Machine Learning for Image Analysis, SoSe 2024
Exercise Sheet #6: U-Net & Resources
Due date: June 4, 2024, before 11 am

Problem 1 (Computing Resources) (13 pt).
Your goal is to compute the memory consumption of an exemplary Neural Network. This network takes as input RGB-images of size 400x400. The first layer is a convolutional layer with 32 filters of size 7x7 each, stride 3 and using no padding. The second layer is a max pooling layer with pooling kernel size 3x3, stride 3 and using no padding. The last layer is a a fully connected layer with output size 10 (e.g. the scores for 10 different classes)

(a) (3pt) Calculate the output size after the convolutional layer and after the pooling layer.

(b) (3pt) Calculate the number of weights (incl. bias) used in the convolutional layer.

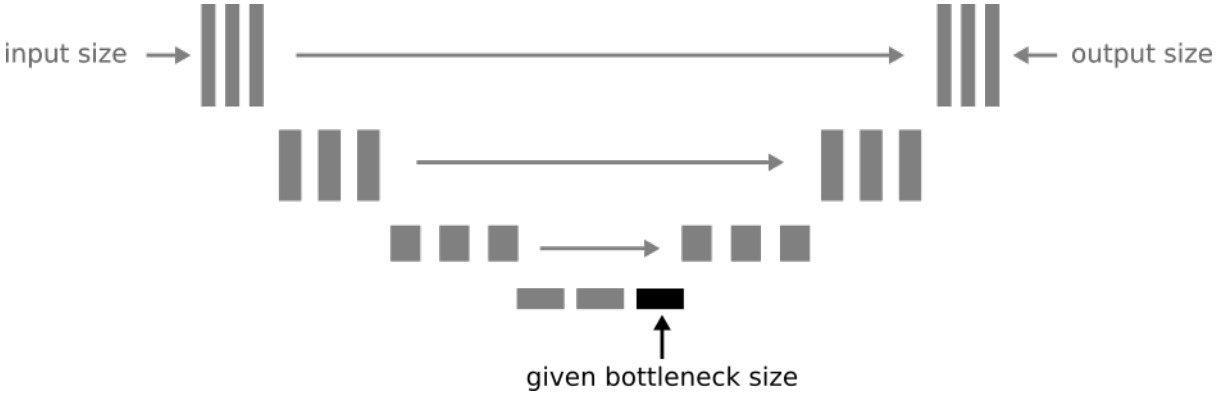
(c) (3pt) Calculate the number of floating point operations (FLOP) needed to compute the output of the convolutional layer.

(d) (4pt) Calculate the number of weights (incl. bias) used in the pooling layer and in the fully connected layer.

(e) (3pt) Calculate the number of FLOP needed to compute the output of the fully connected layer.

Note: In the context of this problem we only consider multiplication as a relevant floating point operation, and ignore the workload by addition.

Problem 2 (U-Net) (12 pt).
You want to apply a 2d 4-level U-Net for grayscale images. At each level, the U-Net has two consecutive convolutional layers with filter size 5×5, stride 1 and no padding. For down- and upsampling we use kernel size 2×2, stride 2 and no padding. We start with 16 feature maps (number of filters) in the first level and double them after each downsampling step (and halve them after each upsampling).



(a) (6pt) Calculate input and output sizes for the following bottleneck sizes:

1. 11×11
2. 20×20

(b) (2pt) Compute the number of weights (incl. bias) for (a1) for the first two convolutional layers.

(c) (2pt) Compute the number of floating point operations (FLOP) for (a1) for the first two convolutional layers. Here we only consider multiplications as relevant FLOP and ignore any additions.

(d) (2pt) Compute the receptive field in the input image of a neuron in the last layer of the bottleneck .