Istanbul Technical University - Computer Engineering
BLG 561E - Deep Learning
Homework 1
Prepared by: Dr. Yusuf H. Şahin
E-mail: sahinyu@itu.edu.tr
Deadline: 07.04.2025

Question 1 (30 pts) - The Turkish Academy Voice Challenge

Figure 1: Five famous Turkish academicians

For this homework, you will use machine learning to distinguish voices of prestigious Turkish academics & researchers. The audio files for this challenge include Aziz Sancar, Biykem Bozkurt, Cahit Arf, Canan Dagdeviren and Koray Kavukcuoglu. You can find the train files under this link. Each audio file has the same length of nearly 5 seconds.

Create your own neural network to classify the audio files. You may use mel-spectrograms to prepare your input. Submit your test results to kaggle using the following link.

Question 2 (30 pts) - The Mine Game

I prepared a computer game in which the main character is trying to reach the end of the map given in Figure 2. You can find game via this link.

By default the character can be controlled using arrow keys and shift key makes him run faster. You should also use pyautogui to take screenshots of the board, decide on the next action and simulate these actions.

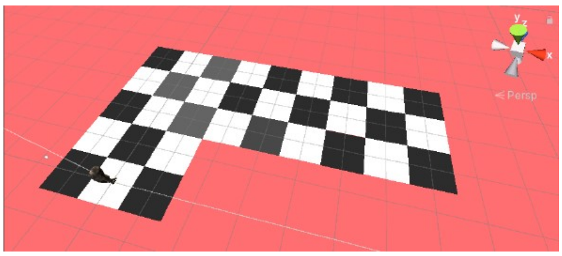

Figure 2: The map of the maze game

The problem with these grids is, some of them are mined. However, if our character is close to a mined grid (according to grid borders), a face image having a "shocked" expression appears on the bottom right as given in Figure 3.

Here, the following strategies should be followed:

- Using features from a pretrained convolutional neural network, detect the difference between "shocked" face and "normal" face. Try to not enter mined grids.
- You can keep a list of visited grids so that, the character should not tend to visit some grids over and over. Thus, you should benefit from corner and edge detection to detect whether the character is entered to another grid.
- Make the character reach the last grids safely.

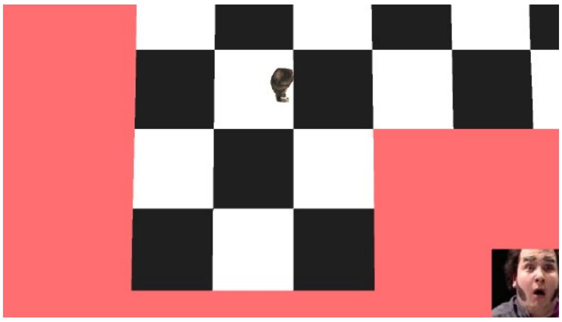

Figure 3: The character is very close to a mined grid.

textbfpyautogui, is used to simulate mouse and keyboard interactions with Python. Also you can take a screenshot of the game. The example script given below clicks random keyboard buttons.

```
import pyautogui

pyautogui.keyDown('shift')
pyautogui.keyDown('w')
time.sleep(1)
pyautogui.keyUp('w')
pyautogui.keyUp('shift')

#pyautogui.keyUp and pyautogui.keyDown functions are used to simulate
holding a button.
#For simple presses, pyautogui.press can be used.
```

Question 3 (40 pts) - Evolution

In the article "Modular Controllers Facilitate the Co-Optimization of Morphology and Control in Soft Robots" Mertan and Cheney designed an evolutionary algorithm to find the soft robots which can walk without a fall on a straight line.

They explored how modular controllers can improve the co-optimization process of soft robots, which involves designing both their body structure and control strategies simultaneously. A key challenge in this process is the tight coupling between a robot's controller (its "brain") and its morphology (its "body"), which can lead to premature convergence in evolutionary optimization.

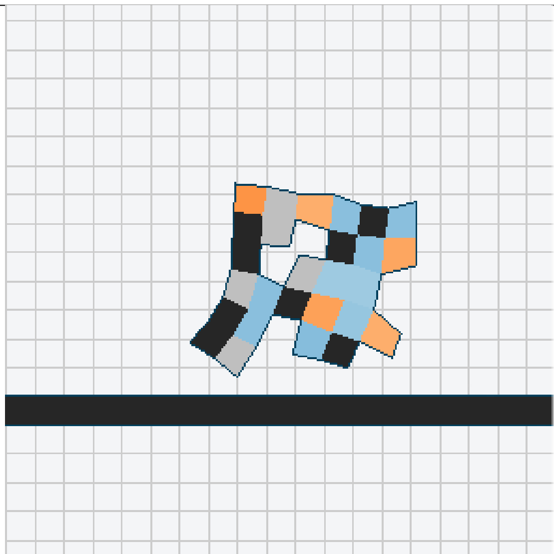

Figure 4: A successfully walking soft-robot

The authors argue that modular controllers, as opposed to global controllers, enhance the robustness of soft robots by making them less sensitive to morphological changes. Their experiments show that robots with modular controllers experience less performance loss after morphological mutations, allowing for a more effective evolutionary search that results in fitter offspring. By increasing the transferability of control strategies across different body plans, modular controllers facilitate a smoother optimization process and lead to better-performing robots over generations.

In this context, the "brain" refers to the control system of the robot, which governs its movements and behaviors. In this study, the brain is represented by controllers—either global or modular—that dictate how the robot's actuators function based on sensory inputs. The "body" represents the robot's morphology, which consists of soft, compliant materials arranged in a specific structure. The body interacts with the environment and determines the robot's locomotion capabilities. The study emphasizes that successful co-optimization requires a balance between the adaptability of the brain and the flexibility of the body, and it demonstrates that modular controllers help maintain this balance by improving robustness to morphological mutations.

Here, a small neural network is designed to act as a "body" of each soft robot. Instead of training by SGD algorithm, random changes like mutations and cross-overs are done on the weights of the network. Using the natural survival of the fittest strategy, after lots of generations, the robots which can walk are obtained.

Change the network structure! Try to obtain a positive score greater than 0.6!