

COMP52715 Deep Learning for Computer Vision & Robotics
(Epiphany Term, 2023-24)

Summative Coursework - 3D PacMan

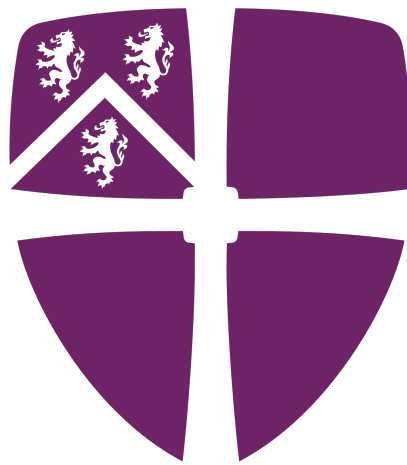
Coursework Credit - 15 Credits

Estimated Hours of Work - 48 Hours

Submission Method - via Ultra

Release On: February 16 2024 (2pm UK Time)

Due On: March 15 2024 (2pm UK Time)



Durham
University

1 Coursework Specification

1. This coursework constitutes 90% of your final mark for this module, where there are two mandatory tasks: Python programming and report writing. You must upload your work to Ultra before the deadline specified on the cover page.
2. The other 10% will be assessed separately based on seminar participation. There are 3 seminar sessions in total, the mark awarding rule is as such: (A) participating in none=0%, (B) participating in 1 session=2%, (C) participating in 2 sessions=5%, (D) participating in all sessions=10%.
3. This coursework is to be completed by students working individually. You should **NOT** ask for help from your peers, lecturer, and lab tutors regarding the coursework. You will be assessed on your code and report submissions. You must comply with the University rules regarding plagiarism and collusion. Using external code without proper referencing is also considered as breaching academic integrity.
4. **Code Submission:** The code must be written in Jupyter Notebook with appropriate comments. For constructing deep neural network models, use PyTorch¹ library only. Zip Jupyter Notebook source files (*.ipynb), your dataset (if there is any new), pretrained models (*.pth), and a README.txt (code instruction) into one single archive. Do **NOT** include the original “*PacMan_Helper.py*”, “*PacMan_Helper_Demo.ipynb*”, “*PacMan_Skeleton.ipynb*”, “*TrainingImages.zip*”, “*cloudPositions.npy*” and “*cloudColors.npy*” files. **Submit a single Zip file to GradeScope - Code entry on Ultra.**
5. **Report Submission:** The report must NOT exceed 5 pages (including figures, tables, references and supplementary materials) with a single column format. The minimum font size is 11pt (use Arial, Calibri, Times New Roman only). **Submit a single PDF file to GradeScope - Report entry on Ultra.**
6. Academic Misconduct is a major offence which will be dealt with in accordance with the University’s General Regulation IV – Discipline. Please ensure you have read and understood the University’s regulations on plagiarism and other assessment irregularities as noted in the Learning and Teaching Handbook: 6.2.4: Academic Misconduct².



Figure 1: The mysterious PhD Lab.

¹<https://pytorch.org/>

²<https://durhamuniversity.sharepoint.com/teams/LTH/SitePages/6.2.4.aspx>

2 Task Description (90% in total)

2.1 Task 1 - Python Programming (40% subtotal)

In this coursework, you are given a set of 3D point-clouds with appearance features (i.e. RGB values). These point-clouds were collected using a Kinect system in a mysterious PhD Lab (see Figure.1). Several virtual objects are also positioned among those point clouds. Your task is to write a Python program that can automatically detect those objects from an image and use them as anchors to collect the objects and navigate through the 3D scene. If you land close enough to the object it will be automatically captured and removed from the scene. A set of example images that contain those virtual objects are provided. These example images are used to train a classifier (basic solution) and an object detector (advanced solution) using deep learning approaches in order to locate the targets. You are required to **attempt both basic and advance solutions**. “*PacMan_Helper.py*” provides some basic functions to help you complete the task. “*PacMan_Helper_Demo.ipynb*” demonstrates how to use these functions to obtain a 2D image by projecting 3D point-clouds onto the camera image-plane, and how to re-position and rotate the camera etc. All the code and data are available on Ultra. You are encouraged to **read the given source codes**, particularly “*PacMan_Skeleton.ipynb*”.

Detection Solution using Basic Binary Classifier (10%). Implement a deep neural network model that can classify the image patch into two categories: target object and background. You can use the given images to train your neural network. It then can be used in a sliding window fashion to detect the target object in a given image.

Detection Solution using Advance Object Detector (10%). Implement a deep neural network model that can detect the target object from the image. You may manually or automatically create your own dataset for training the detector. The detector will predict bounding boxes that contain the object from a given image.

Navigation and Collection Task Completion (10%). There are 11 target objects in the scene. Use the trained models to perform scene navigation and object collection. If you land close enough to the object it will be automatically captured and removed from the scene. You may compare the performance of both models.

Visualisation, Coding Style, and Readability (10%). Visualise the data and your experimental results wherever is appropriate. The code should be well structured with sufficient comments for the essential parts to make the implementation of your experiments easy to read and understand. Check the “Google Python Style Guide”³ for guidance.

2.2 Task 2 - Report Writing (50% subtotal)

You will also write a report (**maximum five pages**) on your work, which you will submit to Ultra alongside your code. The report must **contain the following structure**:

Introduction and Method (10%). Introduce the task and contextualise the given problem. Make sure to include a few references to previously published work in the field, where you should demonstrate an awareness of the relevant research works. Describe the model(s) and approaches you used to undertake the task. Any decisions on hyper-parameters must be stated here, including motivation for your choices where applicable. If the basis of your decision is experimentation with a number of parameters, then state this.

Result and Discussion(10%). Describe, compare and contrast the results you obtained on your model(s). Any relationships in the data should be outlined and pointed out here. Only the most important conclusions should be mentioned in the text. By using tables and figures to support the section, you can avoid describing the results fully. Describe the outcome of the experiment and the conclusion that you can draw from these results.

Robot Design (20%). Consider designing an autonomous robot to undertake the given task in the real scene. Discuss the foreseen challenges and propose your design, including robot mechanic configuration, hardware and algorithms for robot sensing and controlling, and system efficiency etc. Provide appropriate justifications for your design choices with evidence from existing literature. You may use simulators such as “CoppeliaSim Edu” or “Gazebo” for visualising your design.

³<https://google.github.io/styleguide/pyguide.html>

Format, Writing Style, and Presentation (10%). Language usage and report format should be in a professional standard and meet the academic writing criteria, with the explanation appropriately divided as per the structure described above. Tables, figures, and references should be included and cited where appropriate. A guide of citation style can be found at library guide⁴.

3 Learning Outcome

The following materials from lectures and lab practicals are closely relevant to this task:

1. Basic Deep Neural Networks - Image Classification.
2. Generic Visual Perception - Object Detection.
3. Deep Learning for Robotics Sensing and Controlling - Consideration for Robotic System Design.

The following key learning outcomes are assessed:

1. A critical understanding of the contemporary deep machine learning topics presented, and how these are applicable to relevant industrial problems and have future potential for emerging needs in both a research and industrial setting.
2. An advanced knowledge of the principles and practice of analysing relevant robotics and computer vision deep machine learning based algorithms for problem suitability.
3. Written communication, problem solving and analysis, computational thinking, and advanced programming skills.

The rubric and feedback sheet are attached at the end of this document.

⁴https://libguides.durham.ac.uk/research_skills/managing_info/plagiarism