Introduction to Artificial Intelligence

Summer 2025

Project: Ship Object Detection

Version: 1
Teacher: Muhammad Farhan, muhammad_farhan.safdar.dokt@pw.edu.pl

1. Project details

Develop a solution for detecting ships and boats from aerial images. Please use this dataset for this project. This is an object detection task.

Students may use any libraries for doing object detection. Several Deep Learning solutions such as PyTorch's Torchvision [1], TensorFlow's Object Detection [2], OpenMMLab's MMDetection [3] or any of the YOLO implementations [4]. In addition, students are encouraged to compare performance for different model architectures or different hyperparameter tunings.

The goal of this project are as follows:

- Develop an object detection model and evaluate its performance
- Understand data pre and post processing for object detection tasks
- Compare performance of different object detection architectures

2. References

1. https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html
2. https://www.tensorflow.org/hub/tutorials/object_detection
3. https://github.com/open-mmlab/mmdetection
4. Terven J, Córdova-Esparza D-M, Romero-González J-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. Machine Learning and Knowledge Extraction. 2023; 5(4):16801716. https://doi.org/10.3390/make5040083

3. Project Phases and Assessment

For each phase, there will be a report to submit followed by a discussion with the instructor. You may receive up to **25 points** for this project, assessed by the following criteria. Note the expected date (deadline) to complete each phase (earlier submissions are welcome).

Phase 1: Preliminary documentation (**0-5 points**)

Assessment	Deadline	Points
Preliminary report which contains: 1. Description of the dataset based on your observation. Provide descriptive statistics, plot some samples, report interesting patterns/findings. 2. Overview of your plan to tackle this problem. This includes: a. How to split the dataset into training and validation set b. What algorithms will be used, provide a brief description. c. What are the main tools/framework/libraries used for implementation. d. Proposed evaluation methods. How to measure and compare the performance of your algorithm. e. Note: You can use this notebook as a reference on how to develop a Deep Learning project pipeline	**09 May**	2 2
Preliminary discussion 3. Discuss with the instructor regarding your proposed solution. Obtain feedbacks.	**12 May**	1

Phase 2: Midterm solution (**0-5 points**)

Assessment	Deadline	Points
Submit the following files in GitHub: 1. Python code for the current state of implementation. 2. Updated report based on your findings. Including: a. Description of your implementation. b. Preliminary results from the algorithm based on proposed performance metrics. c. Experiment analysis. Based on results from your experiments so far, explain what worked and what didn't (what tweaks you made and its effect).	**26 May**	2 2
Mid discussion 3. Discuss regarding: a. Any changes from the initial plan b. Code demonstration c. Challenges or issues encountered so far	**27 May**	1

Phase 3: Final solution (**0-15 points**)

Assessment	Deadline	Points
Project submission. Submit the following files in GitHub: 1. Final Python code for implementation. 2. Final report. Please see final report guidelines below.	**09 June**	5 5

Final assessment 3. Code demonstration: a. Working training code (run for few epochs if it takes too long)	**13 June**	2
b. Evaluation code for a trained model 4. Analysis of the results 5. Conclusions: what you’ve learned from this project		2 1

4. Handing in guidelines

1. Use the GitHub platform to commit your progress and share the repository link instead of sending files directly (this includes the report). Don’t forget to write clear instructions on how to use the code.
2. All communications (messages, oral discussion) will be on MS Teams.
3. Please be aware of the deadlines for each project phase.
4. Final project submission (committing the final code and report on GitHub) is on **June 15th**. Late project submissions for this final phase will result in **20% points decrease** per week overdue (up to 2 weeks).

5. Final report guidelines

The following is a suggested template for the final report. However, students may add chapters and sub chapters as needed.

1. Problem definition
2. Dataset
    a. Overview (describe the dataset)
    b. Pre-processing (what is done to the images and labels before feeding them to the model)
    c. Post-processing (mention if used, any methods applied to output prediction)
3. Technical Approach:
    a. Architecture (what kind of model is used, describe briefly)
    b. Training details (describe the hardware and software used for this project. Describe the hyperparameters used for training, e.g.: loss function, learning rate, batch size, etc.)
    c. Evaluation details (what performance metrics are used, e.g.: accuracy, precision, recall)
4. Results
    a. Provide a table comparing different approaches and their obtained performance metrics. This is a quantitative analysis.
    b. Provide prediction results on some samples, some from correctly classified and some from incorrect ones. Describe briefly based on your experiments, why the model made these correct/incorrect predictions.
5. Conclusion
6. References
    a. If you rely on algorithms obtained from works of other people, please cite the author and their work (paper, git repo, blog).
    b. Please try your best to cite quotes, facts, images used in your report that you did not create yourself. This will make your work more credible.

6. Parting wisdom

As Francois Chollet (creator of Keras) says, machine learning is an iterative process. Results might not show immediately, so be patient and test out your ideas by making minor adjustments based on your results. Software bugs are real, so check your processing pipeline if there's any unintended variable changes or incorrect use of functions. Finally, and most importantly, don't forget to **have fun while learning**.