Assignment4:

Goal: develop a RGB-Depth fusion architecture for semantic segmentation based on Fully Convolutional Network (FCN) .

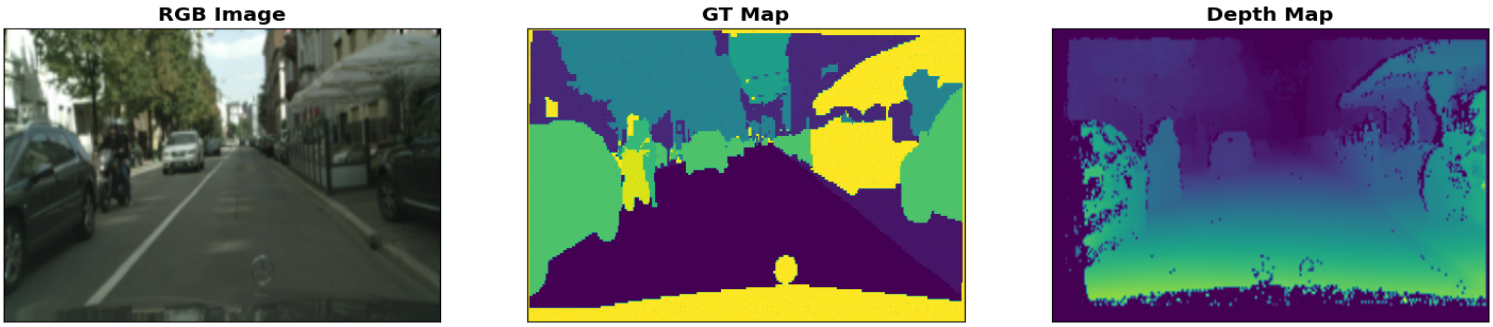
This project should be done as a team work (2 persons) or you can do it alone.

Deadline: 26.03.2024 at 24:00.

Tool: Keras, Jupyter notebook, follow the installation instruction in "GettingStarted" in the "Assignment" section in Moodle.

Data set: Dataset consists of 1100 (per modality) images of road scenes. It is divided into train (600 images), test (200 images) and validation (300 images) datasets. Change the size of all images into 256*256.

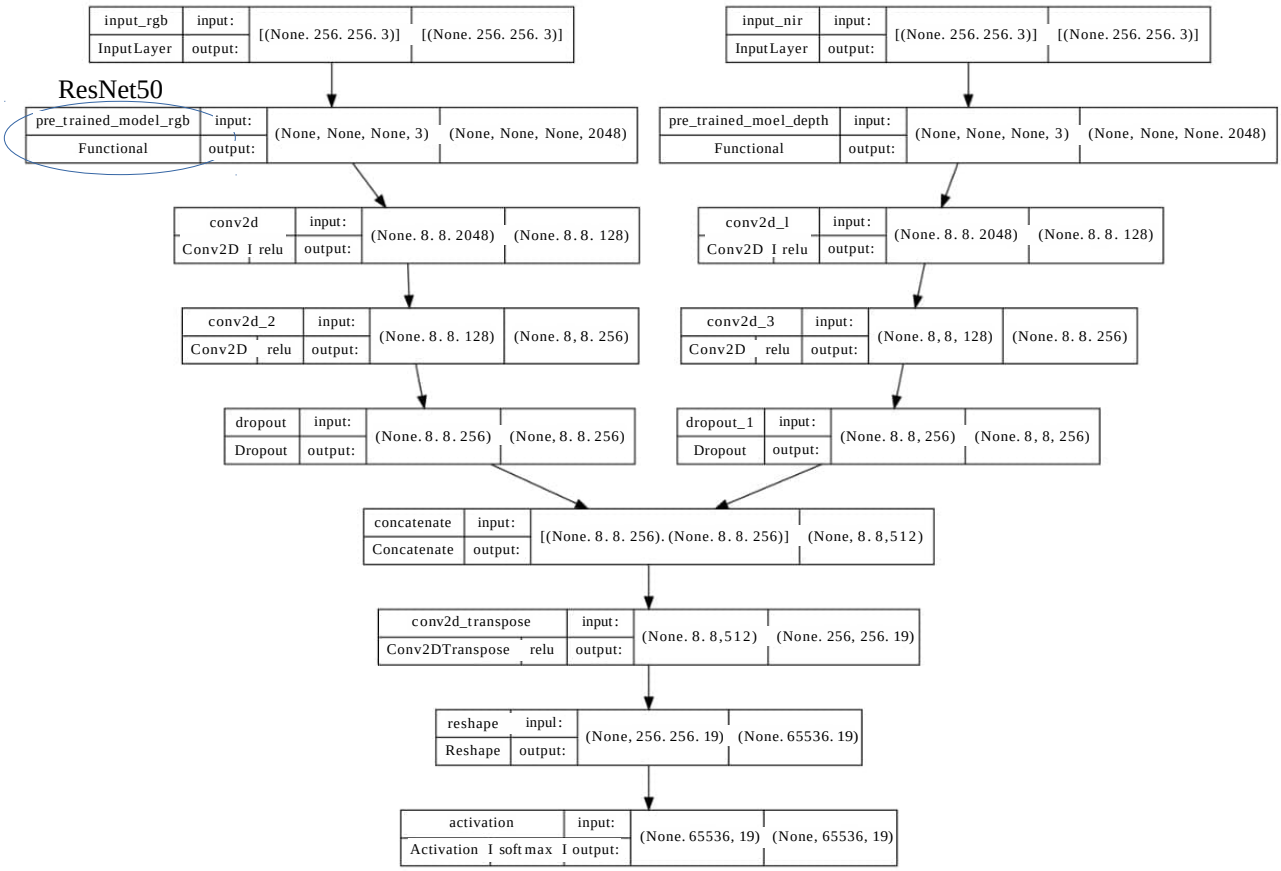
- Pixel-level annotations (ground truth) are available for 19 semantic classes.
- Available modalities: RGB, Depth



Tasks: There is a Jupyter notebook template (**Assignment4_template.ipynb**) for the exercise that you need to write your code in the template. In summary, you will do the following tasks in the template notebook:

- Define a Fully Convolutional Network (FCN) for image segmentaion by fusing RGB and depth images. The network consists of two sterams which each stream has following layers:

  1. Use the pretrained ResNet50 on imageNet
  2. Two Conv layers with 128 and 256 nodes, respectively. Kernel size (3,3), stride (1,1)
  3. Top of the Conv layers, add dropout layer with 0.2
  4. Concatenate two streams.
  5. Add a transposed convolution layer (Conv2DTranspose) with Kernel size (64,64), stride (32,32)
  6. Add a reshape layer (tf.keras.layers.Reshape) to reshapes inputs into the given shape.
  7. Add a softmax activation layer



Layer (type)	Output Shape	Param #	Connected to
input_rgb (InputLayer)	[(None, 256, 256, 3)]	0	[]
input_nir (InputLayer)	[(None, 256, 256, 3)]	0	[]
pre_trained_model_rgb (Functional)	(None, None, None, 2048)	23587712	['input_rgb[O][0]']
pre_trained_moel_depth (Functional)	(None, None, None, 2048)	23587712	['input_nir[O][0]']
conv2d (Conv2D)	(None, 8, 8, 128)	2359424	['pre_trained_model_rgb[0][0]']
conv2d 1 (Conv2D)	(None, 8, 8, 128)	2359424	['pre_trained_moel_depth[0][0]']
conv2d_2 (Conv2D)	(None, 8, 8, 256)	295168	[1conv2d[0][0]1]
conv2d_3 (Conv2D)	(None, 8, 8, 256)	295168	[1conv2d 1[0][0]']
dropout (Dropout)	(None, 8, 8, 256)	0	[1conv2d 2[0][0]']
dropout_l (Dropout)	(None, 8, 8, 256)	0	[1conv2d 3[0][0]']
concatenate (Concatenate)	(None, 8, 8, 512)	0	['dropout[0][0]', 'dropout_l[0][0]1]
conv2d transpose (Conv2DTranspose)	(None, 256, 256, 19)	12582918	['concatenate[0][0]']
reshape (Reshape)	(None, 65536, 19)	0	['conv2d transpose!©][0]']
activation (Activation)	(None, 65536, 19)	0	['reshape!©][0]']

==========

Total params: 65,067,526
Trainable params: 64,961,286
Non-trainable params: 106,240

- Compile the model with SGD(learning_rate=0.008, decay=1e-6, momentum=0.9) and loss="categorical_crossentropy"
- Train the model on the “train” dataset and “validation”dataset for epochs =10.
- Evaluate the model on the test dataset.
  - Print loss and accuracy of model for test dataset.
  - Predict semantically segmented images on 5 random example of test dataset.

Instruction for Submissions: You must convert your notebook to a PDF file and then submit the PDF file to Moodle. There are two ways for converting a notebook to PDF:

1. *Install PyPDF2* by running
→ *pip install PyPDF2*
2. you can manually convert the jupyter notebook to HTML (*File -> Download as -> HTML (.html)*), save the HTML page as a PDF.

Important:
- Please make sure that the submitted notebook has been run and the cell outputs are visible.
- The describtion of each task should be added into the notebook as a comment in order to make your code easier for understanding.

Assignment Evaluation: The notebook is evaluated on 0-20 points scale (20 means that the work is rejected).

Extra 5 points: implement FCNs for each sing modality and compare their accuracy with fusion model. I need the result of the following table in the same notebook.

Modality	**Test accuracy(%)**
RGB	?
Depth	?
RGB+Depth	?