June-July Exam session Projects List

Objectives

The goal of the project is to tackle one of the proposed topics in the field of Computer Vision, developing non-trivial solutions. Students should explore the problem from an original perspective, applying methods that go beyond conventional solutions and demonstrating critical thinking and problem-solving skills.

Lines of conduct

- **Student groups:** The project can be carried out by a group, each consisting of a maximum of 3 people. Projects can also be completed by individual students, but we suggest to work in team.
- **Notebook format:** The project must be implemented using a notebook (e.g., Google Colab, Kaggle) or with an IDE (e.g., VSCode, PyCharm). The code should be optimized to support GPU usage and run without any error. The delivered code must follow the structure outlined below:
  - *Imports*: all the needed packages (for the notebook format)
  - *Globals*: useful variables on the whole code
  - *Utils*: code support functions
  - *Data*: everything related to data management
  - *Network*: code to structure the neural network
  - *Train*: part containing the training cycle elements
  - *Evaluation*: tests needed for the trained network

  You can find a sample template at this link. Try to maintain as much as possible this conceptual structure.
- **Deep learning framework:** All projects **MUST** be done in Python via the Pytorch framework.
- **Project assignment:** You are required to choose a project through this Google Form. In this form, you will provide information about your team and the project, and include the link to the project's GitHub repository. In this repository you have to upload:
  - Code (or notebook) implementing the project
  - Dataset (or a link to it)
  - Project presentation
  - Detailed README to provide a quick overview of the project and instructions on how to run it
- **Project submission:** The project must be presented on one of the exam dates. It can be presented at a different time than the written exam. Both the written exam and the project MUST be completed within the academic year (i.e., between the June 2025 session and the March 2026 session).

  October and March session are reserved to "categories of students referred to in Article 40, paragraph 6, of the General Study Manifesto, and out-of-school students enrolled for the A.Y. 2024-2025 in the third year of a Bachelor's degree and in the second year of a Master's degree".
- **Plagiarism:** Any attempt to plagiarize, whether by copying other students' work, directly replicating code from online resources, or submitting content highly retrived from generative AI models, will be strictly penalized. This course values originality and personal effort; therefore, students must submit independently developed solutions. On the other hand, it is acceptable to consult external resources for inspiration or guidance.

Project 5: Lightweight Convolutional Occupancy Networks for Efficient Virtual Scene Generation

Abstract: The generation of virtual scenes is a critical challenge in robotics, augmented reality, and simulation, where accurate and computationally efficient 3D representations are essential. However, existing solutions often involve a trade-off between reconstruction quality and execution speed, limiting their applicability in real-time scenarios. This project investigates using neural networks to generate 3D scenes, focusing on optimizing the model to reduce inference time while maintaining reasonable reconstruction fidelity. Various optimization strategies will be explored to achieve this, assessing their impact on efficiency and visual quality. The study will provide a comparative analysis against existing approaches, contributing to developing lightweight and effective models for virtual scene synthesis.

Dataset: Synthetic-Rooms (provided)

Task: This project explores using a Lightweight Convolutional Occupancy Network to generate virtual scenes composed of objects from the ShapeNet dataset. The primary objective is to optimize the network to achieve low inference time while preserving the quality of the reconstructed scene. Students will refine the network architecture, adjusting key components to improve computational efficiency without compromising accuracy.

Main objectives:

- *Training and Fine-Tuning*: Train the Lightweight Convolutional Occupancy Network on the Synthetic-Rooms dataset, ensuring accurate reconstruction of structured scenes. Fine-tune network parameters and apply geometrical modifications to optimize the balance between speed and quality.
- *Model Optimization*: Minimize inference time while preserving reconstruction quality by refining the architecture and applying optimization techniques such as model pruning, quantization, and geometrical simplifications.
- *Evaluation*: Compare the optimized model against state-of-the-art approaches using key metrics, including inference time, reconstruction accuracy, and visual fidelity, to assess trade-offs introduced by optimizations.
- *Trade-off analysis*: Analyze the impact of different optimization strategies on computational efficiency and reconstruction accuracy, evaluating whether the optimized model can outperform existing solutions in real-time scene generation.

References:

1 Lionar, S., Emtsev, D., Svilarkovic, D., and Peng, S. (2020). Dynamic Plane Convolutional Occupancy Networks. arXiv.

2 C. M. Tonti, L. Papa and I. Amerini, "Lightweight 3-D Convolutional Occupancy Networks for Virtual Object Reconstruction," in IEEE Computer Graphics and Applications, vol. 44, no. 2, pp. 23-36, March-April 2024, doi: 10.1109/MCG.2024.3359822.

3 Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., and Geiger, A. (2020). Convolutional Occupancy Networks. arXiv.