**Report**

**I have used the stablebaselines3 library.**

**Preprocessing the observation**

* **I convert the observation from a (200,256,3) tebsor to a (84,84,1) tensor by first using opencv to convert it to (84,84) by grayscaling and pooling and then using numpy to make this a (84,84,1) tensor.**

**Editing the Reward function and observation**

* **I changed the reward function to now represent the change in the score in the current timestep as the previous reward function was too sparse.**
* **I cuanged the observation to record the change in the pixel values of the current and previous frames using numpy**

**Fine tuning hyperparameters**

* **I used optuna to create a study which identifies the optimal hyperparameters for a model to learn best.**
* **I ran it for 30k steps for each trial and moved forward with the hyperparameters and weights of the best trial(the one which gave the most mean reward)**
* **The model also looked back atnits previous 10 observations**

**Training**

* **I trained the PPO model with the optimal hyperparameters.**
* **I trained for 100k steps and monitored progress on tensorboard.**

**Evaluating model**

* **I evaluated the model by using the evaluate policy function on sb3.**