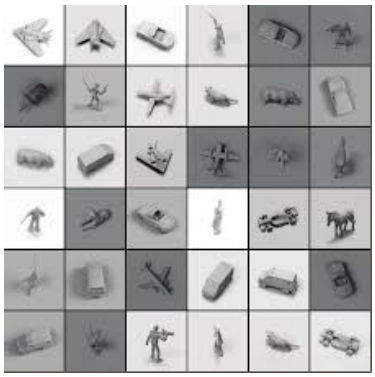
COSC420 Assignment 1

Training neural networks to classify/generate toy images

Weight:20% Lecturer: Lech Szymanski

For this assignment, you will be building and training neural network models using Tensorflow's Keras library.

Dataset



The dataset for your training and testing is the smallNORB dataset. It contains 96x96 pixel grey-scale images of 50 toys classified into 5 categories. The toys where photographed by two cameras, under varying lighting conditions and from different angles. The dataset is provided through Tensorflow API.

Task 1, Classification (5 marks)

Build and train a convolutional neural network (CNN) in Tensorflow to:

- classify whether a given smallNORB image is a four-legged animal, human figure, airplane, truck, or a car;
- predict elevation of the camera from the image (as a regression problem).

You can do this by creating a single network where part of the outputs is dedicated to classification and part to regression (yes, it is possible), or two networks (in separate scripts) that do classification and regression independently. It is up to you how to format the input data (whether to use one image or two, whether to normalise, do augmentation, etc). The data is already split into 24300 samples (of two images each) training and 24300

samples (of two images each) testing datasets for you. It's up to you whether to split the training data into proper training and validation sets, and whether to implement early stopping. It's up to you how to specify the architecture of the network, how to train it and so on (though it needs to be a CNN, as CNNs do better on images than MLFFs). There is no expected accuracy target, as in, I don't have a specific number in terms of performance in mind. This might vary depending on how you train and test your network. Just try to produce a model with best generalisation you can achieve, while being methodical about different choices of hyper-parameters and comparing (perhaps after a fixed/smaller number of epochs to speed up the hyper-parameter testing process) the performance for the purpose of finding the best combination. Starting with the template code provided in the tfintro repository as your code base is perfectly fine. Going with transform learning approach is ok too.

Task 2, Stable diffusion (5 marks)

For this task, you are to build a *diffusion model* and train it to generate images from noise. You should use the smallNORB images for this, though if you are super keen to use a different (say, more colourful) training dataset, please check with the lecturer first (to get his approval). This task is somewhat more open that the previous one, in that it's less specific how you should go about this task, and there is no given template to follow – just try to generate images through the diffusion process and see where you get. There is no specific target that needs to be reached – just do your best to make it work as best as you can.

Task 3, Report (10 marks)

Write a report of what you have done for Task 1 and Task 2. What I am looking for here is a justification for choices made in your implementation, methodical approach to your investigation, summary of the results and analysis. For Task 1, if you try different architectures/regularisations/optimisers etc., record the results, report what happened and provide some justification for why you think it did (or did not) work. Including diagrams and figures, such as plots of training and validation accuracy over the course of training, is a good way to give me more insight into what you have done, and to back up any decisions made about hyper-parameters, strategies, etc. For Task 2, explain how your implemented diffusion model works and is trained. As results, you should include (at least) some examples of images your network has created. This is meant to be a technical report – concise, but clear.

Submission

The assignment is due at **4pm on Monday of 17th April**. For each task I expect submission of:

- Task1: python code for training your final model(s) as well as saved files with of trained model(s) weights – ideally, there should be a flag (like in the tfintro examples), which allows building and training the model from scratch or loading a pre-saved one (that needs to be include in your submission). To help you with loading the data, the script for loading smallNORB into Python is provided (see Blackboard, Assignment 1 entry).
- Task2: python code for training the diffusion model as well as saved files with the trained model(s) weights; the script, again, should have a flag (like in the tfintro examples), which allows building and training the model from scratch or loading pre-saved (submitted) one. Ideally the dataset for training should be downloaded automatically the first time the code is run - like the smallNORB dataset in the provided helper file (I don't want submissions with gigabytes of data containing train images).
- Task3: a pdf file with your report, around 1500 words (rough guid, more is ok, less might be ok, but it might be lacking the required depth of analysis and results reporting).

Zip the folder including your code, saved model files and report and submit electronically via Blackboard.

Don't forget to clean up your code before submission and to add comments. Make sure to save your models after training and to include them (but not the training data) in the submission.

Academic Integrity and Academic Misconduct

Academic integrity means being honest in your studying and assessments. It is the basis for ethical decision-making and behaviour in an academic context. Academic integrity is informed by the values of honesty, trust, responsibility, fairness, respect and courage. Students are expected to be aware of, and act in accordance with, the University's Academic Integrity Policy.

Academic Misconduct, such as plagiarism or cheating, is a breach of Academic Integrity and is taken very seriously by the University. Types of misconduct include plagiarism, copying, unauthorised collaboration, taking unauthorised material into a test or exam, impersonation, and assisting someone else's misconduct. A more extensive list of the types of academic misconduct and associated processes and penalties is available in the University's Student Academic Misconduct Procedures.

It is your responsibility to be aware of and use acceptable academic practices when completing your assessments. To access the information in the Academic Integrity Policy and learn more, please visit the University's Academic Integrity website or ask at the Student Learning Centre or Library. If you have any questions, ask your lecturer.

- Academic Integrity Policy
- Student Academic Misconduct Procedures

Use of ChatGPT as a tool to help with the writing of the report is allowed, but needs to be reported – how did you use it and what for (just polishing the grammar, providing such and such explanations, etc). Apparently, you can't cite ChatGPT, so just include an appendix (in your report) explaining how ChatGPT was used.