# Deep Learning Frameworks and Applications (8DC00)
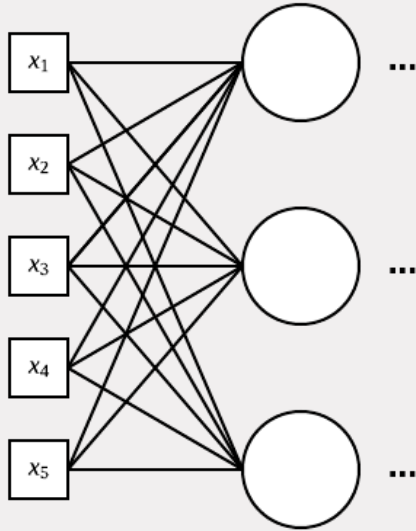**Friso G. Heslinga**

# Previous lecture
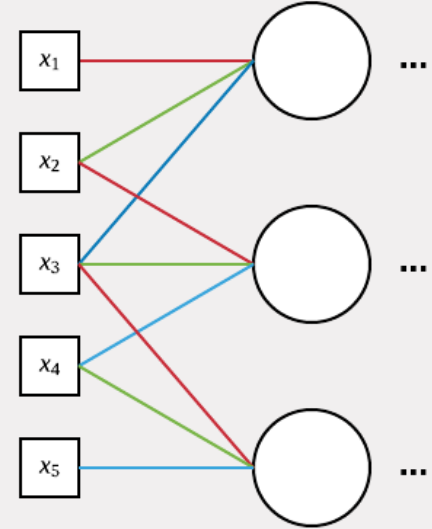
**receptive field**
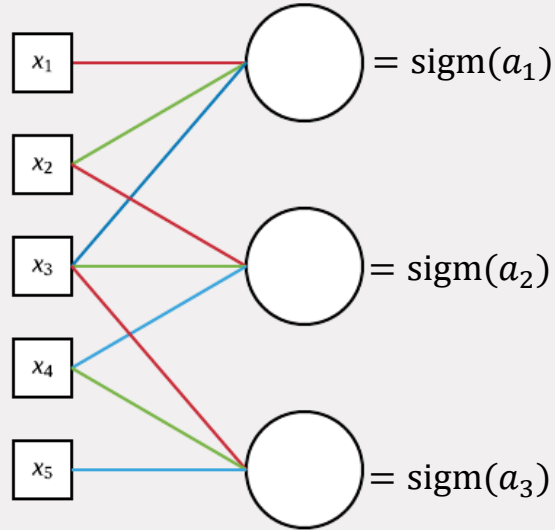
# Reducing # of weights

**"regular" NN**
15 weights
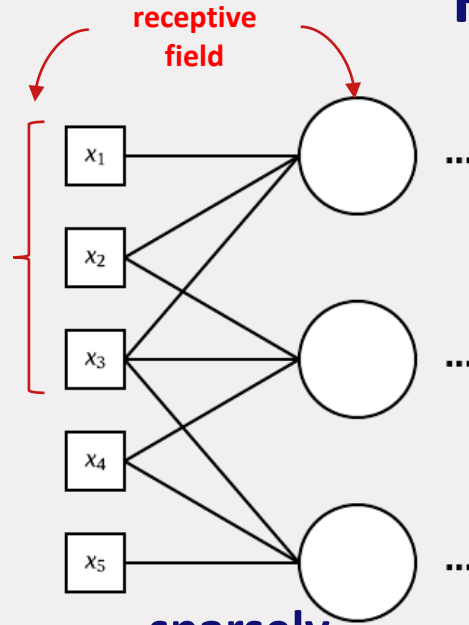
**sparsely connected NN**
9 weights

**shared weights**
3 weights

TU/e

# Previous lecture



$$a_1 = x_1 w_1 + x_2 w_2 + x_3 w_3$$

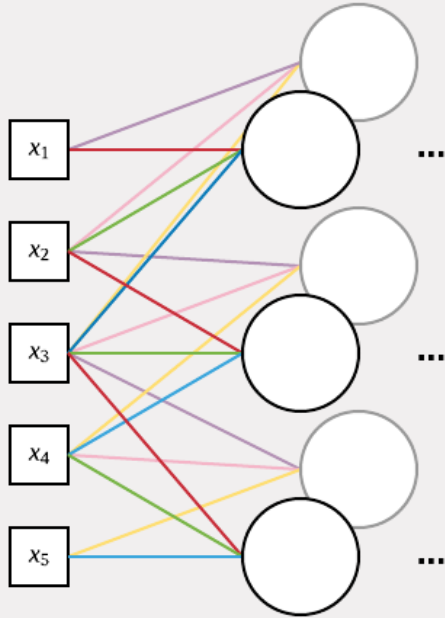$$a_2 = x_2 w_1 + x_3 w_2 + x_4 w_3$$

$$a_3 = x_3 w_1 + x_4 w_2 + x_5 w_3$$

$$[a_1 \quad a_2 \quad a_3] =$$
$$[x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5] * [w_1 \quad w_2 \quad w_3]$$

**convolution**

**TU/e**

# Previous lecture



$$[a_{1,1} \quad a_{1,2} \quad a_{1,3}] =$$
$$[x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5] * [w_{1,1} \quad w_{1,2} \quad w_{1,3}]$$

$$[a_{2,1} \quad a_{2,2} \quad a_{2,3}] =$$
$$[x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5] * [w_{2,1} \quad w_{2,2} \quad w_{2,3}]$$

$[w_{1,1} \quad w_{1,2} \quad w_{1,3}]$, and $[w_{2,1} \quad w_{2,2} \quad w_{2,3}]$ are **convolution kernels**. They extract features.

**two sets** shared weights
6 weights

TU/e

# Previous lecture: properties of CNNs

- Sparse connectivity
- Weight sharing
- Parallel computations

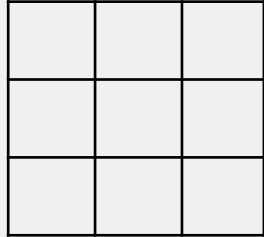TU/e
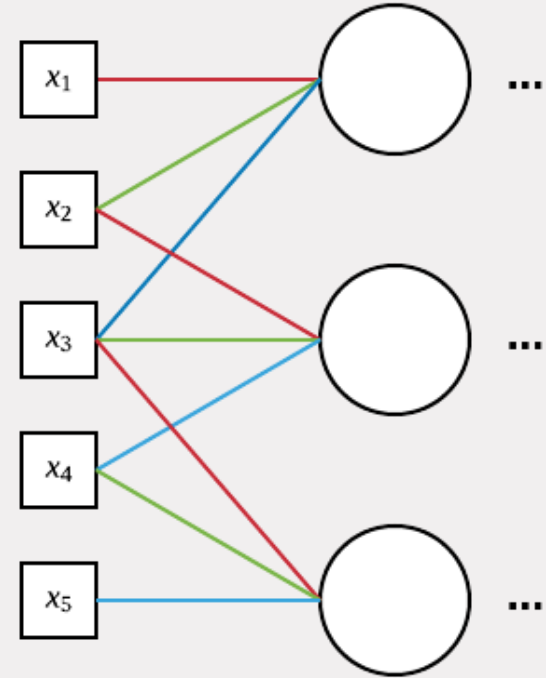
# Previous lecture: Kernel size
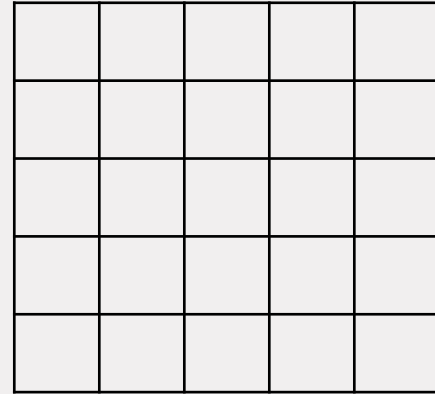
1 x 1                    3 x 3                         5 x 5

- More weights → more information
- More computations / memory
- Receptive field

TU/e

# Previous lecture: Max-pooling

- Reduce size of feature space

- Maximum of features

- Typical kernel size = 2 x 2

*Addition to last lecture:*

Stride = step size of the sliding window

- Typically 1 for convolutions

- Typically equal to kernel size for pooling operations

TU/e

aw your number here

0123456789

Downsampled drawing: 3

First guess: 3

Second guess: 1

ver visibility

t layer  Show

volution layer 1  Show

wnsampling layer 1  Show

volution layer 2  Show

# Previous lecture: Training of CNNs

- Similar to training of 'fully connected' neural networks

- **Choose some (random) initial values for network weights**

- Optimize networks weights with respect to a loss function that describes difference between network output and label/annotation

- Update networks weights iteratively

- Keep track of model performance on train & validation set

TU/e

# Previous lecture

- Convolutional neural networks (CNN)
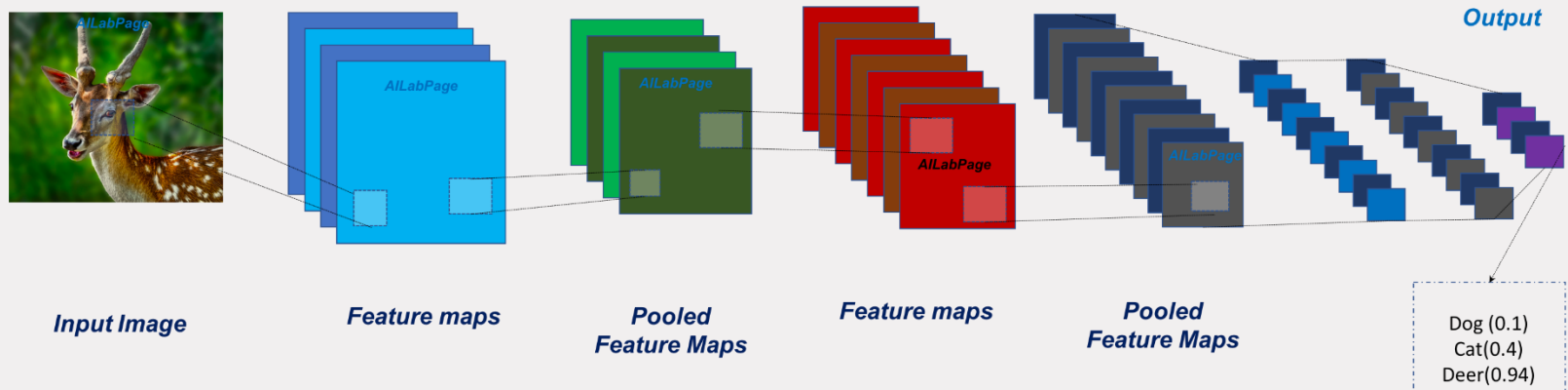


- By now: students can design a (simple) CNN

Image from: https://vinodsblog.com/

TU/e

# Challenges..

Disadvantage of designing yourself..?

➢ New implementation

➢ Design choices

➢ Time-consuming

➢ Debugging

➢ No reference performance

➢ Non-optimized model

➢ Train from scratch

TU/e

# Deep Learning Framework

Model architecture (from literature)

+
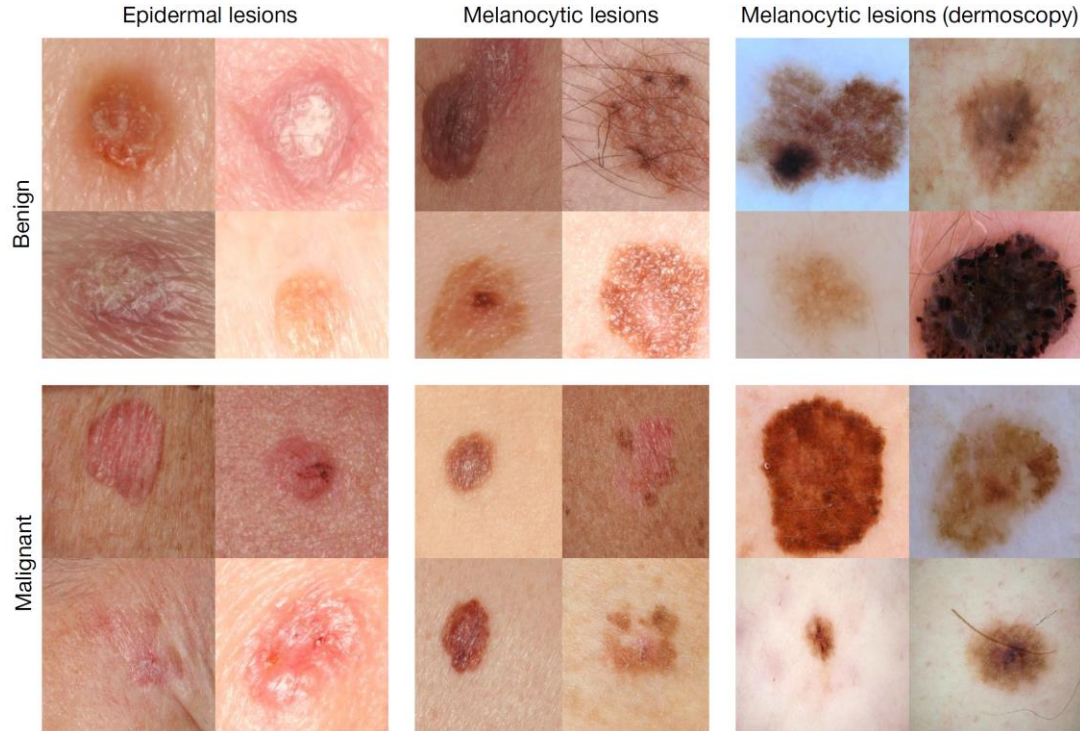
Framework (software packages)

TU/e

# Learning outcomes

- Students will be able to name a few **deep learning models** and describe the **differences in model design**

- Students can motivate why they **would choose a deep learning model** for a specific research application

- Students can name **two key developments** that have enabled deep learning

TU/e

# Lecture outline

- Different applications → Different requirements

- Small exercise: "what should be the output?"

- History of deep learning models

- Examples of deep learning models & applications

- *Break (15 mins)*

- Examples of deep learning software

- Group exercise: *"choosing the right deep learning approach"*

TU/e

# Classification



Epidermal lesions | Melanocytic lesions | Melanocytic lesions (dermoscopy)

Benign

Malignant

Esteva et al., Nature 2017
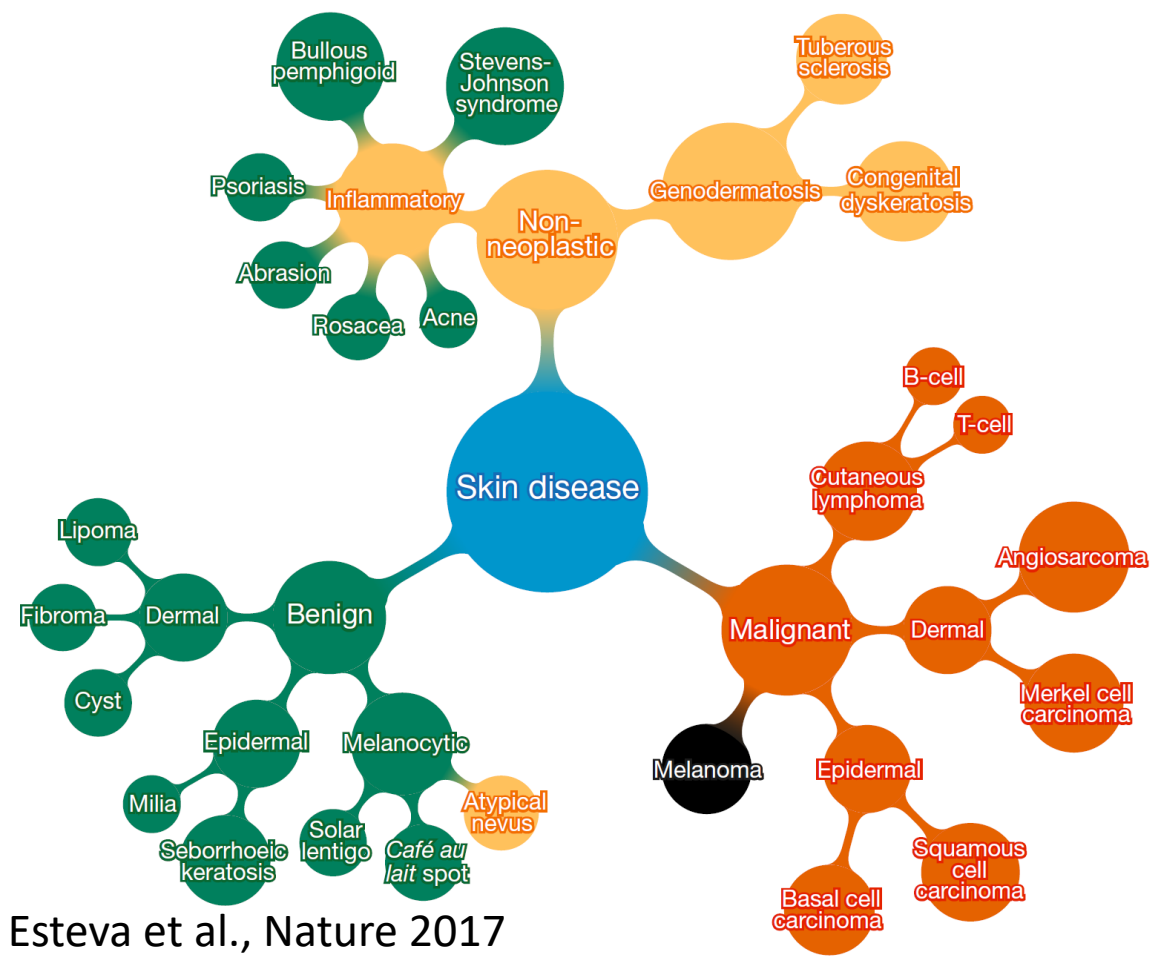
TU/e

# Classification

**Two classes**

e.g. malignant lesions / other

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

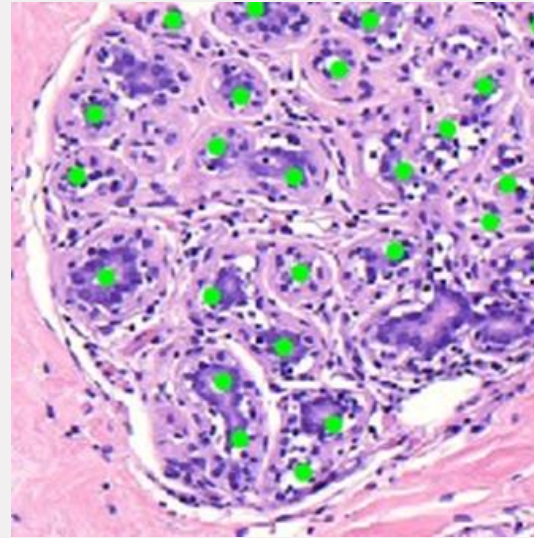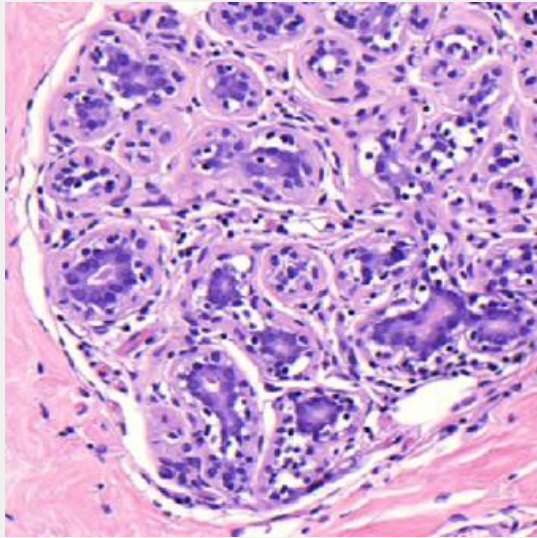**Multi-class**

benign lesions / malignant lesions / non-neoplastic

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$$



Esteva et al., Nature 2017

TU/e

# Detection of Acini centers



Wetstein et al., SPIE-MI 2019

# Regression

Retinal structures
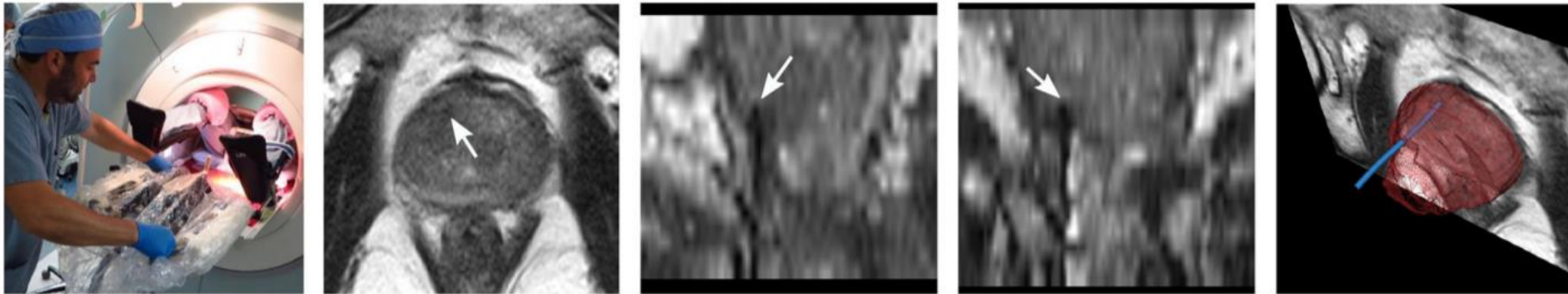
- Arteriolar width

- Venular width

- Vessel Tortuosity

Heslinga et al., SPIE-MI 2019

TU/e

# Segmentation

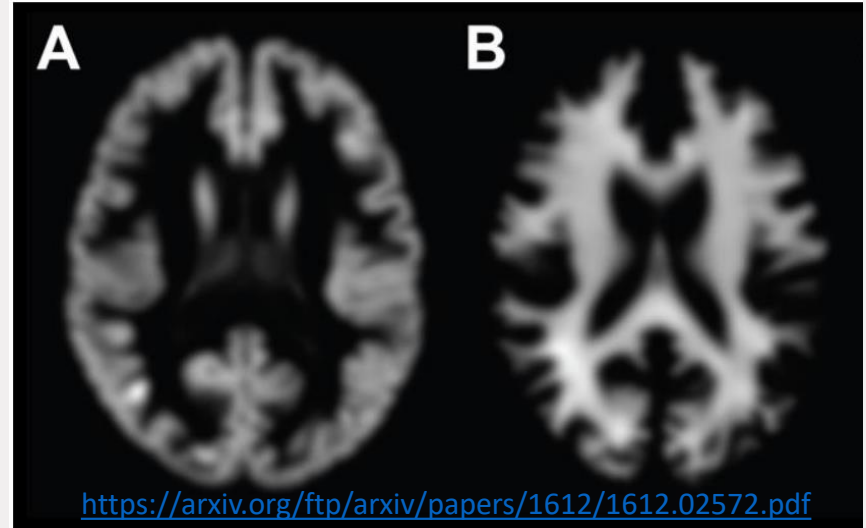- MRI-targeted prostate biopsy
- Needle segmentation

Mehrtash et al., IEEE Trans Med Imaging 2019

# What should be the output of our CNN?
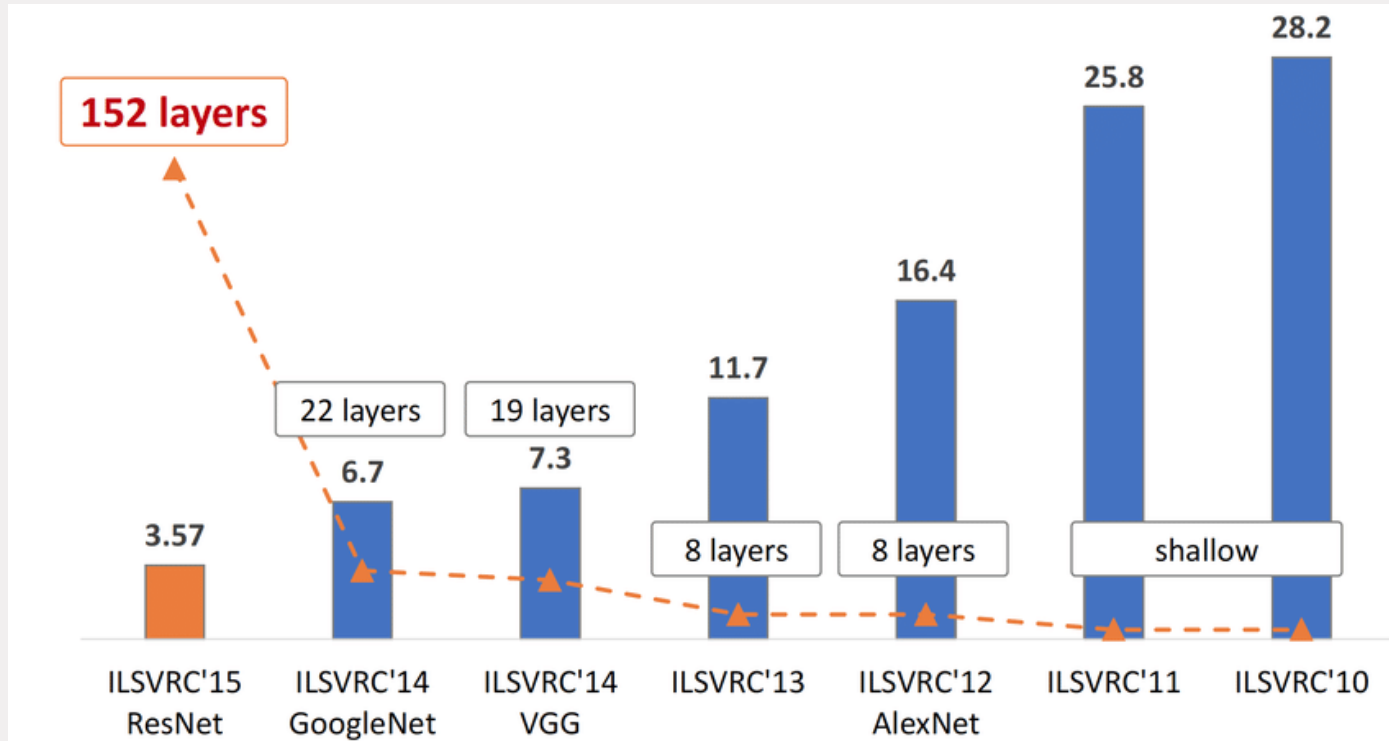


Retinopathy classification
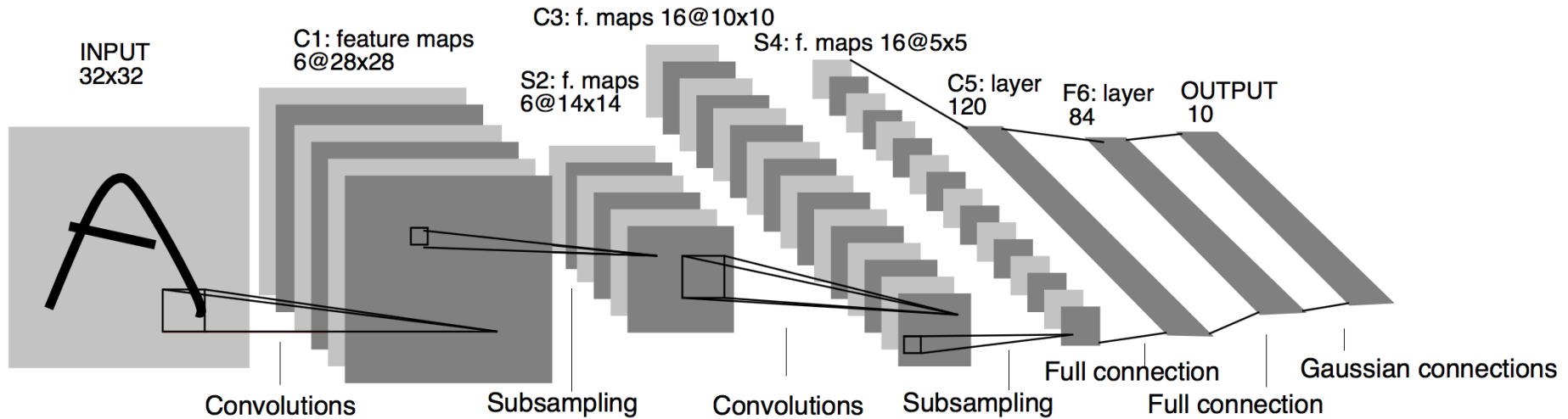


Brain age prediction

# IM*A*GENET

- 1,000 object classes (categories).
- Images:
  - 1.2 M train
  - 100k test.

TU/e

# Error rates ImageNet



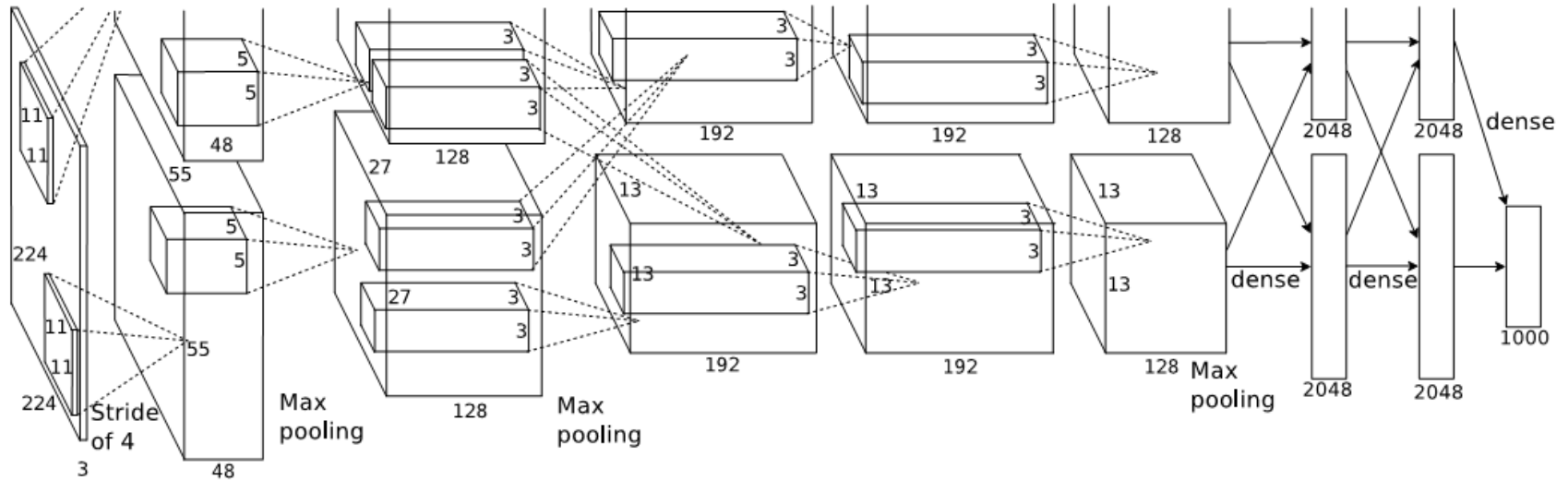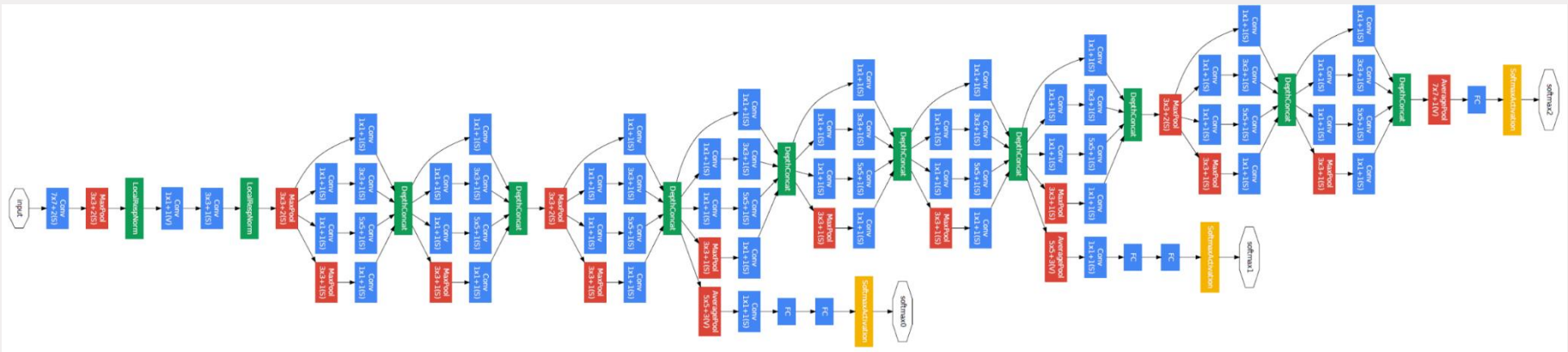8DC00 - Deep Learning Frameworks - Heslinga

TU/e

# LeNet

# AlexNet

- ImageNet top-5 error from 26% to 15.3%

# Inception v1 (GoogleNet)

- Inception modules
- Auxiliary targets
- Global average pooling layer



8DC00 - Deep Learning Frameworks - Heslinga

# Inception module

- Multiple scales

# ResNet



8DC00 - Deep Learning Frameworks - Heslinga

TU/e
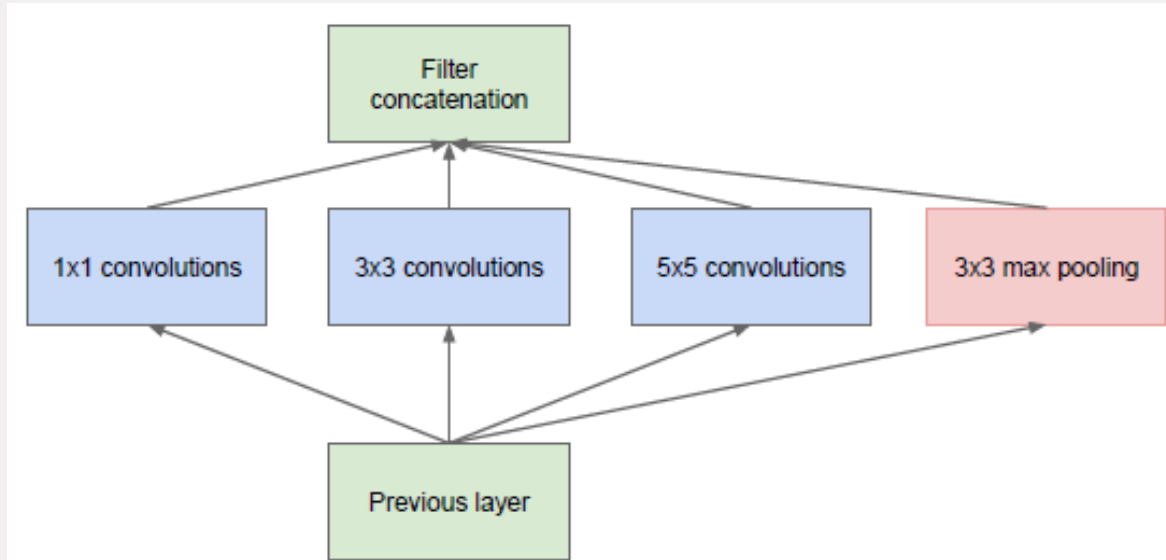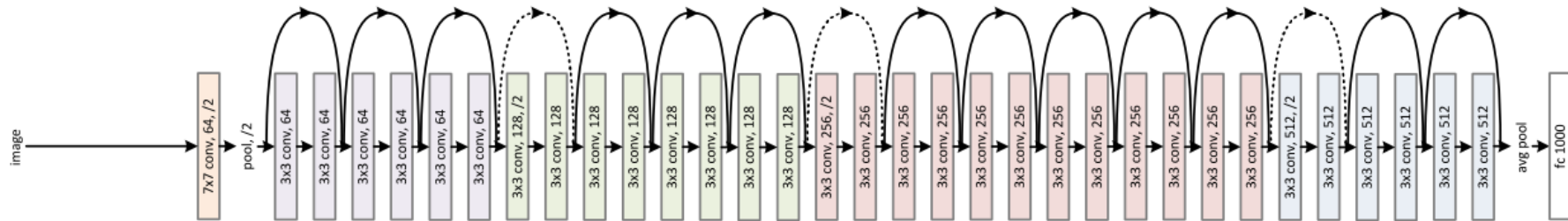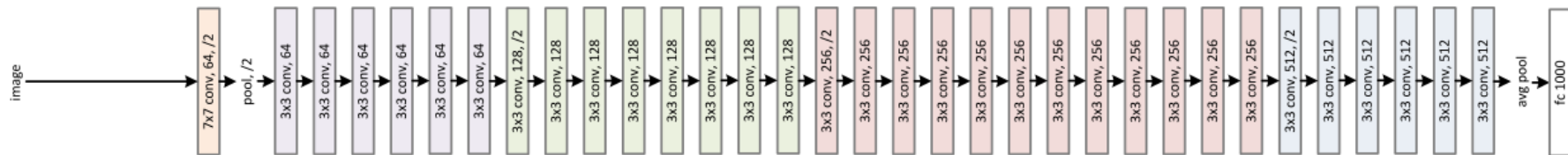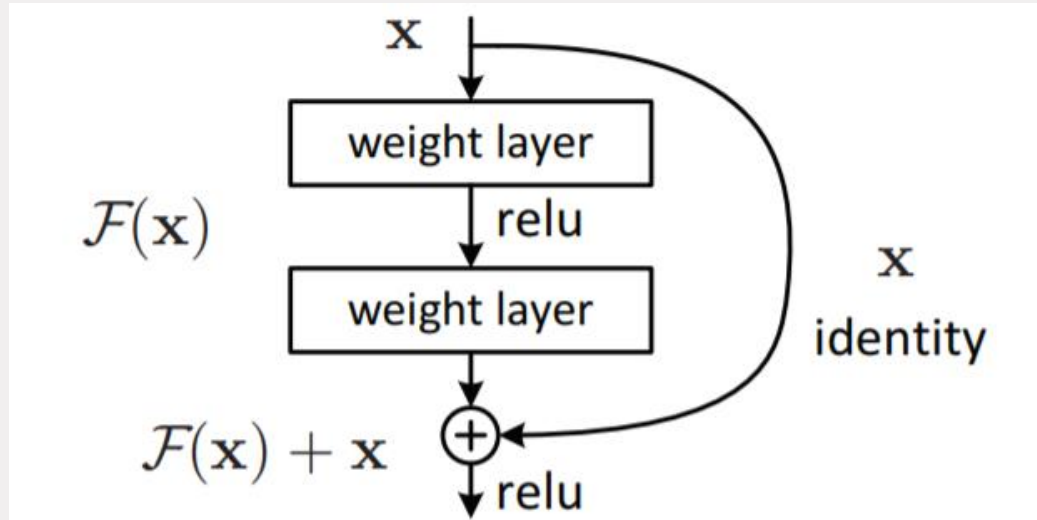
# Residual block

Learn differences between input and output layer = residue

# Error rates ImageNet (top 5 accuracy)



8DC00 - Deep Learning Frameworks - Heslinga
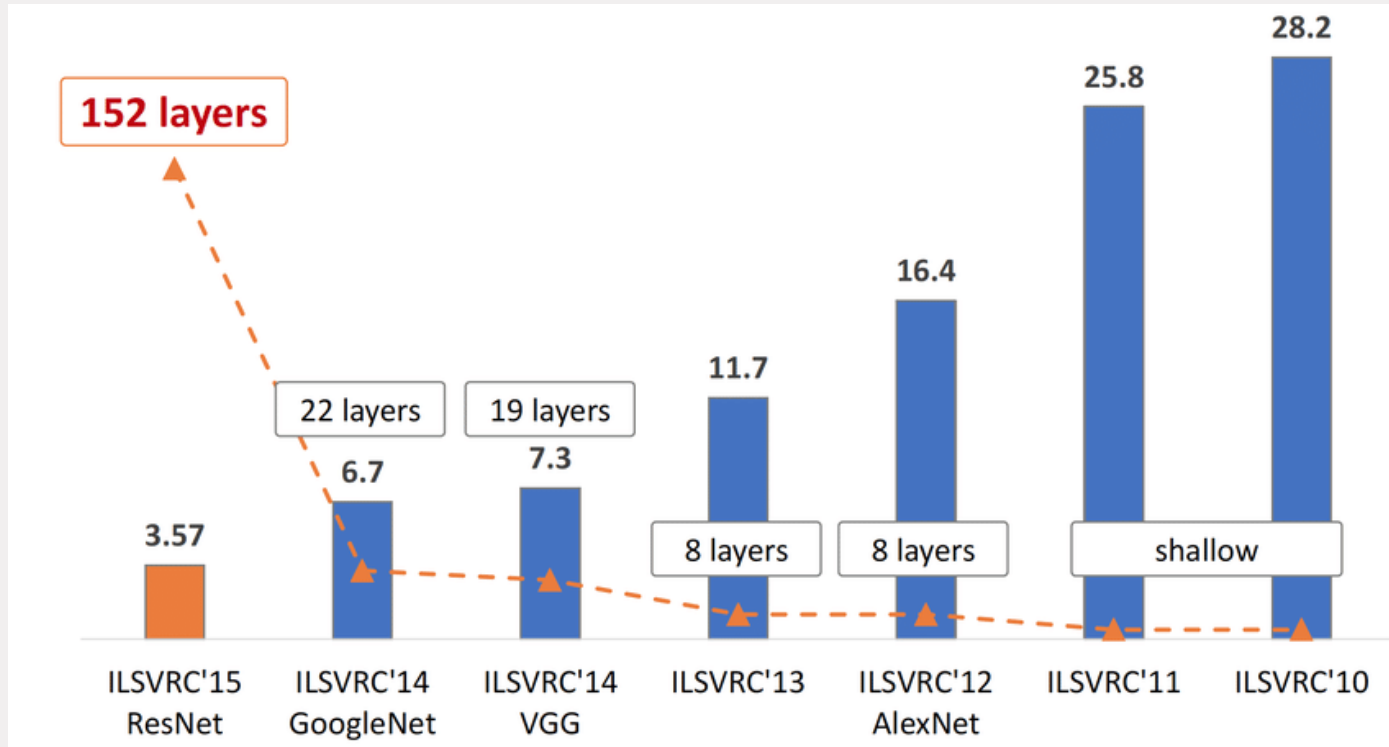
# U-Net: Segmentation

Ronneberger et al., MICCAI 2015

TU/e
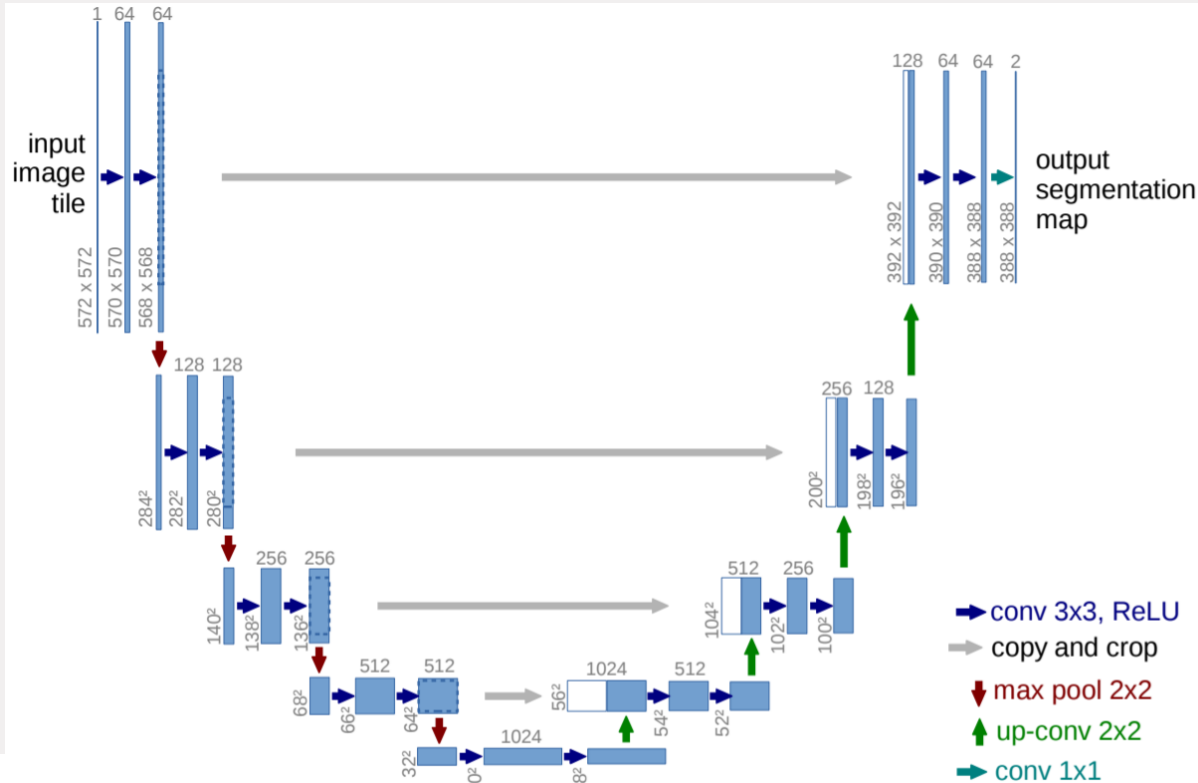
# Neural networks have been around



8DC00 - Deep Learning Frameworks - Heslinga
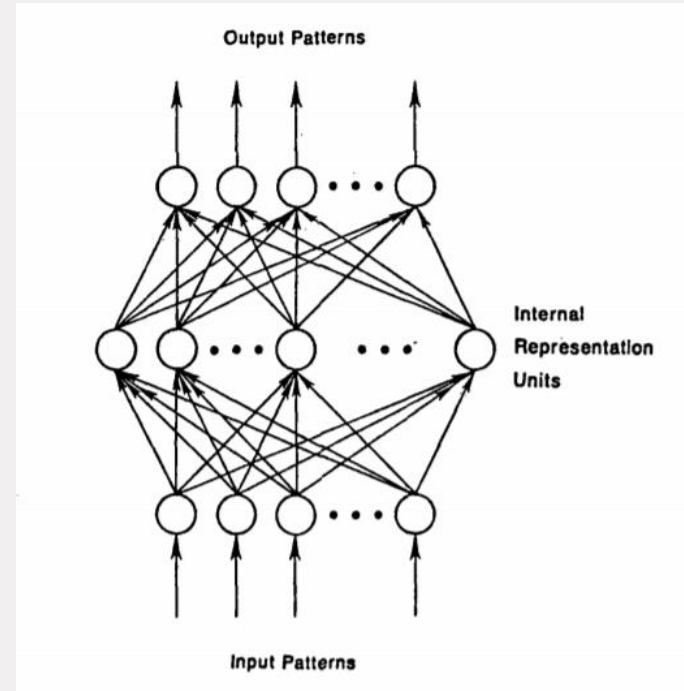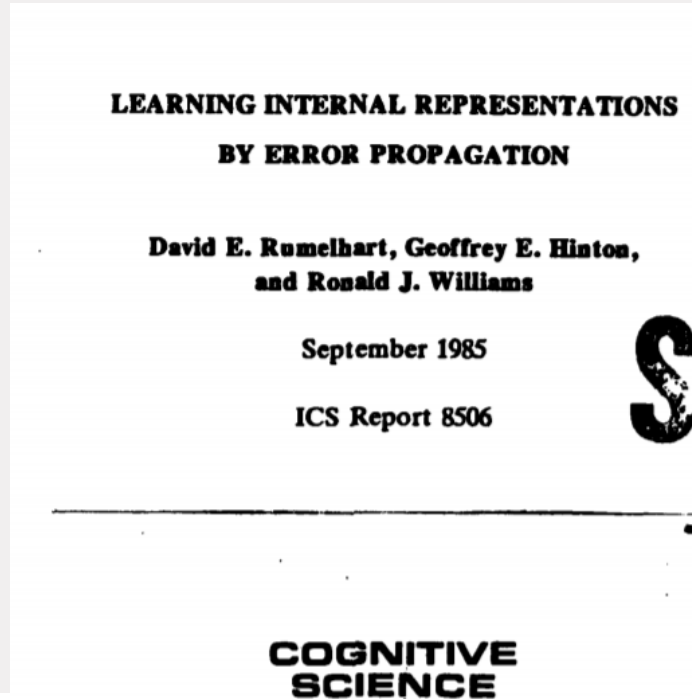
# Deep learning enabled

Why did it take so long to develop these deep learning models?



https://www.edx.org



https://www.greentechmedia.com
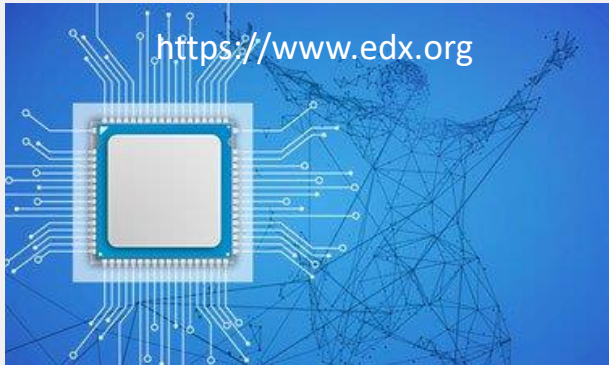
- Hardware improvements

- Parallel computing (GPU)

- Digital data

- More (image) data

= also true for medical imaging

TU/e

# Lecture outline

- Different applications → Different requirements

- Small exercise: "what should be the output?"

- History of deep learning models

- **Examples of deep learning models & applications**

- *Break (15 mins)*

- Examples of deep learning software

- Group exercise: *"choosing the right deep learning approach"*

TU/e

# Example 1: Skin lesion classification

129,450 clinical images

2,032 different diseases

3 classes

LETTER

**Dermatologist-level class**
**with deep neural network**

Andre Esteva[1]*, Brett Kuprel[1]*, Roberto A. Novoa[2,3], Justin Ko[2,
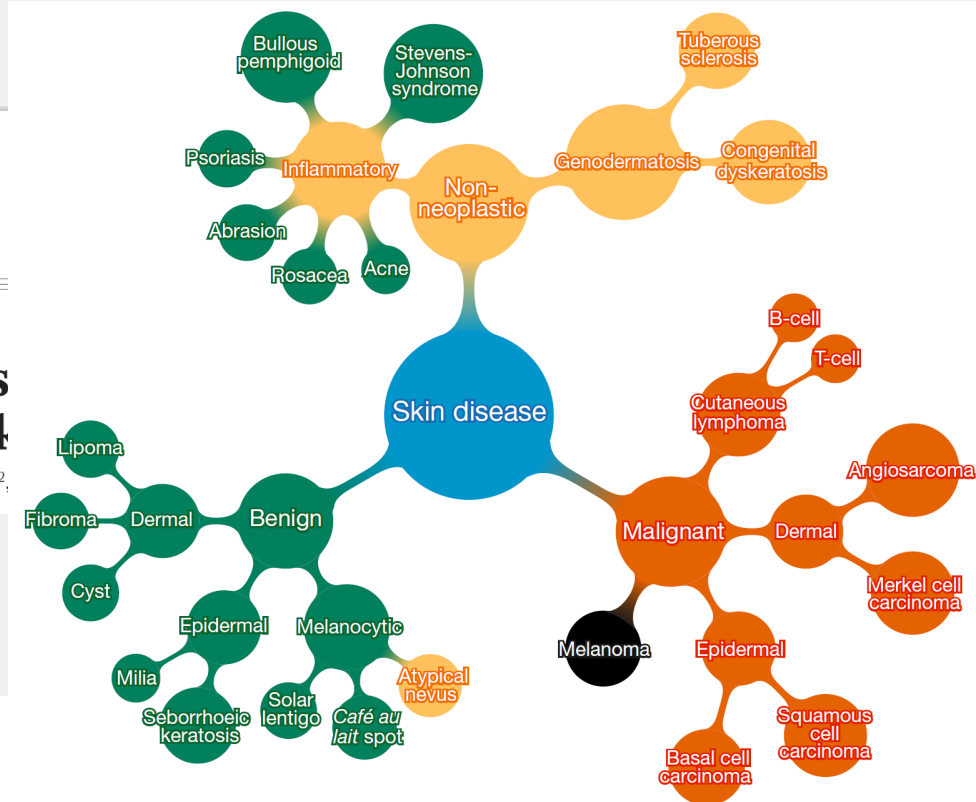
# Inception v3



Skin lesion image — Deep convolutional neural network (Inception v3) — Training classes (757)

- Convolution
- AvgPool
- MaxPool
- Concat
- Dropout
- Fully connected
- Softmax

Training classes:
Acral-lentiginous melanoma
Amelanotic melanoma
Lentigo melanoma
…
Blue nevus
Halo nevus
Mongolian spot
…

Inference classes (vary by task)

92% malignant melanocytic lesion

8% benign melanocytic lesion

TU/e

# Dermatologist-level?

- 3 class classification → 72.1 ± 0.9% (mean ± s.d.) accuracy

- Two dermatologist → 65.56% and 66.0% accuracy
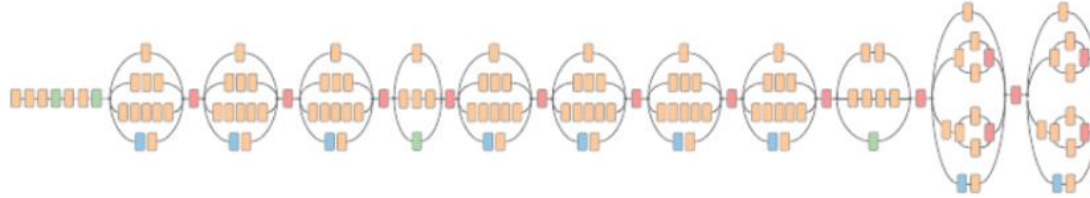
- But, only for 1 task!



LETTER

doi:10.1038/nature21056

Dermatologist–level classification of skin cancer with deep neural networks

Andre Esteva[1]*, Brett Kuprel[1]*, Roberto A. Novoa[2,3], Justin Ko[2], Susan M. Swetter[2,4], Helen M. Blau[5] & Sebastian Thrun[6]

TU/e

# Example 2: Retinal vessel geometry approximation

- Color fundus images

- Vessel geometry related to systemic diseases (e.g. Type 2 Diabetes, hypertension)

- Approximate geometries measured with (slow) classic image analysis tools

| Biomarker | Description |
|---|---|
| Arteriolar width | Average width of arterioles within 0.5 to 1.0 disc diameters from the optic disc border |
| Venular width | Average width of venules |
| [...] | |

TU/e

# ResNet-50

$$\hat{y} = \boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}$$

- Replace final layer
  - # of output nodes = # of biomarkers
- Regression problem!
  - No activation function in the final layer

TU/e
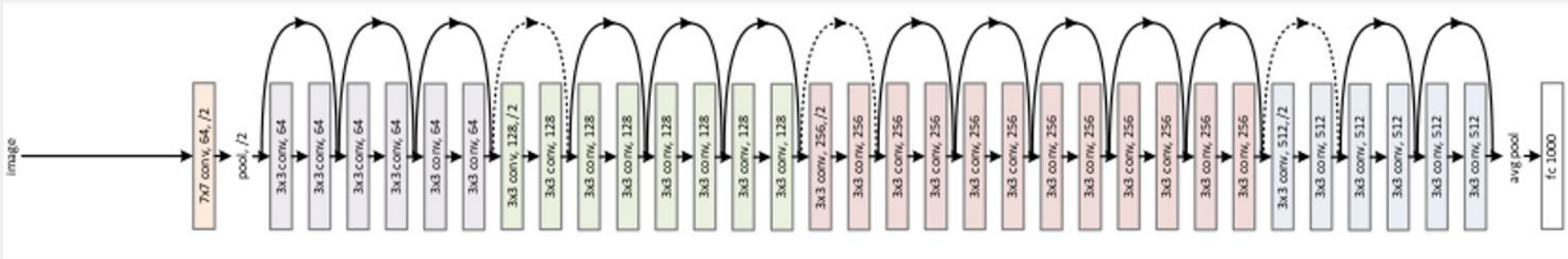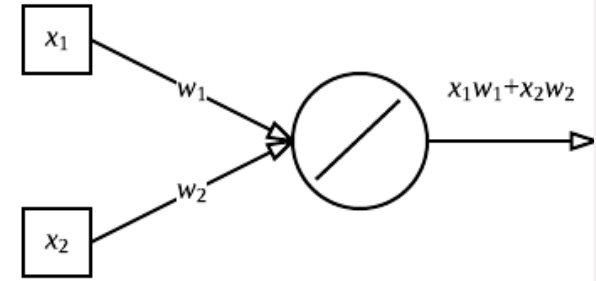
# Results – Intra-Class Correlation

Venules

Arterioles

8DC00 - Deep Learning Frameworks - Heslinga

TU/e

# Example 3: Brain Tumor Segmentation

- MRI brain volumes (3D)

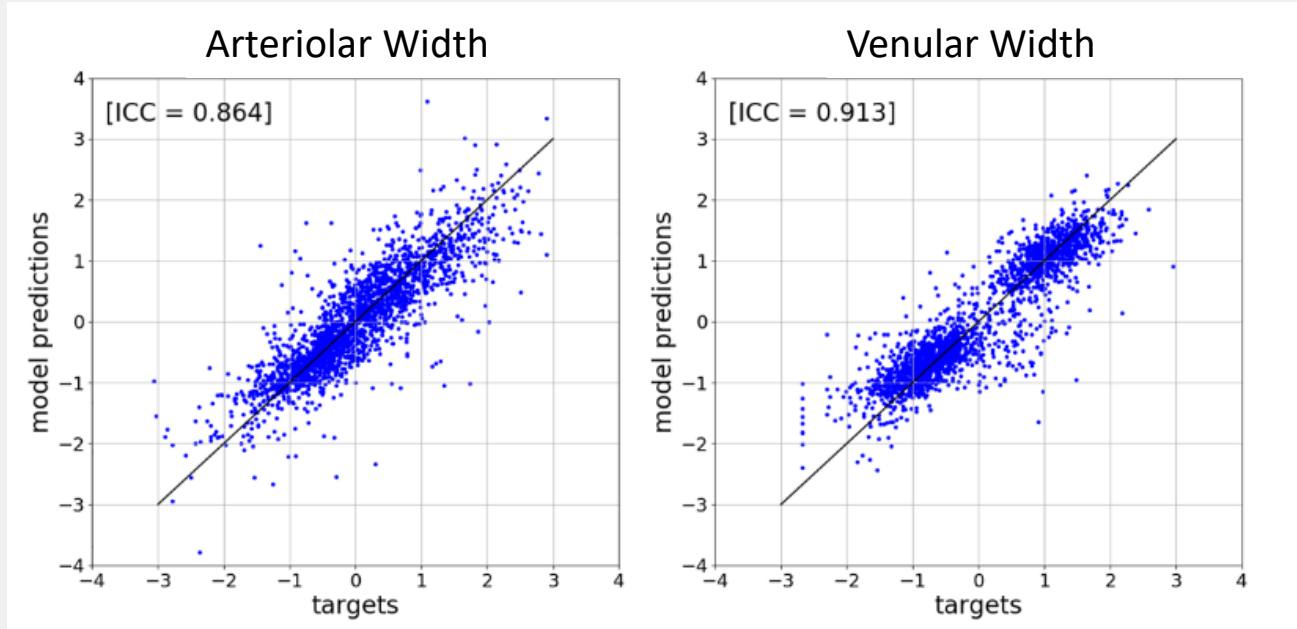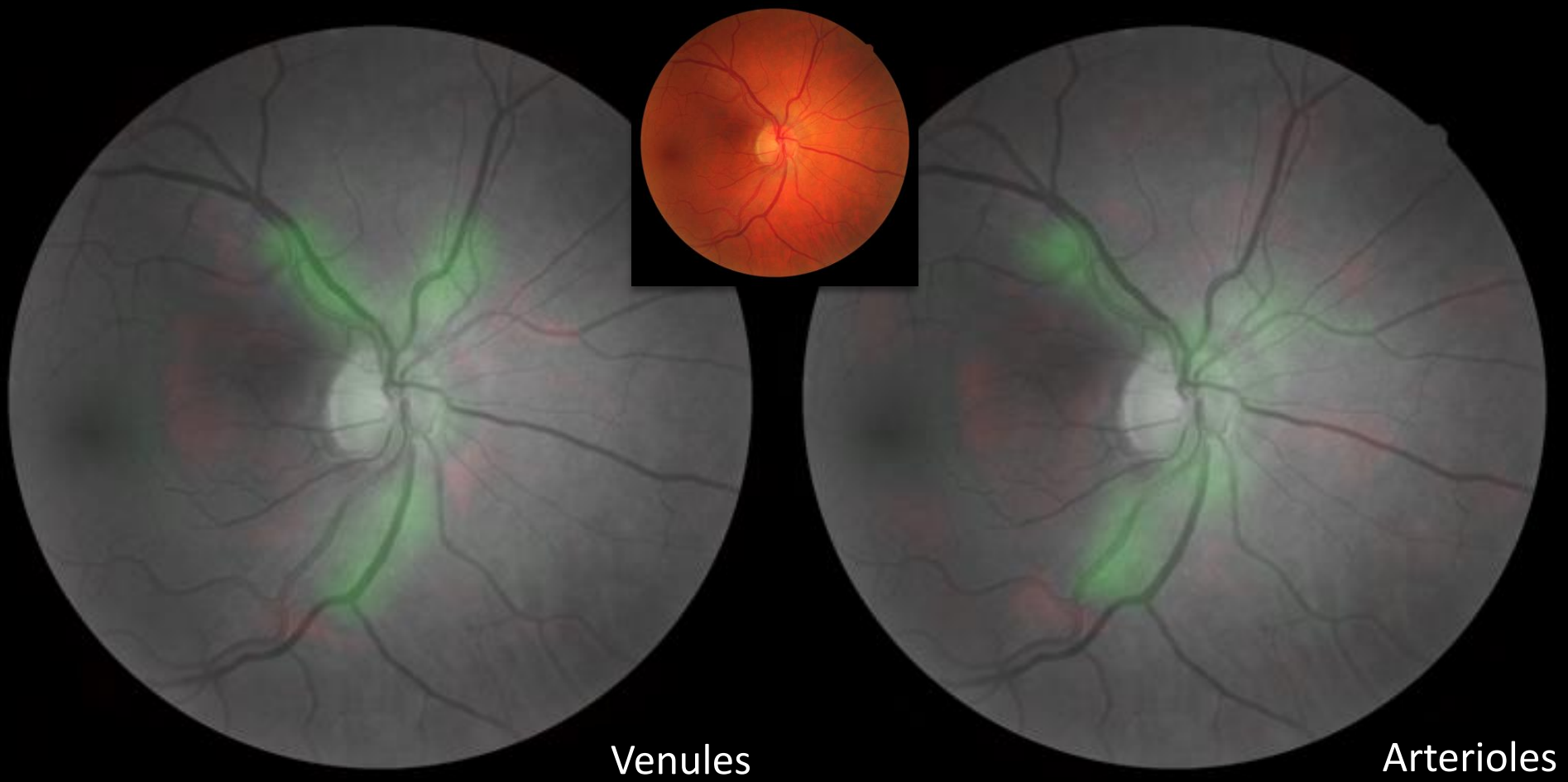- FLAIR images

- Tumor segmentation



**Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks**

Hao Dong [†1], Guang Yang [‡2,3], Fangde Liu [†1], Yuanhan Mo [1], Yike Guo [‡1]
[1] Data Science Institute, Imperial College London, SW7 2AZ, London, UK
[2] Neurosciences Research Centre, Molecular and Clinical Sciences Institute, St. George's, University of London, London SW17 0RE, UK
[3] National Heart and Lung Institute, Imperial College London, SW7 2AZ, London, UK
[†] Co-first authors contributed equally to this study.
[‡] Corresponding author.
{hao.dong11, g.yang, fangde.liu, y.mo16, y.guo}@imperial.ac.uk

TU/e

# U-Net

TU/e

# Segmentation results



8DC00 - Deep Learning Frameworks - Heslinga
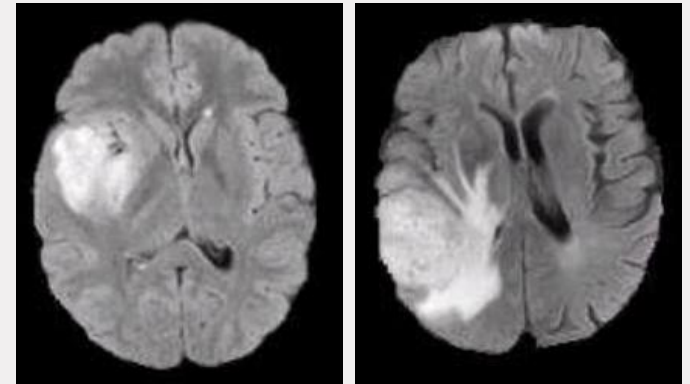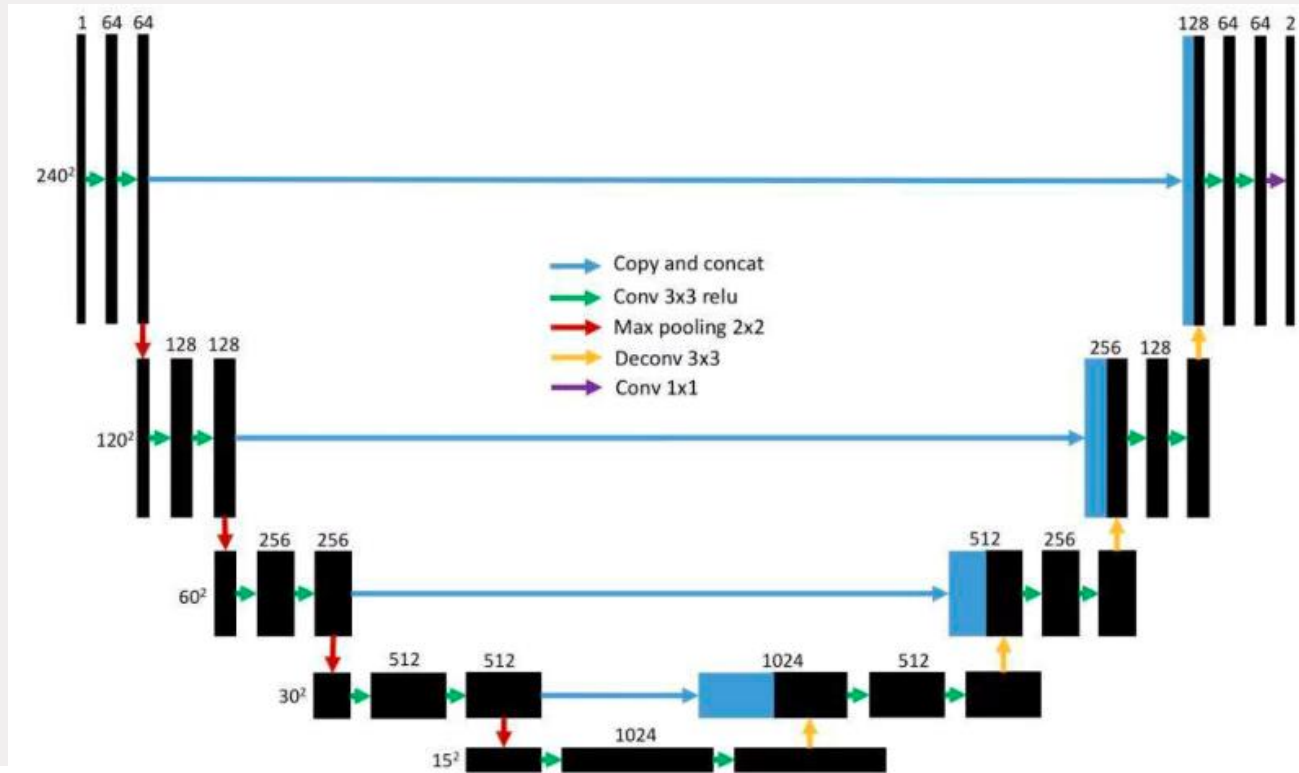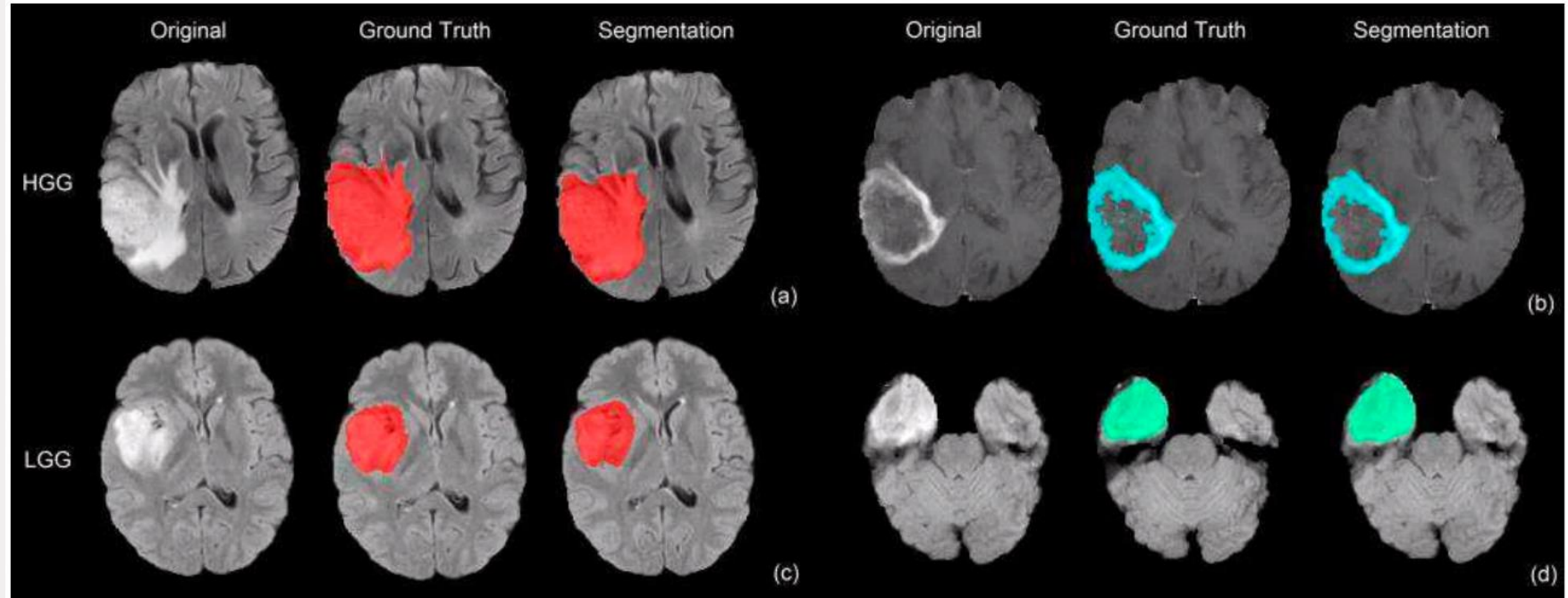
# Break!



**Brown-throated sloth (Panama)**
https://en.wikipedia.org/wiki/Sloth

TU/e
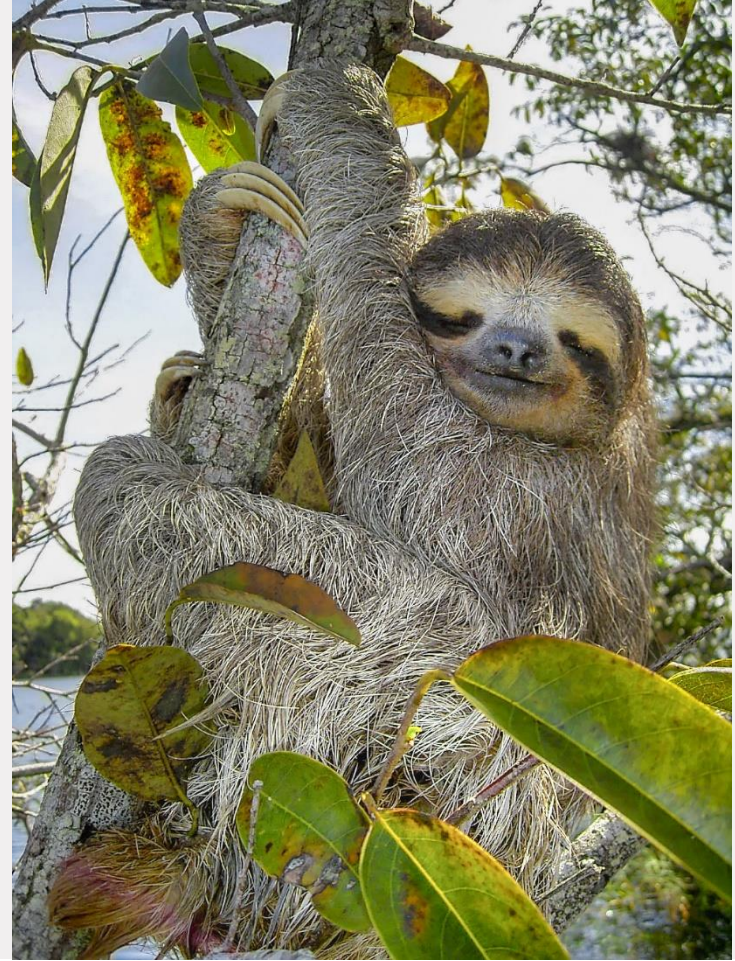
# Deep learning frameworks

- Python / Matlab / R

- Tensorflow / PyTorch / Theano [RIP] / Deep learning toolbox for Matlab

- Keras / Sonnet


- Models on GitHub

TU/e

# Keras

## Classify ImageNet classes with ResNet50

```python
from keras.applications.resnet50 import ResNet50
from keras.preprocessing import image
from keras.applications.resnet50 import preprocess_input, decode_predictions
import numpy as np

model = ResNet50(weights='imagenet')

img_path = 'elephant.jpg'
img = image.load_img(img_path, target_size=(224, 224))
x = image.img_to_array(img)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)

preds = model.predict(x)
# decode the results into a list of tuples (class, description, probability)
# (one such list for each sample in the batch)
print('Predicted:', decode_predictions(preds, top=3)[0])
# Predicted: [(u'n02504013', u'Indian_elephant', 0.82658225), (u'n01871265', u'tusker', 0.1122357),
#             (u'n02504458', u'African_elephant', 0.061040461)]
```

TU/e

# Group exercise

TU/e
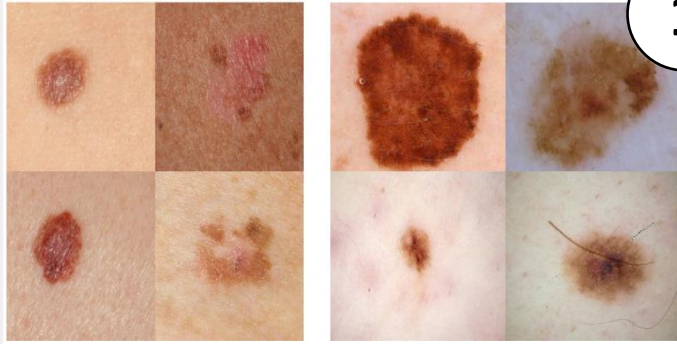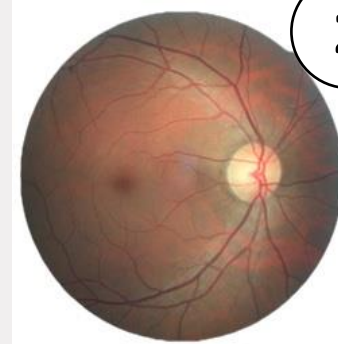
# Group exercise

- Type of problem?

- Preprocessing steps?

- What kind of 'labels' do you need?

- What should the output look like?

- What should be the activation function in the final layer?

- What model do you choose?

- How do you evaluate performance?

- Other considerations?

TU/e

# Problem 1

The Netherlands Cancer Institute (NKI) would like to develop an app that enables users to automatically assess skin lesions. The app should be deployable on any medium-end phone and all calculation should be done locally. The NKI has requested your group for general advice on the deep learning approach they should take.



Esteva et al., Nature 2017

TU/e
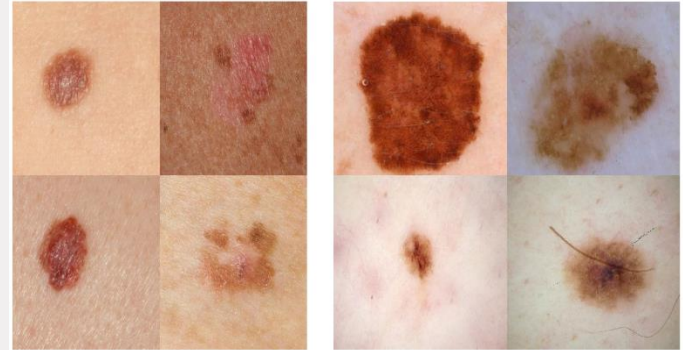
# Problem 2

Your team is competing in an online challenge. The objective: classification of retinal fundus images in one of five stages of diabetic retinopathy. Class 0 = no retinopathy. Classes 1-4 correspond to increasing severity of retinopathy. What is your approach?



https://www.kaggle.com/c/diabetic-retinopathy-detection

TU/e
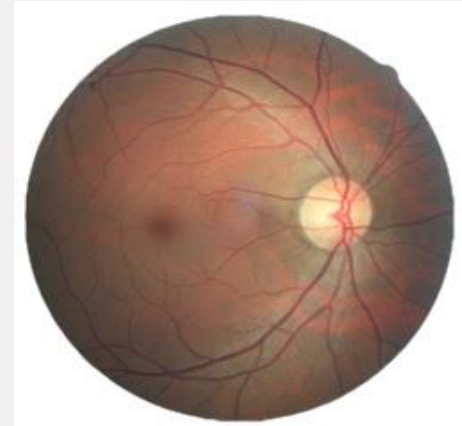
# Problem 3

As part of a university project you are working on the development of an app that uses pictures of people's face to predict somebody's age. You can use the server of the university for the calculations.



https://www.freecodecamp.org/news/how-to-build-an-age-and-gender-multi-task-predictor-with-deep-learning-in-tensorflow-20c28a1bd447/

TU/e

# Problem 4

A vascular surgeon performs endovascular surgery using x-ray angiography. She would like to continuously see the veins and arteries clearly distinguished from the background on the monitor next to her. How would you approach this?



https://blog.definitivehc.com

TU/e

# Summary

- Different applications → different requirements

- History of deep learning models

- Deep learning enablers (computational power and data)

- Deep learning frameworks

**TU/e**

8DC00 - Deep Learning Frameworks - Heslinga

TU/e