

Homework 1

Image Classification

TA. 單宇晟 曹博鈞 吳年茵 曾偉杰





Introduction

- For this assignment, you are required to build a neural network with **PyTorch** and train it to carry out a **classification task**.
- To ensure that students can meet the assignment's requirements, **the use of pretrained weights and existing models is not permitted.**



What to do?



- Please modify the provided `.ipynb` notebook to complete your assignment. **A Colab notebook** with reference implementations is available for your reference.
- You can import any package and design any additional classes or functions if you need. However, the utilization of ready-made neural network and pre-trained weight is forbidden.
- **Discovery of any infringement of this cardinal rule will incur a penalty of a zero score for this assignment.**



Dataset

1. The dataset consists of sports images from **100 different categories**. The size of each image is **224*224*3**.
2. A total of 10000+ images are provided for training.
3. link:

[https://drive.google.com/drive/folders/1siSBbhgRAaKIFR-6yRgnjaUXS6W0XDKQ?usp=drive link](https://drive.google.com/drive/folders/1siSBbhgRAaKIFR-6yRgnjaUXS6W0XDKQ?usp=drive_link)



Grading (Performance 90% + Report 10%)

- **Top-5 Accuracy (75 points)**

65% ≤ Accuracy, get full points

60% ≤ Accuracy < 65%, get 65 points

55% ≤ Accuracy < 60%, get 55 points

50% ≤ Accuracy < 55%, get 45 points

Accuracy < 50%, get 0 points

- **Number of Parameters (15 points)**

This evaluation is based on the number of parameter of your model. The fewer parameters you use, the higher score you will get.

Score formula: $\text{Round} \left(\frac{n - r}{n - 1} \times 15 \right)$

n: number of students

r: your ranking (r = 1 being the best rank)



Report

- In the report, briefly describe your approach.
- e.g. the design of the model architecture, strategies for balancing model performance with computational and parameter efficiency, and the development of an effective training strategy

Grading

- Please generate "pred_{student_id}.csv" based on the weight(.pth) you provided. (We already provide sample code in Colab). Make sure your output format match the format we want.
- You can use the given code cell to evaluate your model parameters.

```
model = ClassificationModel()  
total_params = sum(p.numel() for p in model.parameters())  
print("# parameters:", total_params)
```

```
# parameters: 10244
```



Grading

- Your `pred_{student_id}.csv` should look like.

```
file_name,pred1,pred2,pred3,pred4,pred5
001.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
002.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
003.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
004.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
005.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
006.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
007.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
008.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
009.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
010.jpg,air hockey,ampute football,archery,arm wrestling,axe throwing
```



Submission

- Your submission should contain:
 - Code: `{student_id}.ipynb`
 - Model Weight: `w_{student_id}.pth`
 - Your Prediction: `pred_{student_id}.csv`
 - Report : `{student_id}.pdf`

```
hw1_{student_id}.zip
├ {student_id}.ipynb
├ w_{student_id}.pth
├ pred_{student_id}.csv
└ {student_id}.pdf
```

- Compress all files into a single zip file named `hw1_{student_id}.zip` without any extra folder layers.
- Make sure not to include unnecessary files or folders (e.g., `.DS_Store`, `__MACOSX`).



Penalty

- Format penalty - **20 points**
 - If you have any incorrect file format or name, then you will get **-20 points**.
- Late penalty - **10% per day**
- **Deadline: 2025/10/5 23:59**



Note

- If you have any questions, please contact us via the E3 system and include **ALL** TAs in your message

