Project 3

Description

You should create a system that will solve the problem of computer vision of your choice with the use of neural networks.
The problem you choose, along with the dataset, should be unique in all groups e.g:

- several pairs can solve the same problem but must have different datasets
- or you can use the same dataset but solve a different problem

Main requirements:

- Preparation and presentation of a selected data set (anyone based on visual data),
- Preparation and presentation of the main functionality that should be based on neural networks,
- Presentation of the report,
- The final grade depends on the number of points scored for each element of the project

Problem:

Requirements

Problem	Points
Semantic segmentation	1
Instance segmentation	3
Depth estimation	1
Domain adaptation	3
Super-resolution	3
Image inpainting	3
Search engine	2
OCR of handwritten text	2
3D reconstruction	4

There can be no classification or regression as the primary problem, others are possible for discussion.

Additional points:

- Solving an additional problem to improve prediction quality (for example, additional loss functions for object detection and classification) +1pk

Model:

Requirements

1st model	Points
pre-trained model on the same problem	0
pre-trained model on the different problem (transfer-learning)	1
ready architecture trained from scratch	1
own architecture (over 50% of own layers)	2

Additional points:

- Each subsequent model with a different architecture +1pk, for next own architecture (over 50% of own layers) +2pk
- +1pk for a every non-trivial solution in own architecture (use of attention, GAN, RL, contrastive learning, metric learning)

Dataset:

Minimal size of input image 200x200px.

Requirements

- At least 1000 photos
- not mnist

Additional points:

- Evaluation on a set with at least 10000 photos +1pk
- Your own part of the dataset (> 500 photos) +1pk

Training:

Requirements

- The correctly selected loss function
- Split data into train, validation and test set
- Performance metrics (at least 2)
- Training dynamics metrics (at least 3 not counting loss)

Additional points:

- Hyperparameter estimation +1pk
- Adaptive hyperparameters +1pk
- Architecture tuning (at least 3 architecture) +1pk
- Overfitting some examples from the training set +1pk
- Data augmentation +1pk
- Cross-validation +1pk
- Distributed learning +1pk
- Federated learning +2pk
- Testing various loss functions (at least 3) +1pk
- Calculating intrinsic dimension +1pk
- Custom optimizer +1pk

Tools:

Requirements

Git with Readme

Additional points:

- MLflow,Tensorboard, Neptune, Weights & Biases (along with some analysis of experiments) +1pk
- Run as docker/ docker compose +1pk
- REST API or GUI for example Gradio, Streamlit +1pk
- DVC +2pk
- Every other MLOps tool +1pk
- Label Studio or other data labeling tools +1pk
- Explanation of 3 predictions - e.g. which inputs were most significant +2pk

Report

Requirements:

- description of the data set, with a few image examples
- description of the problem
- description of used architectures with diagram showing the layers; For large models containing blocks, the blocks and the connections between them can be shown separately.
- model analysis: size in memory, number of parameters,
- description of the training and the required commands to run it
- description of used metrics, loss, and evaluation
- plots: training and validation loss, metrics

- used hyperparameters along with an **explanation** of each why such value was chosen
- comparison of models
- list of libraries and tools used can be a requirements.txt file
- a description of the runtime environment
- training and inference time,
- preparation of a bibliography - the bibliography should contain references to the data set (preferably the article in which the collection was presented) and all scientific works and studies, including websites with tips on the solution.
- **A table containing all the completed items with points.**
- Link to Git

Grading

If you do the project alone the total number of points needed for the grade is lower than if you do the project in a pair.

3:
All elements from report, and code
Problem >= 1pk
Model >= 1pk
Sum of additional points from dataset, training, tools, report >= 3pk
Sum of points >=6 (pair) 5 (alone)

4:
Problem >= 1pk
Model >= 2pk
Sum of additional points from dataset, training, tools, report >= 3pk
Sum of points >=10 (pair) 9 (alone)

5:
Problem >= 1pk
Model >= 3pk
Sum of additional points from dataset, training, tools, report >= 3pk
Sum of points >=14 (pair) 12 (alone)

6:
Problem >= 1pk
Model >= 3pk
Sum of points >=18 (pair) 15 (alone)

Milestones

Milestones

1. 11.01.2026 Filling columns team name, problem and dataset in https://docs.google.com/spreadsheets/d/10QhJAgm7pixE1truTpTNQf2vaNmaWRlh9gNj4jwFppU/edit?usp=sharing

1. 11.01.2026, the e-mail containing:
    - team name and members
    - selected problem and dataset
    - a brief description of what you are going to do
2. (26.01.2026 / 28.01.2026) **live presentation** (max 5 min) **+ report + link to git**

Failure to complete a milestone over time -0.5 grade.