

# Little Fighter 2

## Introduction

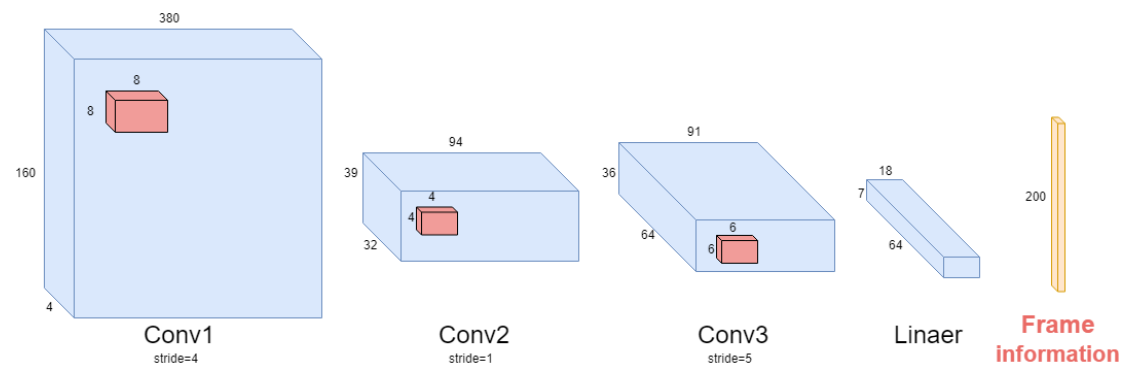
- Environment
  - [lf2gym](#)
  - Pytorch 0.4
  - Python 3.6.5
  - Ubuntu 16.04
  - Setup
    - ◆ Action list: [上下左右, 攻防跳, 角色技能]
    - ◆ Observation state: frame stack plus character information
    - ◆ Training step limitation: 150 steps
    - ◆ Opponent difficulty: [Dumbass, Challenger, Crusher] (Crusher selected)
- Objective
  - Try to survive
  - Beat the enemy
    - ◆ Use normal attack
    - ◆ Use combo skills
- Method
  - DQN
    - ◆ Batch size: 32
    - ◆ Epsilon greedy: 0.9
    - ◆ Memory capacity: 2000

- MDP

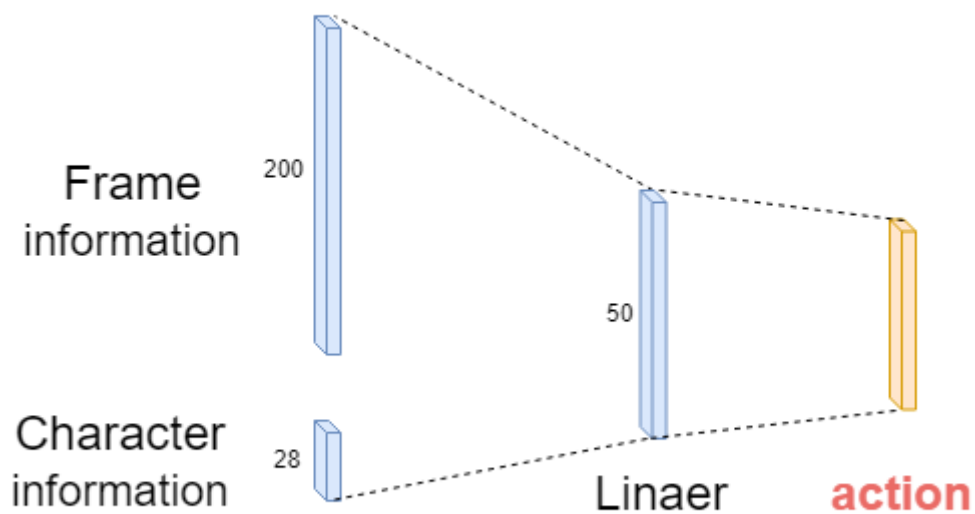
- 每一次 step 的 observation 是抓取整個遊戲畫面的 frame 加上兩個角色的 information，包括角色位置、當前角色的面向、角色的 HP、角色的 MP 等。
- Action 為鍵盤上的按鍵，例如：上、下。
- 每一個 action，會拉長成四個 step 去完成，若是普通的 action，例：按[上]，就會拉長成四個一樣的 action，變為[上,上,上,上]；若為組合鍵招式，則會拉長為該招式所需之組合按鍵，例：[防,右,攻,0]，這個 0 代表為 idle，此 action 不做任何事。此四個 step 會形成個 stack 作為 observation。
- HP: 自己的 HP 係數設為 -1，敵人的 HP 係數設為 +1
- MP: 自己的 MP 係數設為 -1，敵人的 MP 係數設為 0
- Reward
  - ◆ 每一次 step 的 Reward 都會加上前一次的 Reward。
  - ◆ 每一次 step，所有人都將自己前一次 step 的 HP 扣掉這次 step 自己的 HP。如果是我方 HP 減少，即為被攻擊，相減下來為正數，但乘上自己的係數後即為負數，Reward 會越加越小；反之若敵人 HP 減少，相減下來為負數，Reward 會越加越大。
  - ◆ MP 為魔力值，因為我們不在乎敵人的魔力值(有可能是敵人亂放招)，所以將敵人的 MP 係數設為 0。相對的我們也要避免我們的腳色亂放招，因此我們也將自己的 MP 係數設為 -1，避免腳色一味地對空氣放招。
  - ◆ 最後將 HP 跟 MP 的重要比例設為 10 : 1，因為這遊戲主要目的是將敵人 HP 歸零，因此 HP 重要程度遠大於 MP。

# Model

- Picture based: 將  $[4, 160, 380]$  的 frame stack 放進 cnn 後直接經過 Linear 層得到 action value
- Feature based: 將 28 維的 character feature 直接經過 Linear 層
- Mixed: 結合上述兩者，架構圖如下



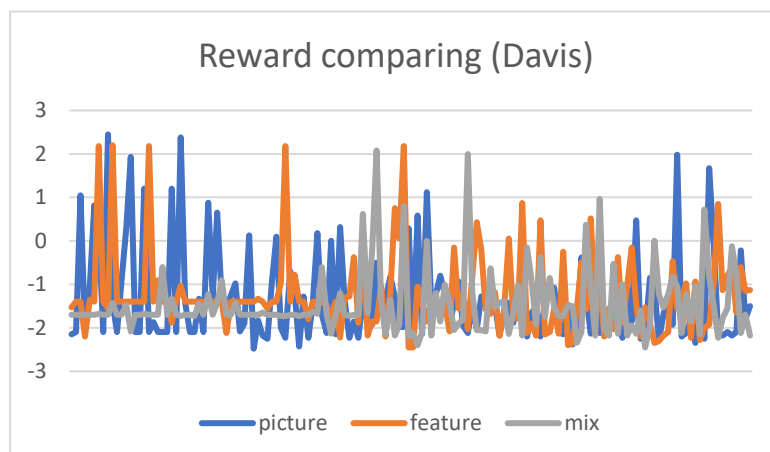
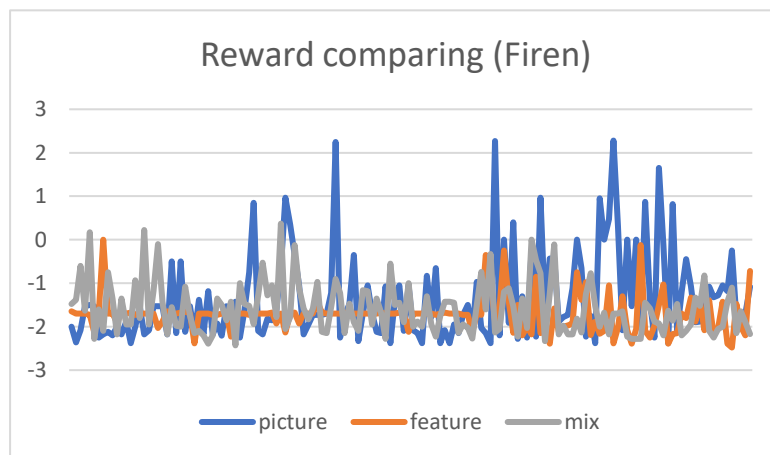
1. 先將 frame stack 轉為 200 維的 feature



2. 再與 28 維的 character information 結合後取得 action

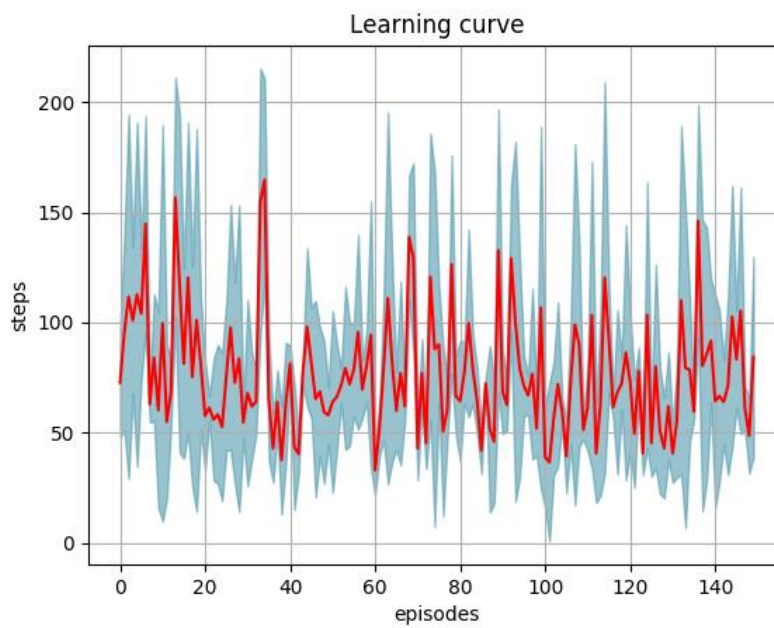
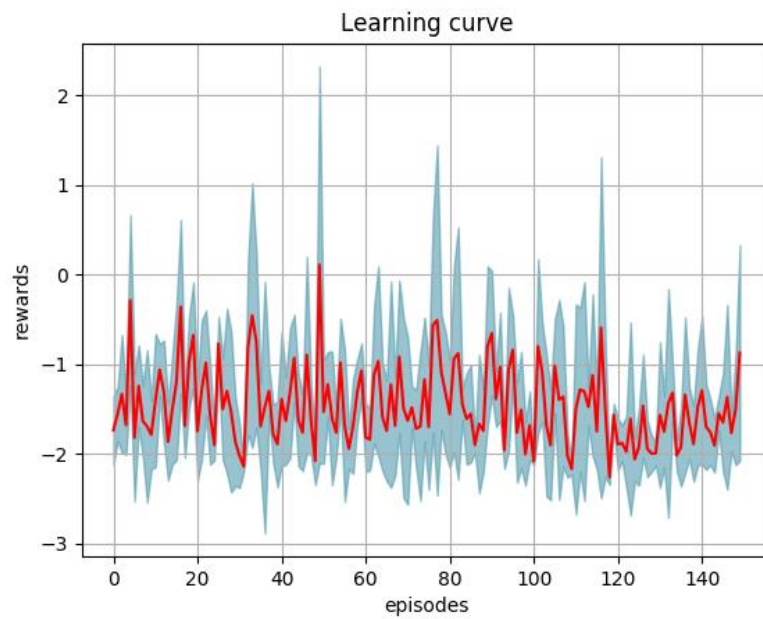
# Analysis

- Use different types of mode
  - Frame stack: Only use a two dimensional picture
    - ◆ Davis with picture mode
    - ◆ Firen with picture mode
  - Feature: Only use two characters' information (One dimensional)
    - ◆ Davis with feature mode
    - ◆ Firen with feature mode
  - Mix: Mix above two types
    - ◆ Davis with mix mode
    - ◆ Firen with mix mode
- Result



- **feature mode** 是有抓到位置的趨勢，因此我們將兩者結合做出 **mix** 版。
- 上兩張圖也可以看出不同角色用同樣的訓練方式有所差異。

- Learning Curve



- 根據上兩張圖，因為這遊戲屬於兩人對戰遊戲，整體環境變因太複雜，因此還無法得到一個較好結果。

# Review

- Attempt
  - 因為我們之前做 training 時，episode 上限都是設為 200 次，我們想試試看是不是次數不夠造成效果沒有到很好，因為之前有找過網路上有人嘗試了 3000 episodes 甚至 7000 episodes，因此我們嘗試將 episode 設為 2000。但 training 過程中經過 12 小時只 train 了 500 多個 episode，而且發現當前效果與 episode = 200 的結果是差不多的。
- Discovery
  - 從結果影片可發現角色會趨於一直選擇同一個動作，推斷應該是 Reward function 定義或使用的 RL Algorithm 造成的問題。
  - 從我們的 Agent 與 Opponent 互動過程中發現，Opponent 在環境當中對位置的深度判斷有些許的錯誤，因此判斷 lf2gym 在位置的資訊上有可能存在某些錯誤。
- Improvement
  - 探討 lf2gym 中的 feature 意義以此重新定義 reward function。
  - 尋找是否有解決重複動作的方法。

# Team works

- Github: <https://github.com/HaoTse/lf2-rl>
- Modified environment: <https://github.com/HaoTse/lf2gym>
- Member

	鄭皓澤	徐銘宏
Email	top30339@gmail.com	denlee0125@gmail.com
Contribution	1. Observation process 2. Model 3. Analysis	1. Environment 2. Character feature 3. Report