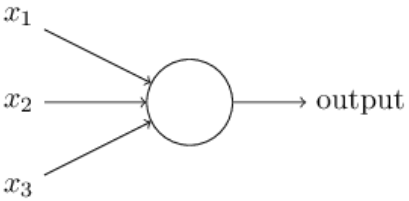
Deep Neural Networks for Cell Segmentation and Classification

Len Brandes
11.03.2021

HelmholtzZentrum münchen
German Research Center for Environmental Health

One Neuron



- Input vector

$$\boldsymbol{x} = \{x_1, x_2, x_3\} =$$

- Output

  $P(cat)$ = probability that an image contains a cat

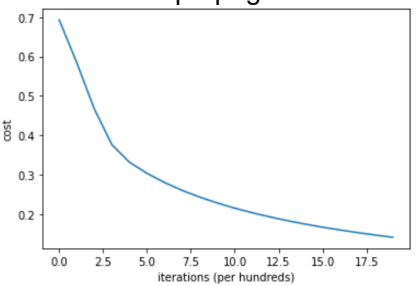
- Output depends on free, adjustable parameters

Training

- Supervised learning: use labelled training data

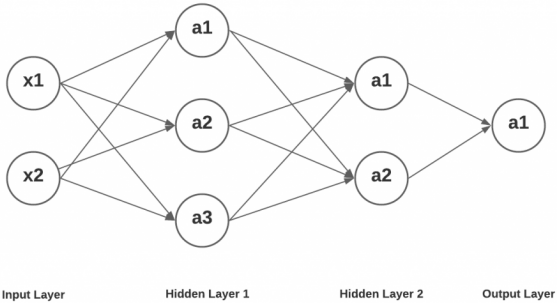
$$\left(\text{  }, P(cat) = 1 \right), \left(\text{  }, P(cat) = 0 \right), \ldots$$

- Neuron parameters trained via backpropagation



- Afterwards can use neuron on new data → Machine Learning
- But: limited to simple representations

Deep Neural Networks

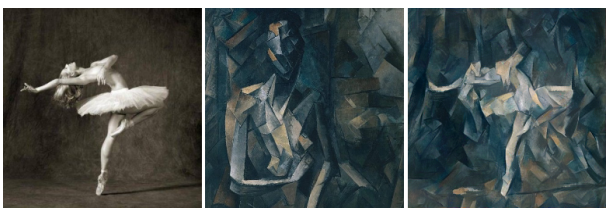


$P(cat)$

- Input (cat/non-cat pictures) and output (cat probability) as before
- Use multiple neurons in consecutive layers
- More parameters to train
  - Allows more complex representations
  - Needs more training data
  - Numerically more costly

Deep Neural Networks

- Computers used to be good at logical tasks, but bad at pattern recognition
- Huge advances in computational power and labelled datasets
- Reached and even surpassed human level performance for multiple tasks:
  - Speech recognition (Siri, Alexa)
  - Computer vision (tumor detection in medical images)
  - Natural language processing (DeepL, Google Translate)



- Much hyped and very fast research area

Convolutional Neural Networks

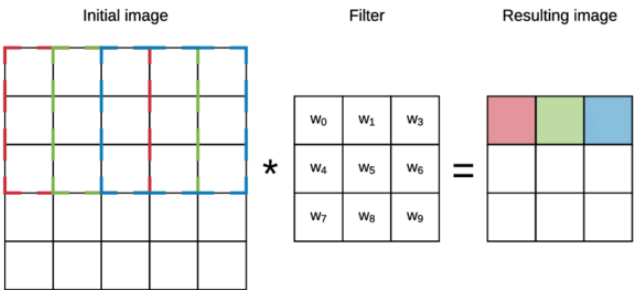
- Problem for image analysis: huge number of inputs

  Example: 512×512 RGB image $\rightarrow \sim 800{,}000$ Pixels

- Want to detect features (eye, nose, wheel, ...)

  $\rightarrow$ Spatially invariant

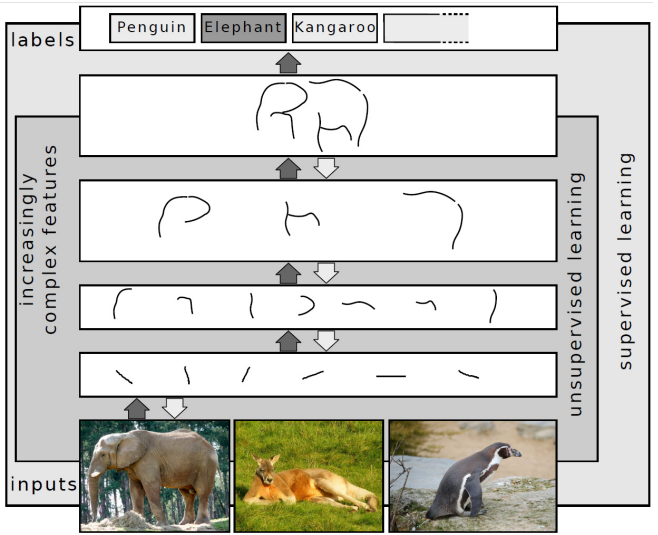
- Convolve image with a filter



- Reduced number of parameters

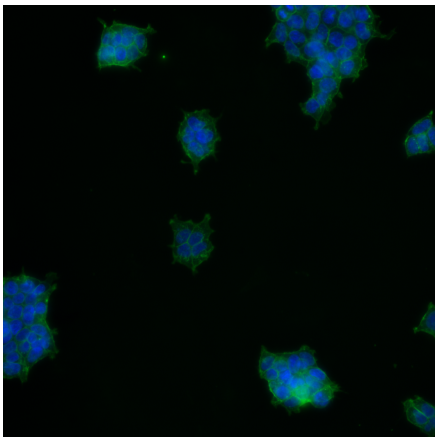
Convolutional Neural Networks

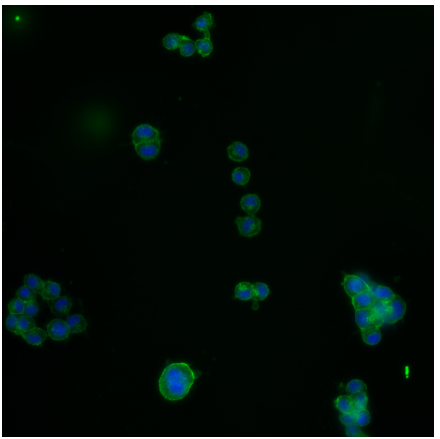
- Earlier layers detect simpler features, later layers more complex ones



Input Data

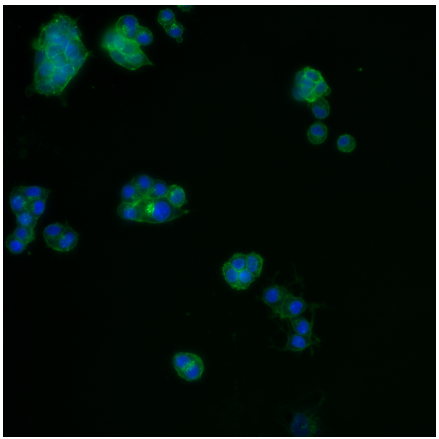
- Two celltypes, HEK and N2a
- Cytoplasm channel (green), nucleus channel (blue)
- Images with just one celltype and mixed images





Input Data

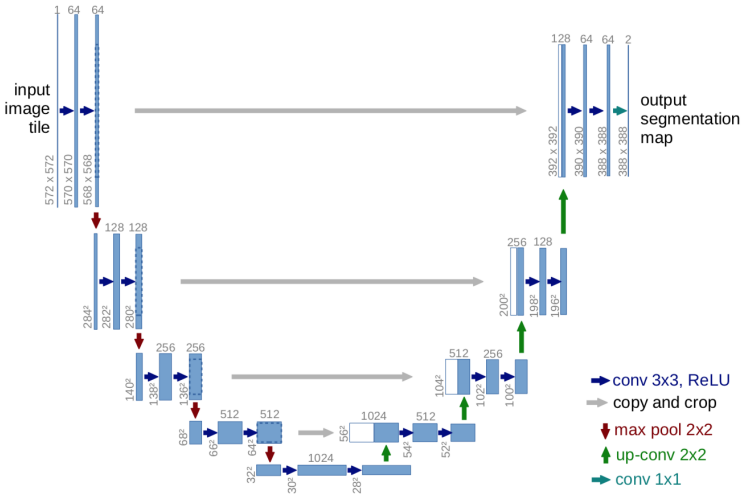
- Tasks: segment and classify cells
- Train existing Convolutional Neural Network models



Segmentation - Model

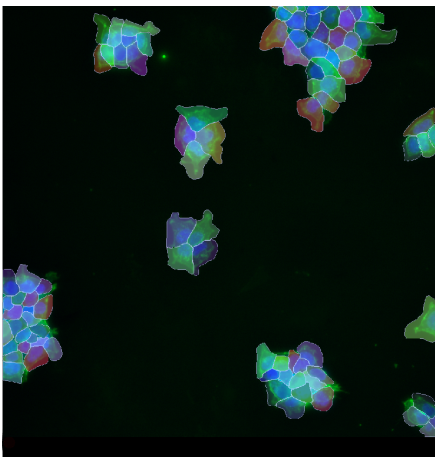
- Use U-Net type model *cellpose* for segmentation

  [C. Stringer, T. Wang, M. Michaelos & M. Pachitariu, Nature Methods **18**, 100–106 (2021)]



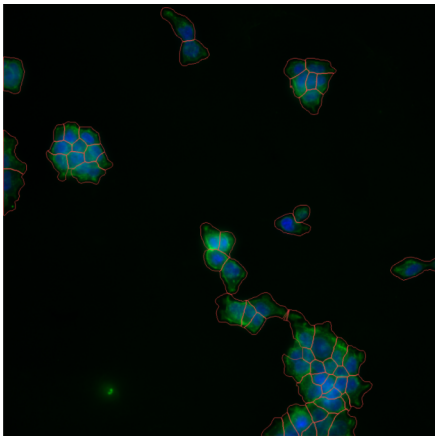
Segmentation - Training

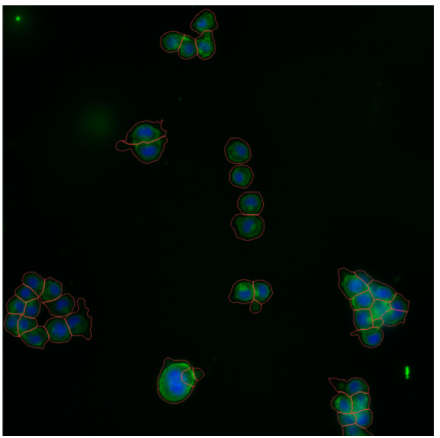
- Pre-trained on huge set of celldata (70,000 segmented objects)
- Use 60 hand-segmented images for training on own celltypes
- Cloud computing for necessary computational power (Kaggle, Google colab)



Segmentation - Results

- Already good results, could be improved with more training data and computation time
- Takes only $\sim$ 5s per image
- No classification of celltypes possible

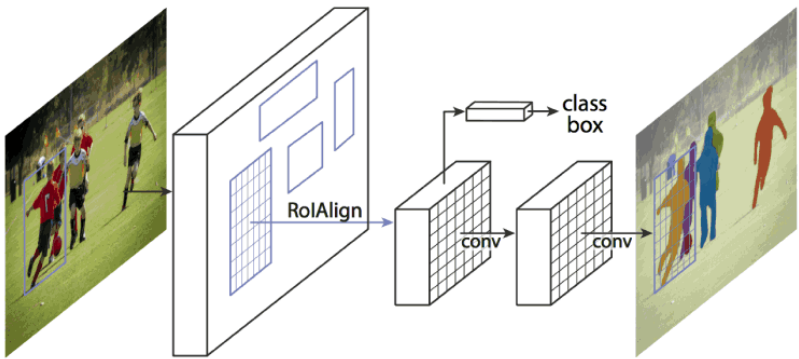




Segmentation & Classification - Model

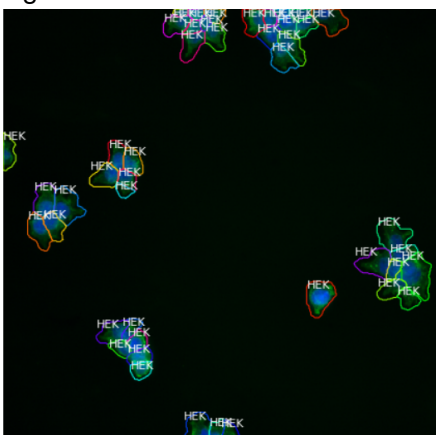
- Use *Mask R-CNN* model for object detection
  [K. He, G. Gkioxari, P. Dollar & R. Girshick (FAIR), International Conference on Computer Vision (ICCV), 2017]

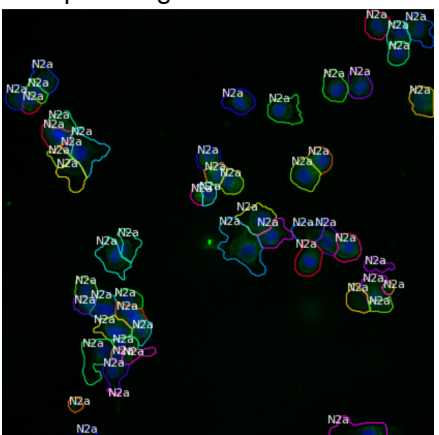
- Based on segmentation from cellpose



Segmentation & Classification - Results

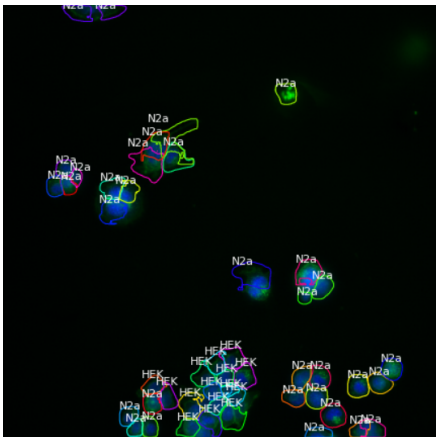
- Pre-trained on standard COCO dataset (330K images, 80 object categories)
  [T.-Y. Lin et al., European Conference on Computer Vision (ECCV), 2014]
  → Expect basic features to be the same
- Use image-name to label celltypes for training
- Segmentation & classification takes only ~ 2s per image

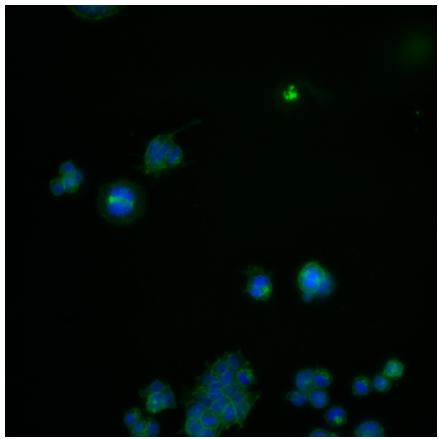




Segmentation & Classification - Results

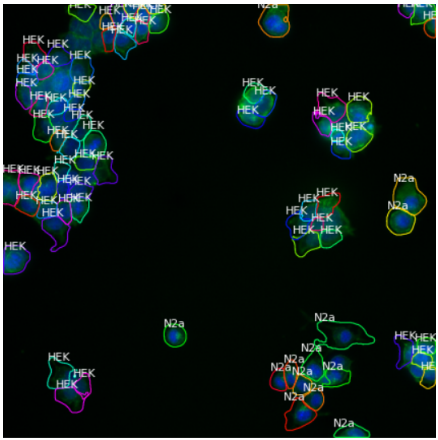
- Segmentation misses a few cells
- Expect ~50% of each celltype on mixed images, got 49.6% / 50.4%

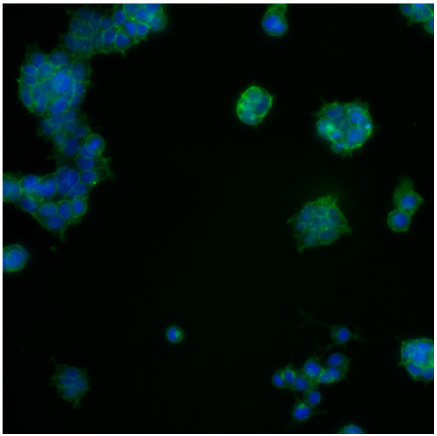




Segmentation & Classification - Results

- Further optimization possible:
  - Mixed training images
  - Hyperparameter adjustment
  - More computation power





Summary

- Deep Neural Networks have become very successful at pattern recognition
- Can be applied to cell segmentation and classification, here:
  - Pre-trained cellpose model for cell segmentation
  - Based on this segmentation, Mask R-CNN model for additional cell classification
- For that necessary:
  - Labelled training data
  - Large computational power via cloud computing