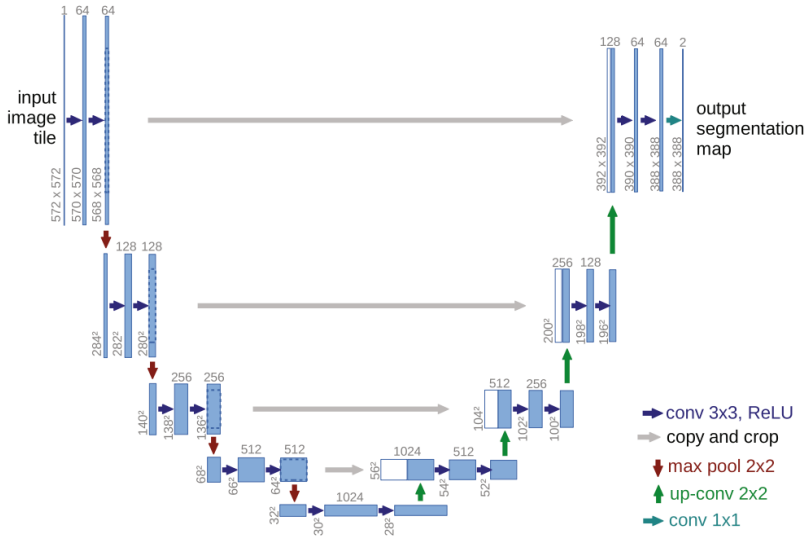


Image pyramid

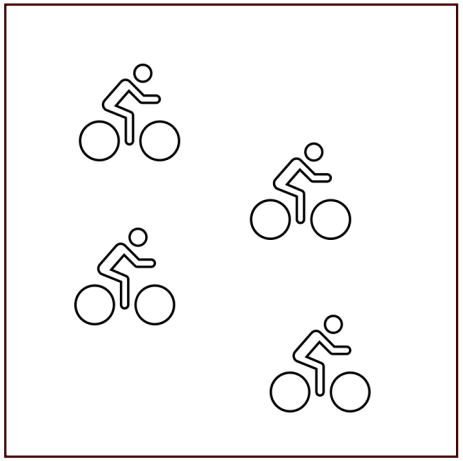


By Cmglee - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=42549151



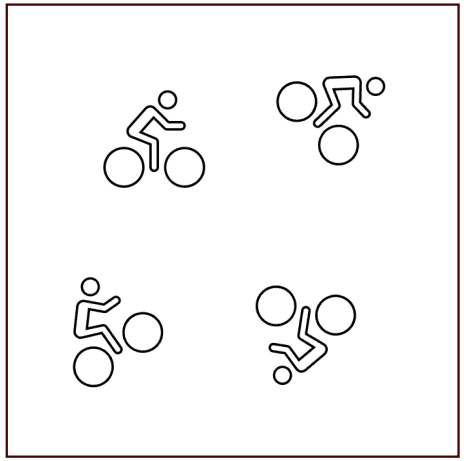
Invariances

Translation



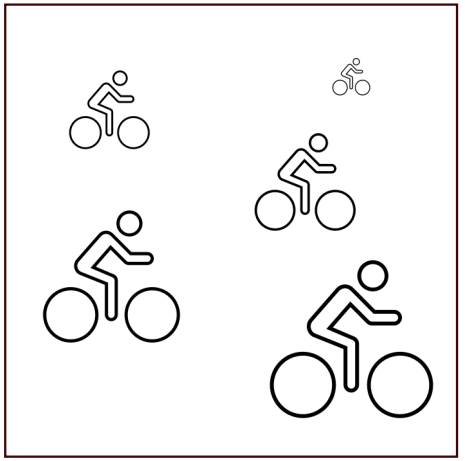
Convolution/Attention

Translation + Rotation



Augmentation/Geometric Deep Learning

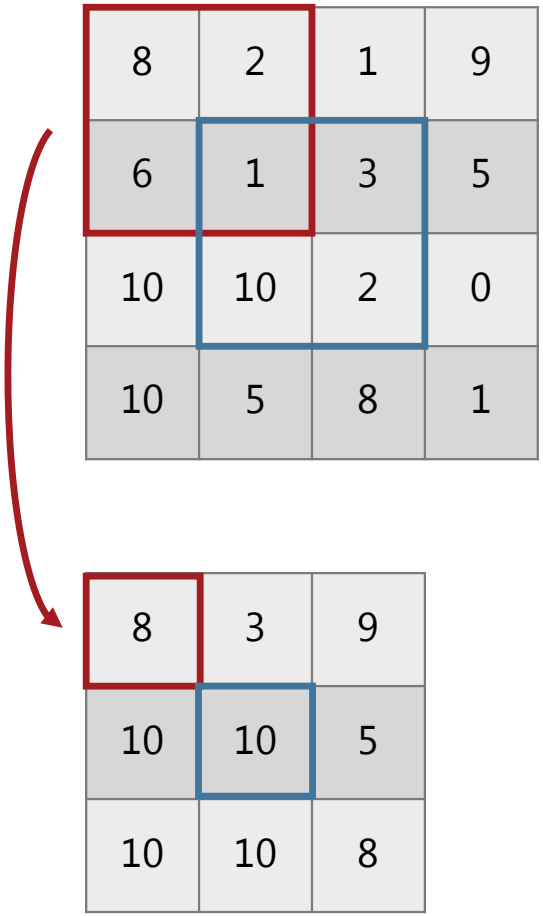
Translation + Scaling

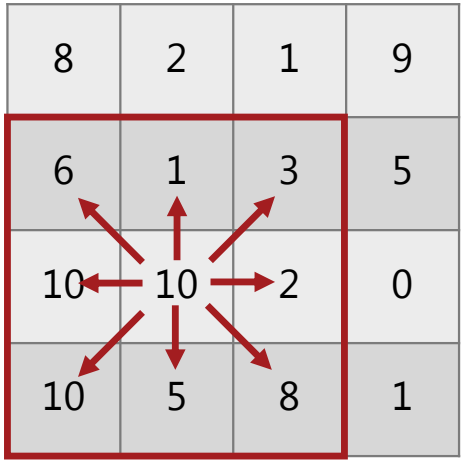


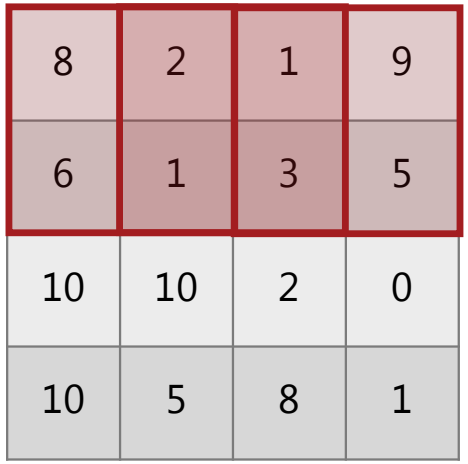
Augmentation/UNet

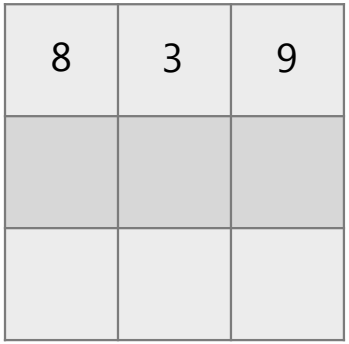
Max Pooling

$$f(i,j) = \max_{i=[0,M-1],j=[0,N-1]} f(i+m, j+n)$$

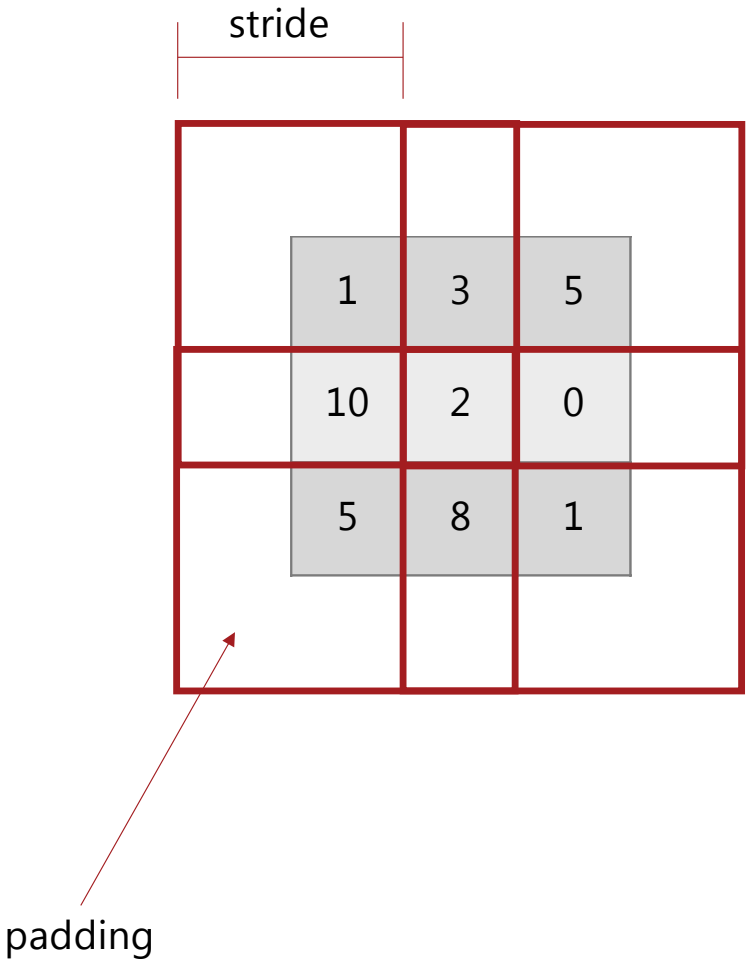








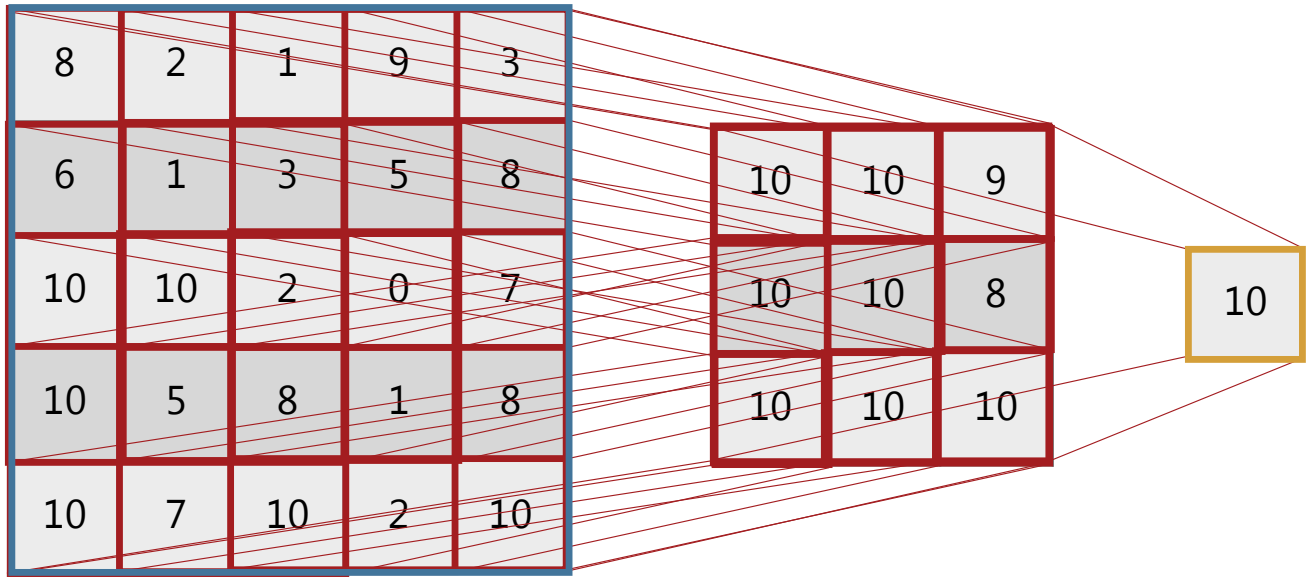
Padding & Stride

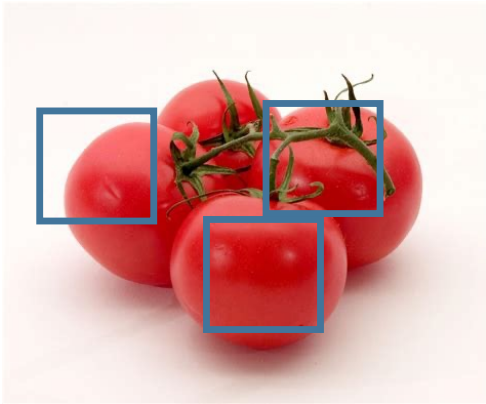


10	5
10	8

Receptive field

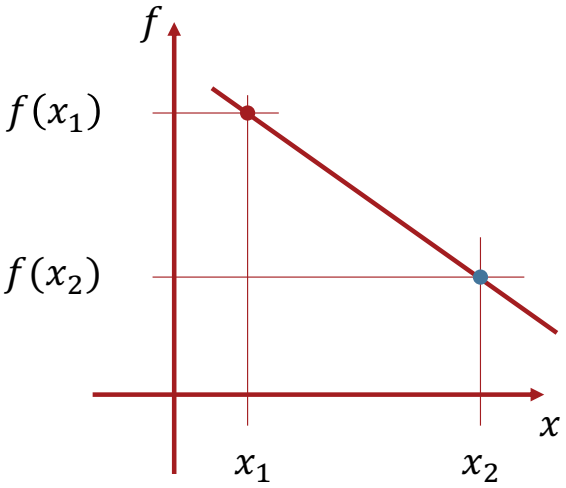
3x3 max pooling:

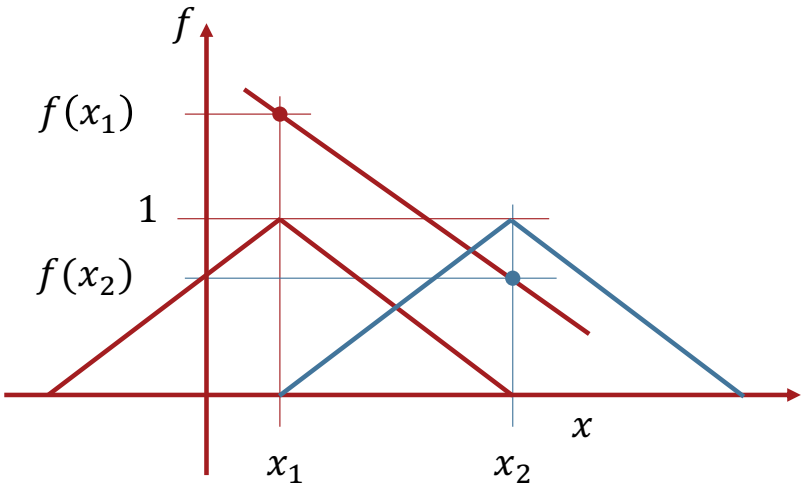




https://www.feedipedia.org/node/7791

Upsampling / upconvolution





10	7.5	5
10	8.25	6.5
10	9	8

$x \in [x_1, x_2]$

$$f(x) = \frac{x_2 - x}{x_2 - x_1} f(x_1) + \frac{x - x_1}{x_2 - x_1} f(x_2)$$

$$f(x) = a_1(x) f(x_1) + a_2(x) f(x_2)$$

Interpolation: $a_1(x) + a_2(x) = 1$

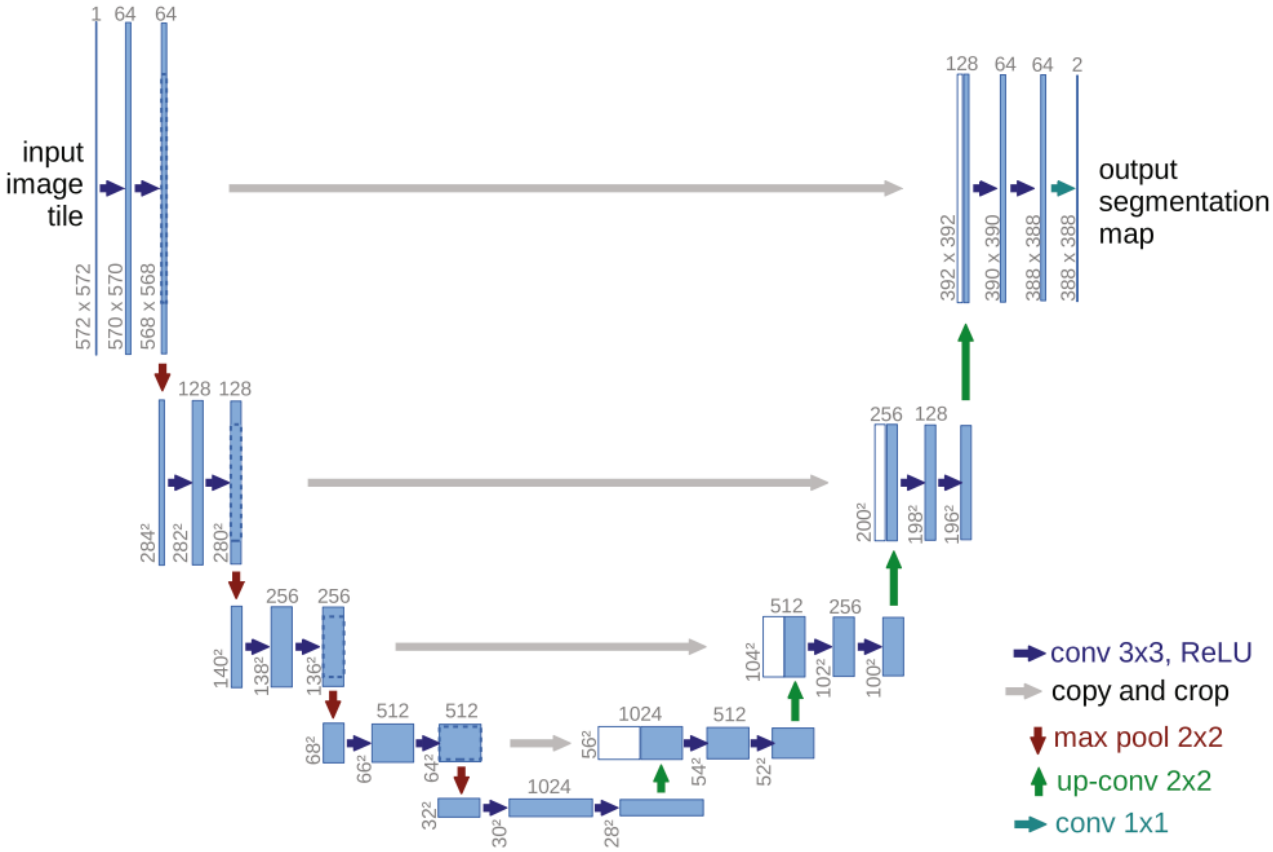
$$\tilde{f}(x) = (f * k)(x)$$
$$k(x) = [a_2, a_1](x)$$

$x \in [x_1, x_2], y \in [y_1, y_2]$

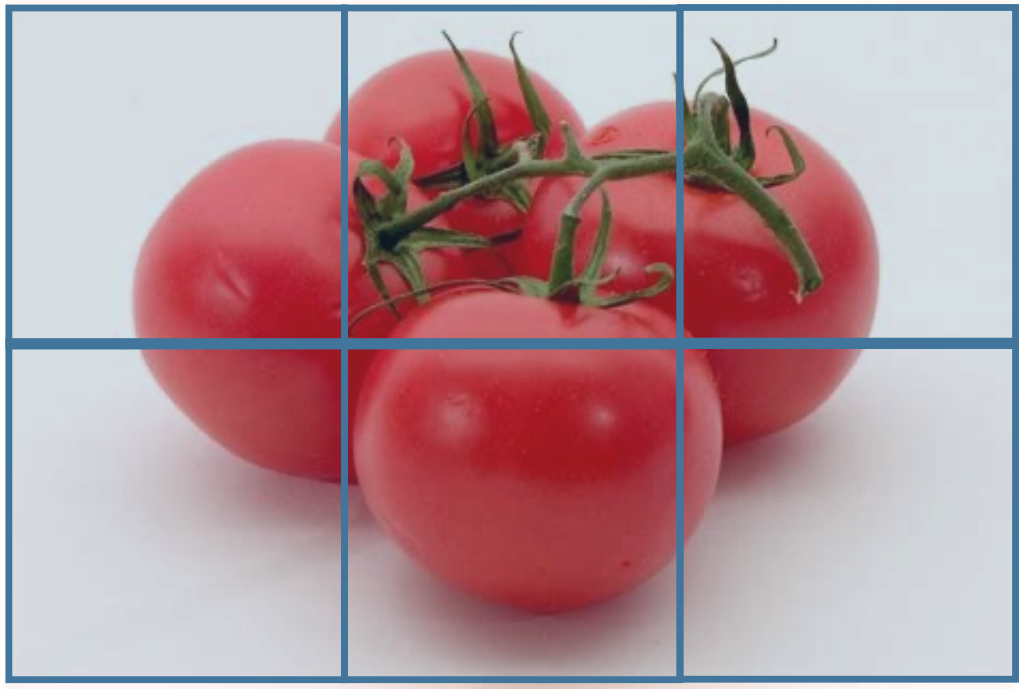
$$f(x, y) = \frac{y_2 - y}{y_2 - y_1} f(x, y_1) + \frac{y_1 - y}{y_2 - y_1} f(x, y_2)$$

$$\tilde{f}(x, y) = (f * k)(x, y)$$

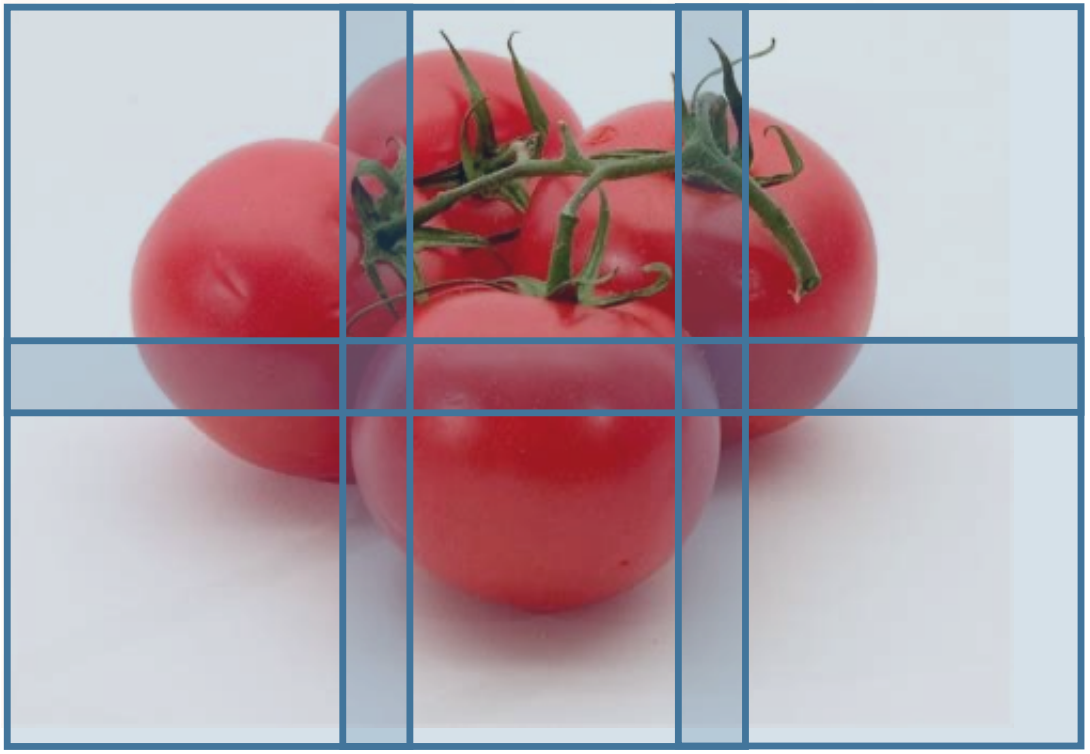
Unet: Convolutional network – encoding and decoding



Patch processing



https://www.feedipedia.org/node/7791



You never have enough training data

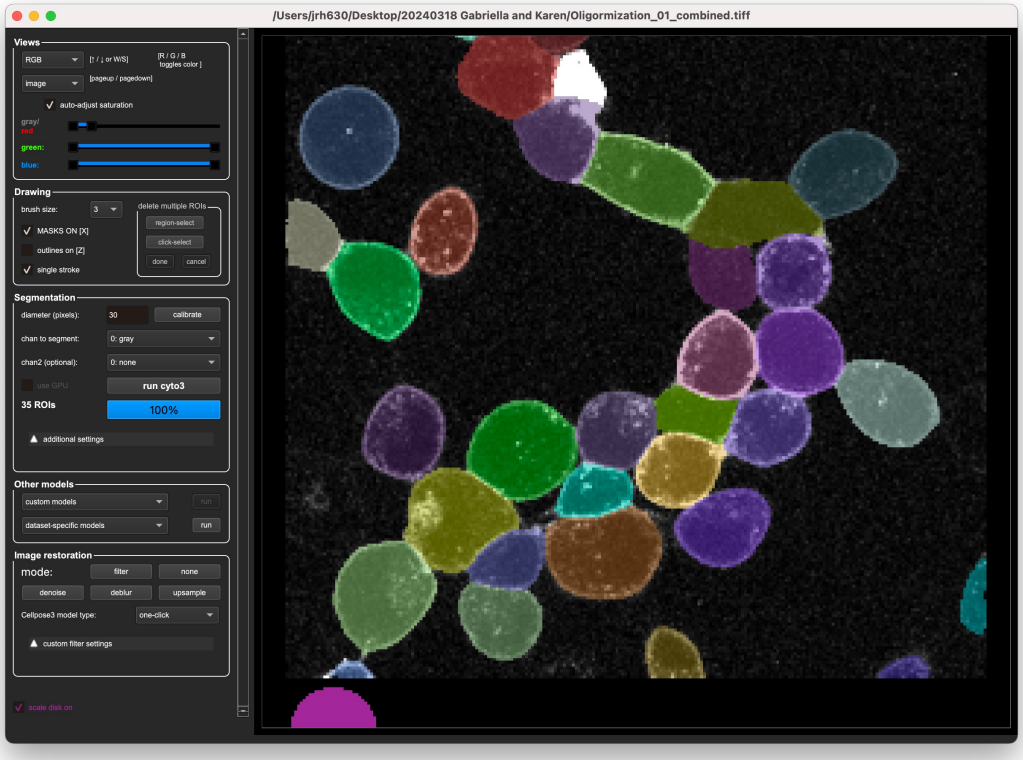


Image augmentation:

Color

Background

Translation

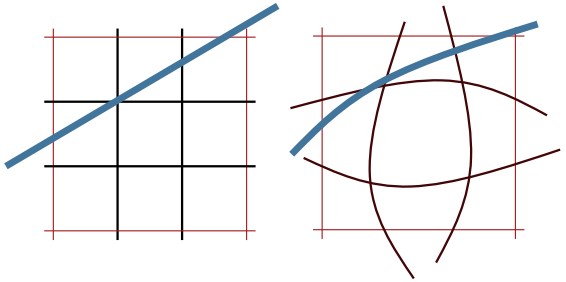
Rotation

Scaling

Flipping

Skewing

Non-linear deformations



Convergence issues: Normalization?

Ideas:

1. Normalize input – both at training and prediction!

$$x \to \frac{x - \mu}{\sigma} \in [0,1]$$

2. Classification with one-hot encoding

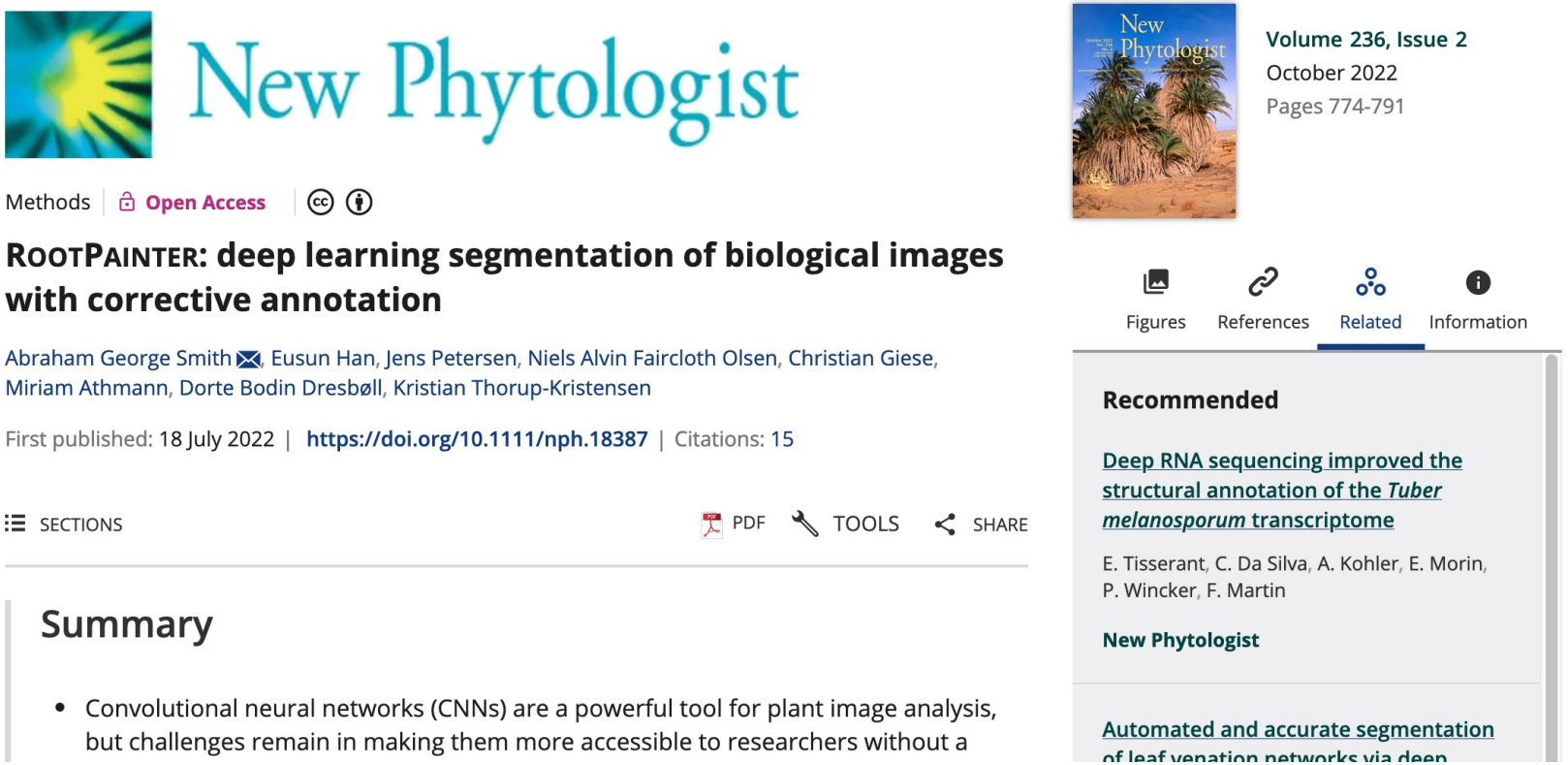
$$y \in \{0,1\}^N$$

3. Normalize inner layers – both training and prediction:

$$x \to \alpha \odot \frac{x - \mu_B}{\sigma_B} + \beta$$

Trainable parameters

Corrective segmentation: Root painter
https://github.com/Abe404/root_painter



New Phytologist

Methods | Open Access

ROOTPAINTER: deep learning segmentation of biological images with corrective annotation

Abraham George Smith, Eusun Han, Jens Petersen, Niels Alvin Faircloth Olsen, Christian Giese, Miriam Athmann, Dorte Bodin Dresbøll, Kristian Thorup-Kristensen

First published: 18 July 2022 | https://doi.org/10.1111/nph.18387 | Citations: 15

SECTIONS PDF TOOLS SHARE

Summary

- Convolutional neural networks (CNNs) are a powerful tool for plant image analysis, but challenges remain in making them more accessible to researchers without a

Volume 236, Issue 2
October 2022
Pages 774-791

Figures References Related Information

Recommended

Deep RNA sequencing improved the structural annotation of the *Tuber melanosporum* transcriptome

E. Tisserant, C. Da Silva, A. Kohler, E. Morin, P. Wincker, F. Martin

New Phytologist

Automated and accurate segmentation of leaf venation networks via deep

Where to be inspired: https://paperswithcode.com/

Search

Browse State-of-the-Art Datasets Methods More

Top New Greatest

Semantic Segmentation

5207 papers with code • 125 benchmarks • 311 datasets

to categorize each pixel in an image into a map of an image, where each pixel is are Cityscapes, PASCAL VOC and (Mean IoU) and Pixel Accuracy

Add a Result

Content

- Introduction
- Benchmarks
- Datasets
- Subtasks
- Libraries
- Papers
  - Most implemented
  - Social
  - Latest
  - No code

Image Classification

3788 papers with code • 142 benchmarks • 238 datasets

Image Classification is a fundamental task in vision recognition that aims to understand and categorize an image as a whole under a specific label. Unlike object detection, which involves classification and location of multiple objects within an image, image classification typically pertains to single-object images. When the classification becomes highly detailed or reaches instance-level, it is often referred to as image retrieval, which also involves finding similar images in a large database.

Source: Metamorphic Testing for Object Detection Systems

Benchmarks

These leaderboards are used to track prog

Trend	Dataset	Best Model
	ImageNet	OmniVec(V
	CIFAR-10	ViT-H/14
	CIFAR-100	EffNet-L2 (

Object Detection

3714 papers with code • 91 benchmarks • 262 datasets

Object Detection is a computer vision task in which the goal is to detect and locate objects of interest in an image or video. The task involves identifying the position and boundaries of objects in an image, and classifying the objects into different categories. It forms a crucial part of vision recognition, alongside image classification and retrieval.

The state-of-the-art methods can be categorized into two main types: one-stage methods and two stage-methods:

- One-stage methods prioritize inference speed, and example models include YOLO, SSD and RetinaNet.
- Two-stage methods prioritize detection accuracy, and example models include Faster R-CNN, Mask R-CNN and Cascade R-CNN.