The assignments in EML must be completed individually and written individually. Group discussions are allowed and encouraged, but in such cases you should list the group members in your handin.

Exercise 3

Neural Networks

In this exercise, you will develop and evaluate a neural network for a medical image segmentation task. You will hand in
1) a report (single pdf document, not zipped) based on your experiments and including your answers to the questions below, including code snippets or pseudocode where relevant (upload to Absalon);
2) your code (source files, zipped in a single file; upload to Absalon);
3) your segmentation result images (5 binary images, indicating 1 for each pixel taht is a vessel and 0 for each background pixel, named the same as their input, so 1_test.tif … 5_test.tif, zipped in a single file named abc123.zip, where abc123 should be replaced with your kuid); This zip file with the segmentation result images, (not the pdf with the report) should be uploaded to https://www.dropbox.com/request/zpT6939DCgMUnMC6B0uF .

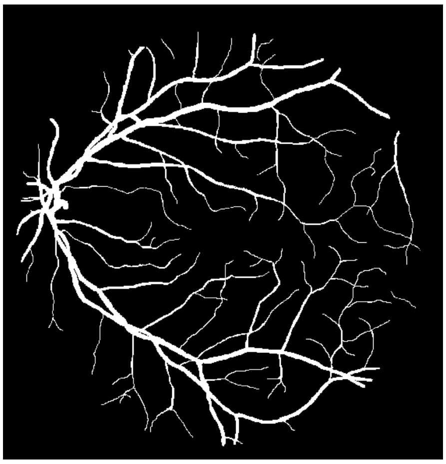
Please note that even though you provide your code as well, we should be able to understand all your answers from the main pdf document, you should not expect us to dig through the code. Use code-snippets in the pdf to help explain your answers.
We recommend using a cloud-based GPU computing resource such as https://colab.research.google.com/ for this exercise (we also tested it on a laptop with a Nvidia 950m GPU with 2GB of memory, which worked as well).

The task is to segment the blood vessels on photographs of the retina. This is an important step in recognizing different eye diseases.

Download the training data with their labels obtained by manual segmentation, as well as the test data (without labels) from
https://www.dropbox.com/sh/jhvugjvowcnuovb/AADHwtqWk7p2y7KkO8OUg_lha?dl=0

The expected result is 5 binary images, one for each image in the test set in with dimensions matching the original test images, indicating which of the pixels are blood vessels (0 = background, 1 = blood vessel; see example in the figure).

Retinal photograph



Vessel segmentation of image on the left

1-5; Preprocessing, designing network and experiments - 25 points, 5 each

1. Describe how you will use the training images to ensure a good generalization to the test images.
2. Visualize the images, both in color and all three (RGB) channels separately. Consider if preprocessing might be required to train a good model. If not, describe why not. If yes, describe the preprocessing steps in detail and why you expect them to help.
3. Consider what neural network architecture would be suitable to solve this task and describe and justify your choice. For instance, consider the number, type, and size of layers, what connectivity, what non-linearity, which loss function.
4. What is the receptive field of this network? Explain your computation. (Note, that receptive field was covered in the Convolutional neural network lecture. In case you chose a network that has both a contracting and an expanding path, such as U-Net, it is sufficient to compute the receptive field for only the contracting path.)
5. Neural networks can be time consuming to train. Consider - and describe - at least two different ways to simplify the problem to keep training time manageable. Make sure to also describe their drawbacks. For instance, do you need to train on all images/whole images/full resolution/all three RGB channels? What are other ways to reduce computation time? In the rest of the exercise, feel free to make choices that might lead to less accurate results but would make your experimentation significantly faster, but make sure to explain your choices and the problems it may cause.

6-10 implementation, training, and analysis - 50 points, 10 points each.

6. Implement your network of choice and train it. Note: you need to make the implementation yourself, downloading and applying a pretrained model is not allowed. Explain your implementation.
7. Show in a figure how the loss evolves during training. Do the same on a smaller training set. Do you have enough training data to make a good model? How do you know?
8. Does the model overfit or underfit? How do you know? Show specific results that back up your arguments. If your model underfits, try to make it overfit, and if it overfits try to make it underfit, by adjusting hyperparameters and show your results. Describe and explain your observations.
9. Experiment with at least two different loss functions that are suitable for this problem, and explain why they are suitable. Examine the performance of these models and visualize the results. What do you observe? Why does this happen?
10. Experiment with at least two sets of choices for the model hyperparameters and show the results. Try explain the differences you observe.

11-12 Results evaluation and possible improvements - 25 points - 20 points for 11, 5 for 12.

11. A common performance metric for segmentation is the Dice Similarity Coefficient, defined as twice the volume of the intersection of the resulting segmentation with the ground truth divided by the sum of the two segmentation volumes: $2\times|A \cap B| / (|A| + |B|)$, where A is the **binary** segmentation output image and B is the ground truth segmentation (label) image.
Submit your segmented test images for the model that you think performs best on this metric. What value of Dice coefficient do you expect this result to have (approximately), and why? Show specific results to back up your arguments.
Hint: Note that, if you made any changes to the test images such as cropping, down/up sampling, or rotating, the intersection with the ground truth segmentations is likely to be low.

12. How could you improve these results, if you had more time? List at least two suggestions and justify your answer.