

Computer vision in deep learning implementation and its applications

DeepDream+StyleTransfer

接下來的課程安排

周次	教學主題	單元名稱	課程目標	內容簡介
15(12/19)	強化學習	強化學習介紹	瞭解強化學習之基礎理論及應用方法。	<ol style="list-style-type: none">強化學習基本原理Deep Q-Learning/Policy gradients 介紹強化學習模型實作
15(12/19)	期末專題介紹	專案案例背景知識介紹	瞭解案例之背景知識及待解決問題。	<ol style="list-style-type: none">形成演算法架構演算法開發整合演算法驗證測試
16(12/26)			期末專題實作	
17(1/2)	深度學習	Attention與Vision Transformers	瞭解Attention機制及ViT核心理論	<ol style="list-style-type: none">RNN介紹Attention/ViT介紹ViT模型實作
18(1/9)	期末專題作業	專案部署及作業繳交	習得專案上線部署技巧及系統成效分析檢討能力。	<ol style="list-style-type: none">部署專案成果簡報檔(含報告錄音檔)

Deep Dream

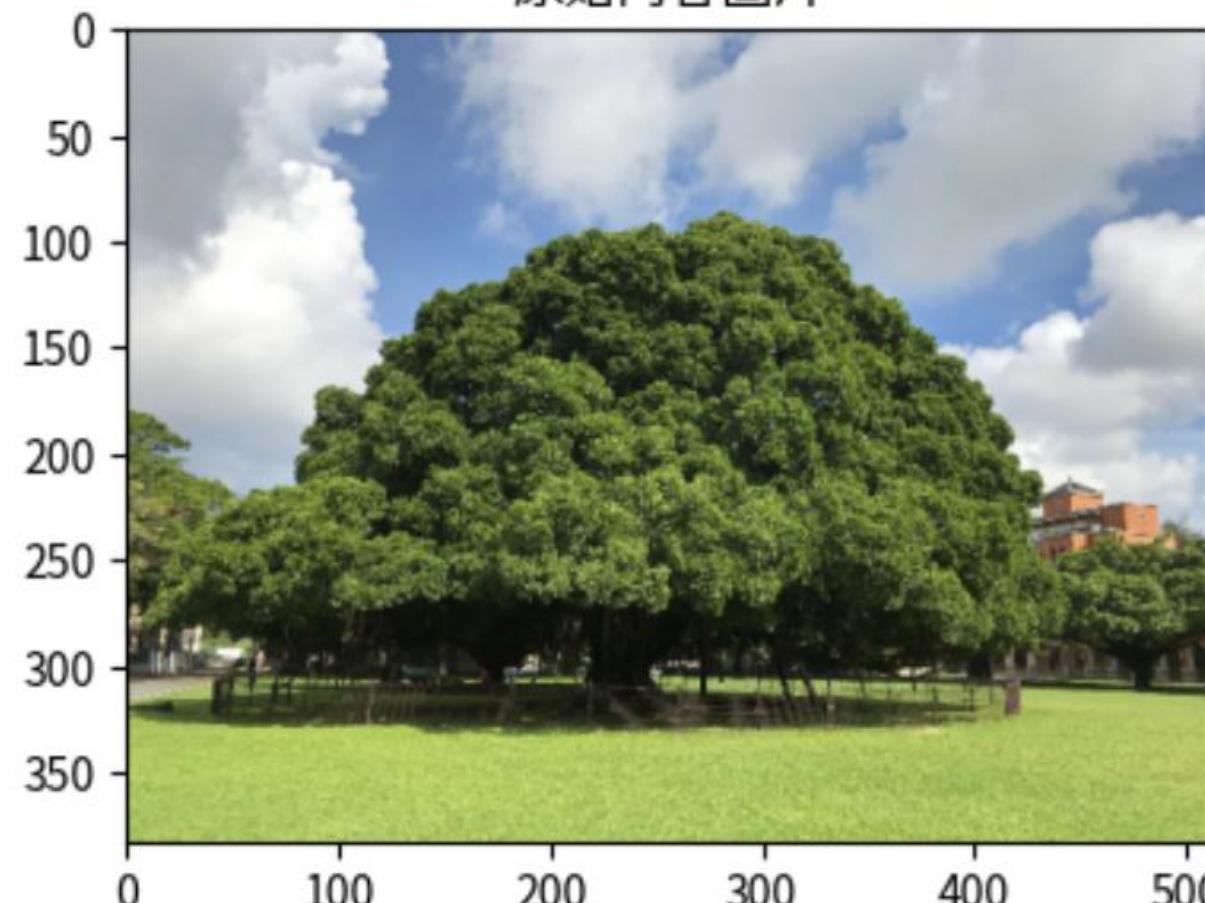
https://colab.research.google.com/drive/1d_09wsI660dBUjWeebD1wINTER7LZJr?usp=sharing



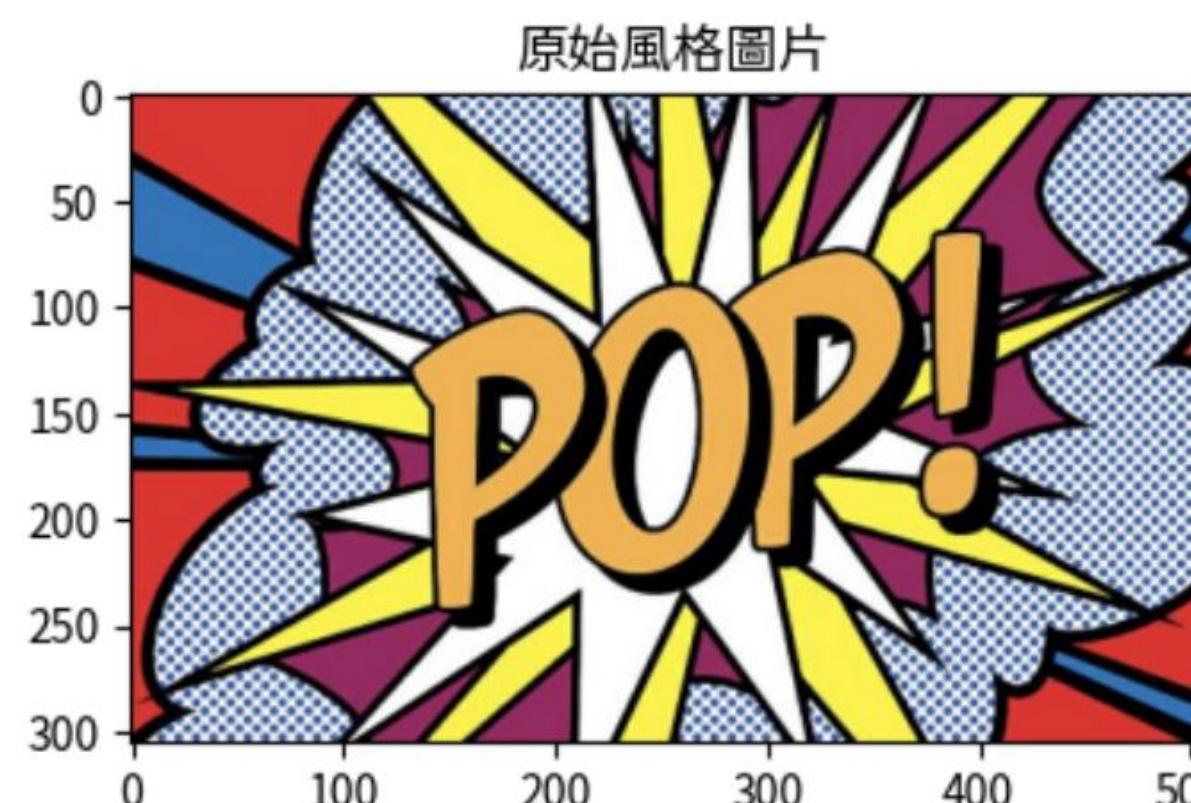
Style Transfer

<https://colab.research.google.com/drive/1777GHCoQIV7w67QmkNXLHdGN4kvHqln?usp=sharing>

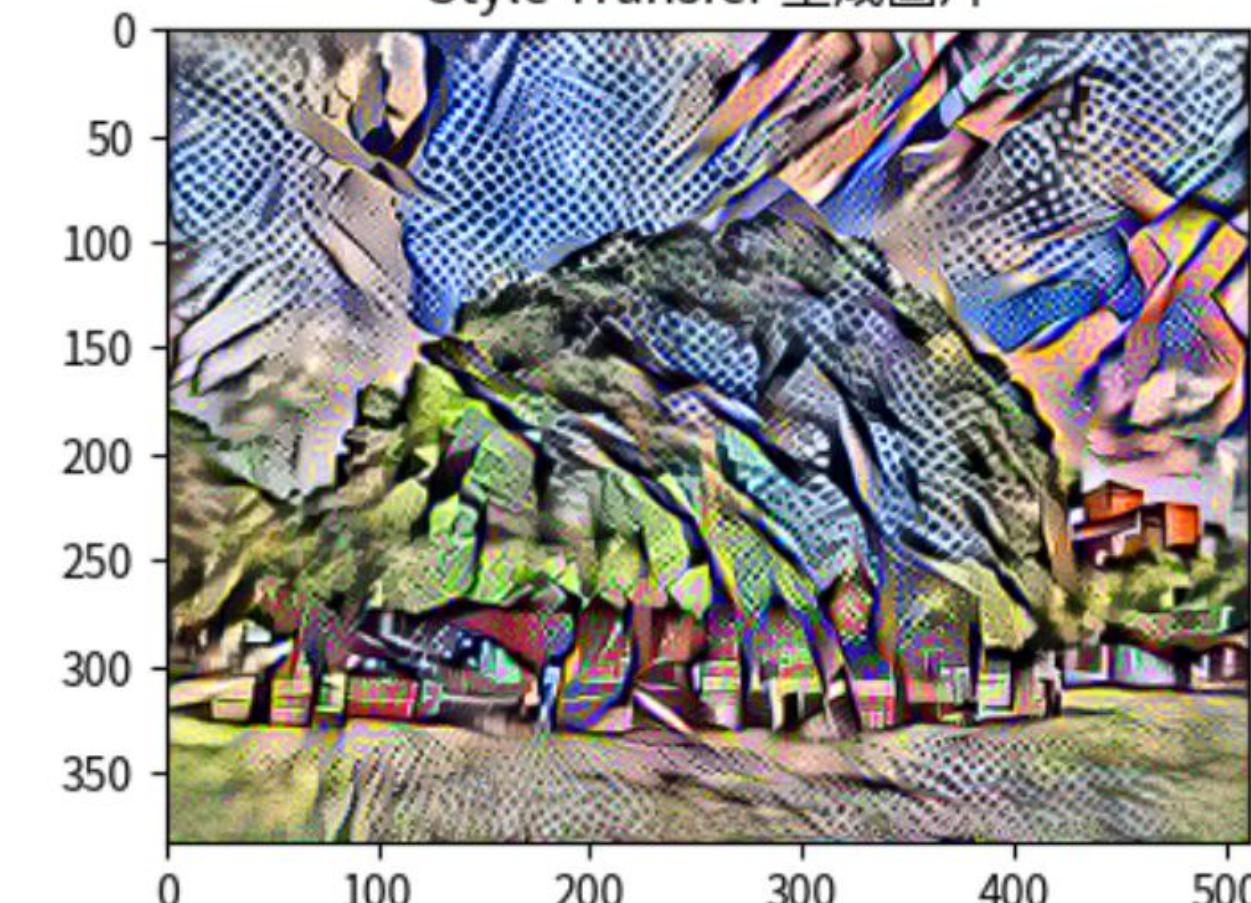
原始內容圖片



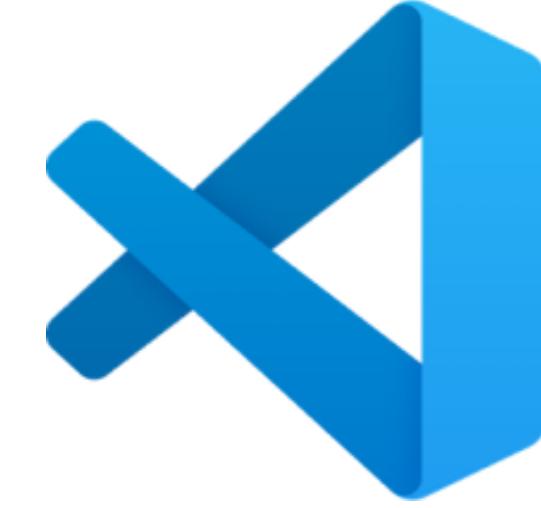
原始風格圖片



Style Transfer 生成圖片



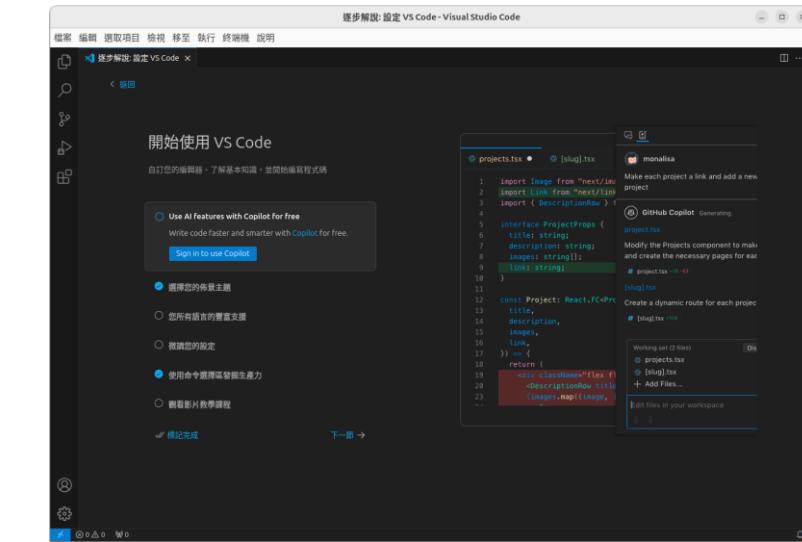
安裝本機端程式編輯器VS Code



◆ Visual Studio Code (簡稱 VS Code) :

是一款由微軟開發且跨平台的免費原始碼編輯器

◆ VS Code 是一款功能強大又彈性的程式編輯器，適合初學者與進階開發者使用。支援 Windows、macOS、Linux，適合用來寫 Python、C++、JavaScript、HTML/CSS、Go、Rust 等各種程式語言，可以依需求擴充，非常靈活。



優點	說明
🌐 跨平台支援	可在 Windows / macOS / Linux 安裝使用
🔌 豐富的擴充功能	有上萬種extension(外掛)，如 Python、Git、Jupyter、Markdown Preview...
🎨 直觀 UI 與主題切換	使用介面現代化，支援暗色主題、字型、配色客製化
🐞 內建除錯功能	支援 breakpoint、變數監看、step-by-step 除錯
💾 內建 Git 支援	可以直接在 VS Code 裡管理 Git / GitHub 專案
🧠 智慧提示 (IntelliSense)	自動補全、參數提示、函式文件查詢，支援多種語言與框架
📁 快速開啟資料夾 / 多檔案編輯	對大型專案管理很方便
📦 支援遠端開發與 Docker、WSL	可遠端連接 Linux 環境、容器進行開發

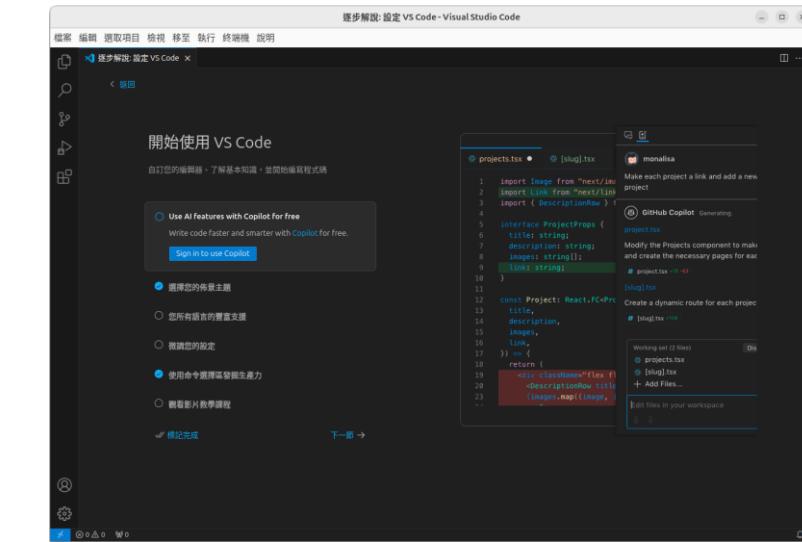
安裝本機端程式編輯器VS Code



◆ Visual Studio Code (簡稱 VS Code) :

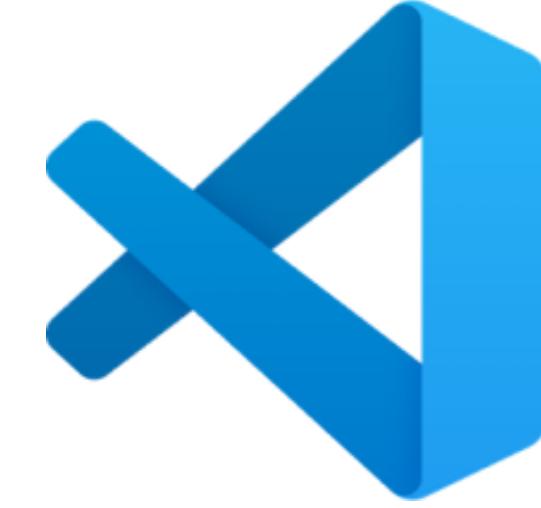
是一款由微軟開發且跨平台的免費原始碼編輯器

◆ VS Code 是一款功能強大又彈性的程式編輯器，適合初學者與進階開發者使用。支援 Windows、macOS、Linux，適合用來寫 Python、C++、JavaScript、HTML/CSS、Go、Rust 等各種程式語言，可以依需求擴充，非常靈活。



缺點	說明
⌚ 啟動速度稍慢 (安裝多插件後)	太多 extension 可能讓執行變慢，特別是老舊電腦
☐ 需要手動設定環境	初學者安裝 Python、C++ 等時需手動設定編譯器、解譯器、路徑等
🛠 不是完整 IDE	雖然功能多，但不如 Visual Studio、PyCharm 那樣針對某些語言深度整合
🌐 需連網下載外掛與套件	初期需下載多個擴充功能來支援特定語言或框架

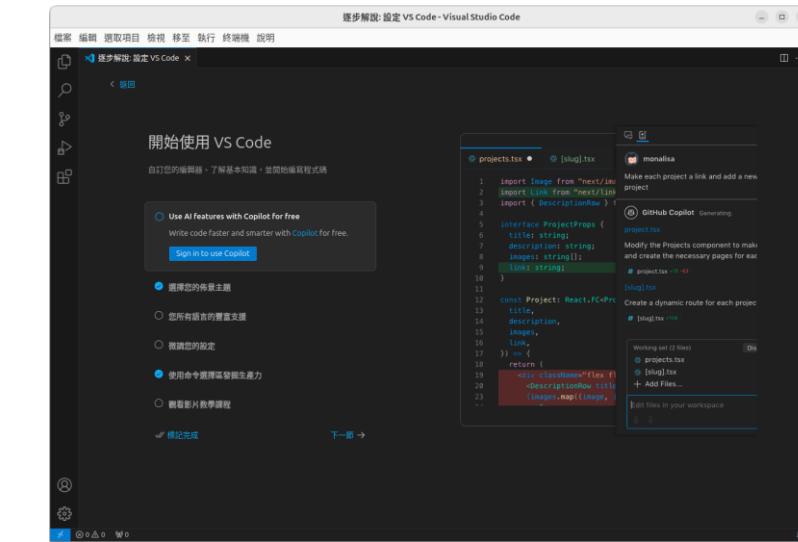
安裝本機端程式編輯器VS Code



◆ Visual Studio Code (簡稱 VS Code) :

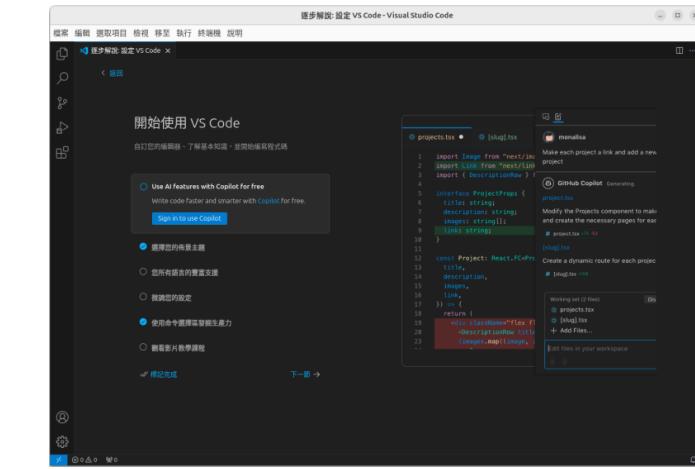
是一款由微軟開發且跨平台的免費原始碼編輯器

◆ VS Code 是一款功能強大又彈性的程式編輯器，適合初學者與進階開發者使用。支援 Windows、macOS、Linux，適合用來寫 Python、C++、JavaScript、HTML/CSS、Go、Rust 等各種程式語言，可以依需求擴充，非常靈活。



語言 / 工具	推薦擴充套件
Python	Python、Jupyter、Pylance
C++	C/C++、Code Runner
網頁前端	HTML/CSS Support、Live Server、Prettier
Git 工具	GitLens、GitHub Copilot (需註冊)
Markdown	Markdown All in One、Markdown Preview Enhanced

安裝本機端程式編輯器VS Code



◆ 安裝步驟：

■ Windows 安裝 VS Code

✓ 步驟：

- 前往官方網站：<https://code.visualstudio.com>
- 點選「Download for Windows」下載安裝檔 .exe
- 執行安裝檔並依序勾選以下選項（推薦）：
 - 在桌面建立捷徑
 - 將「Code」加入 PATH**
 - 在右鍵選單中加入「使用 VS Code 開啟」
- 安裝完成後啟動 VS Code

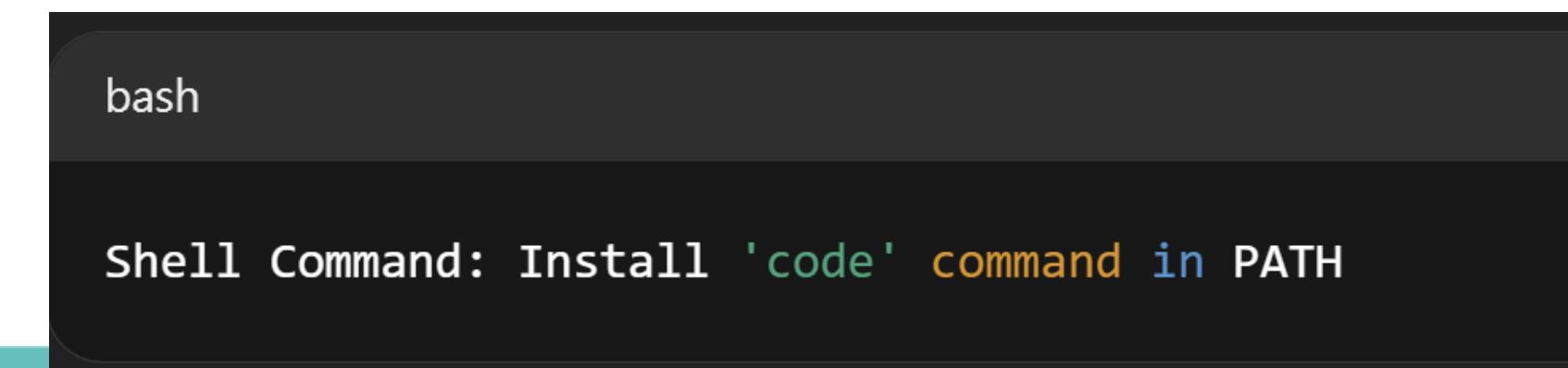
■ 小提醒

- 建議定期更新 VS Code 與 Extensions
- 可安裝中文語言包（搜尋 Chinese 即可）
- 安裝code runner(**非必要!**)

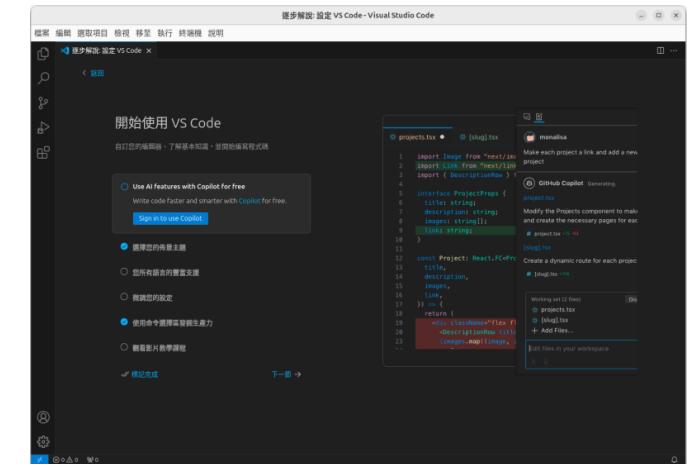
■ macOS 安裝 VS Code

✓ 步驟：

- 前往官網：<https://code.visualstudio.com>
- 點選「Download for macOS」，下載 .zip 安裝包
- 解壓縮後，將 Visual Studio Code.app 拖進「應用程式」資料夾。
- 開啟 VS Code，首次開啟會跳出安全提示
→ 點選「仍要打開」
- (推薦) 加入終端機啟動功能
在 VS Code 中按 Cmd + Shift + P，輸入：



安裝本機端程式編輯器VS Code



◆ 建立虛擬環境：

✓ 步驟：

打開 VS Code 終端機 (Terminal) 輸入：**python -m venv venv\venv\Scripts\activate**

建立虛擬環境的好處：

-這會建立一個「獨立的 Python 執行環境」。

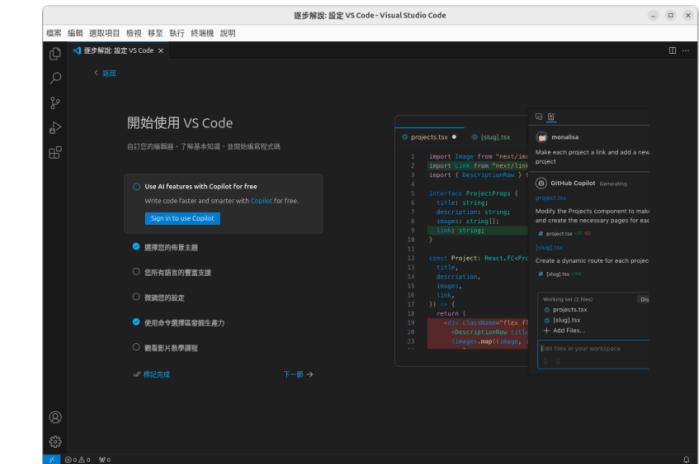
1. 避免套件衝突 (最重要)

- 假設你同時做兩個不同專案：
 - 專案 A 用 TensorFlow 2.11
 - 專案 B 用 PyTorch 2.2
- 如果都裝在全域 (global)，版本衝突很常見。
- 用虛擬環境可以讓每個專案有自己的獨立套件版本，互不干擾。

2. 保持系統乾淨

- 不污染你電腦上的全域 Python。
- 不怕哪天升級 pip 或套件導致整台電腦出錯。

安裝本機端程式編輯器VS Code



◆ 建立虛擬環境：

✓ 步驟：

打開 VS Code 終端機 (Terminal) 輸入：**python -m venv venv\venv\Scripts\activate**

建立虛擬環境的好處：

-這會建立一個「獨立的 Python 執行環境」。

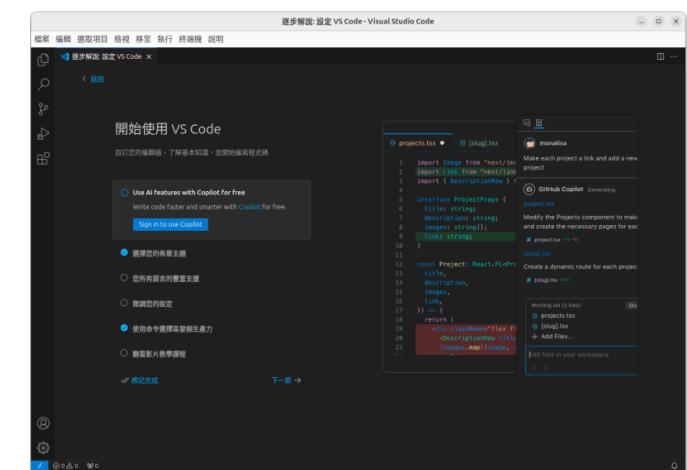
3. 版本可控、方便部署

- 你可以針對這個環境輸出requirements.txt：
 - 指令：pip freeze > requirements.txt
- 方便將這個環境移植到伺服器或其他人的電腦上。

4. 符合專業團隊開發流程

- 多人協作時，使用虛擬環境與 requirements.txt 是業界標準。
- 可搭配 VS Code 自動偵測虛擬環境，讓 IntelliSense 與執行更順暢。

安裝本機端程式編輯器VS Code



◆ 建立虛擬環境：

✓ 步驟：

打開 VS Code 終端機 (Terminal) 輸入：**python -m venv venv.\venv\Scripts\activate**

建立虛擬環境的好處：

-這會建立一個「獨立的 Python 執行環境」。

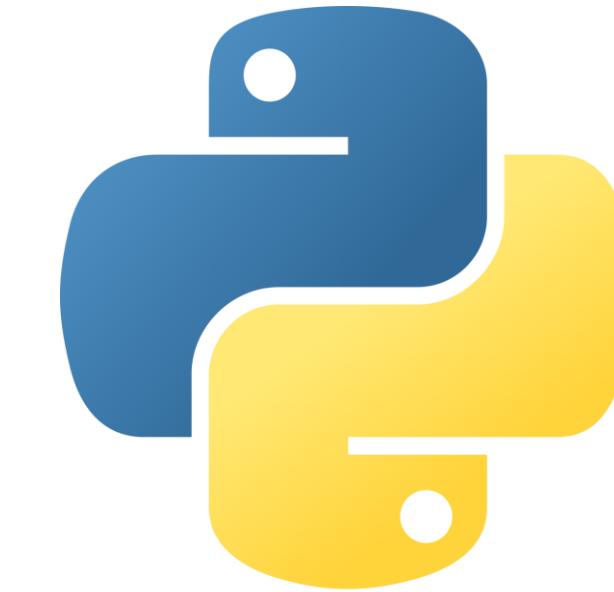
5. VS Code 相容性佳

- VS Code 會自動識別 .venv 目錄，提供套件提示與除錯。
- 啟動時會問你要不要用這個虛擬環境，很方便。

🕒 小提醒

- 虛擬環境建立後，只需執行一次 activate 指令即可使用：
 - 指令：.\venv\Scripts\activate
- 若你看到前面出現 (venv)，就表示啟動成功。

本機端Python環境安裝與設定



◆ 安裝步驟：

■ 安裝Python

✓ 步驟：

1. 安裝 Python 本體(版本Python 3.9.20)：<https://www.python.org/downloads>

2. 在 VS Code 安裝 Python Extension：

- 點選左側 Extensions 圖示（或 Ctrl+Shift+X）
- 搜尋並安裝「Python」

3. 第一次執行 .py 時，會自動偵測 Python 路徑，如無法偵測：

- 按 Ctrl+Shift+P → 選擇 Python: Select Interpreter → 選你安裝的 Python

■ 小提醒

- 建議定期更新 VS Code 與 Extensions

在VS Code裡面安裝與使用TensorFlow+Keras搭配GPU(CUDA+cuDNN)

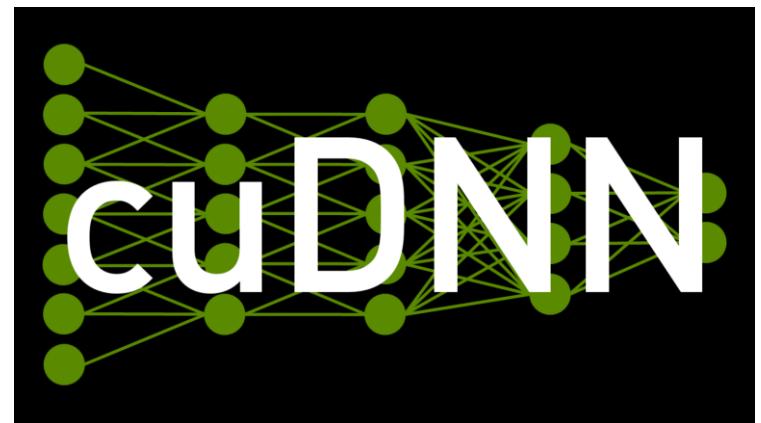
◆ 安裝步驟：

✓ 步驟：

1. 確認你的 GPU 支援 CUDA

➤ 可以使用NVIDIA官網查詢是否支援（如 RTX 3050 支援 CUDA 12.x）。

<https://developer.nvidia.com/cuda-gpus>



2. 安裝正確版本的 CUDA Toolkit(本課程使用：TensorFlow 2.10 搭配 CUDA 11.8)

➤ 到 NVIDIA 官方下載對應 TensorFlow 的 CUDA 版本：

<https://developer.nvidia.com/cuda-toolkit-archive>

3. 安裝對應版本的 cuDNN(本課程使用：cuDNN 8.6.0.163)

➤ 下載 cuDNN（必須註冊 NVIDIA 帳號）：

<https://developer.nvidia.com/cudnn>

➤ 解壓縮後手動將檔案放入 CUDA 資料夾中：

• bin/, lib/, include/ 對應複製到 C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.X\

在VS Code裡面安裝與使用TensorFlow+Keras搭配GPU(CUDA+cuDNN)

◆ 安裝步驟：



4. 設定環境變數 (Windows)

➤ 將以下路徑加入環境變數 Path 中：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.X\bin

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.X\libnvvp

5. 安裝 TensorFlow (GPU 版)

➤ 開啟 VS Code Terminal，輸入：

pip install tensorflow==2.10.0

注意：

- TensorFlow 2.10.0是最後一個支援 Windows GPU 的版本（官方從 2.11 起不再支援 Windows 上的 GPU 執行）。
- 安裝環境請確保版本一致，否則會報錯（例如 DLL 載入錯誤）。

Python、TensorFlow、Keras、CUDA與cuDNN的版本相容性

◆ 注意：

1. TensorFlow 2.11開始停止支援Windows GPU安裝，只支援Linux + Docker。Windows需使用2.10(最穩定，推薦使用)或更早版本。
2. 直接使用TensorFlow內建的Keras，不要單獨安裝 Keras 套件，除非你知道自己在做什麼。

【相容性對照表總覽】

TensorFlow 版本	Python 版本	CUDA 版本	cuDNN 版本	Keras 版本
2.15.x	3.8 ~ 3.11	11.8	8.6	2.15.x (內建)
2.14.x	3.8 ~ 3.11	11.2	8.1	2.14.x (內建)
2.10.x	3.7 ~ 3.10	11.2	8.1	2.10.x (內建)
2.9.x	3.7 ~ 3.9	11.2	8.1	2.9.x (內建)
2.8.x	3.7 ~ 3.9	11.2	8.1	2.8.x (內建)

【建議安裝步驟順序及摘要】

(1)安裝Python 3.10.11，(2) 安裝CUDA 11.8，(3)安裝cuDNN 8.6.0.163，(4)安裝TensorFlow 2.10.0。

Python、TensorFlow、Keras、CUDA與cuDNN之間的關係

USE-CASES



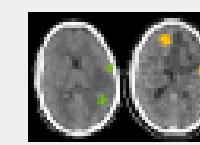
Speech



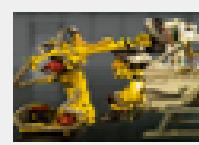
Translate



Recommender



Healthcare



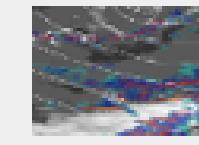
Manufacturing



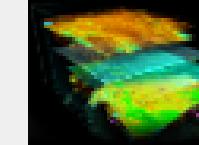
Finance



Molecular Simulations



Weather Forecasting



Seismic Mapping

CONSUMER INTERNET

INDUSTRIAL APPLICATIONS

SUPERCOMPUTING

APPS & FRAMEWORKS



Amber
NAMD

+600
Applications



CUDA-X LIBRARIES

MACHINE LEARNING

cuDF cuML cuGRAPH

DL / HPC

cuDNN CUTLASS TENSORRT CUDA Math Libraries

LANGUAGES

python OpenACC



LLVM Compiler
For CUDA

CUDA

CUDA TOOLKIT

CUDA
COMPILER

DEVELOPER TOOLS

DEBUGGERS PROFILERS

CUDA C++
CORE

CUDA DRIVER

MEMORY
MANAGEMENT

WINDOWS &
GRAPHICS

COMMS
LIBRARIES

OS PLATFORMS



CentOS



Windows Server

Python、TensorFlow、Keras、CUDA與cuDNN之間的關係

元件	角色說明
Python	<ul style="list-style-type: none">程式語言，用來撰寫呼叫深度學習框架的使用者端程式。透過 import tensorflow as tf 在 Python 裡啟動 TensorFlow。
TensorFlow	<ul style="list-style-type: none">Google 開發的深度學習框架，內部以 C++ 撰寫核心運算，外層提供 Python API。負責建立計算圖、管理資料流，以及調度到 CPU 或 GPU 上執行。
Keras	<ul style="list-style-type: none">高階神經網路 API，內建在 TensorFlow (tf.keras) 中。提供更簡潔易用的模型定義、訓練與評估介面。底層仍呼叫 TensorFlow 的運算核心。
CUDA	<ul style="list-style-type: none">NVIDIA 推出的 GPU 計算平台與程式模型。提供編譯工具鏈(nvcc)、驅動程式，以及執行時支援，讓 TensorFlow 能在 NVIDIA GPU 上跑。
cuDNN	<ul style="list-style-type: none">NVIDIA 專門為深度學習提供的 GPU 加速函式庫(Convolution、RNN、Pooling...)。TensorFlow 在執行卷積、BatchNorm、RNN 等運算時，會呼叫 cuDNN 的高效實作，大幅提升速度。

Python、TensorFlow、Keras、CUDA與cuDNN之間的關係

執行流程簡述：

1. 撰寫

使用 Python + tf.keras 定義模型、編譯與訓練參數。

2. API 呼叫

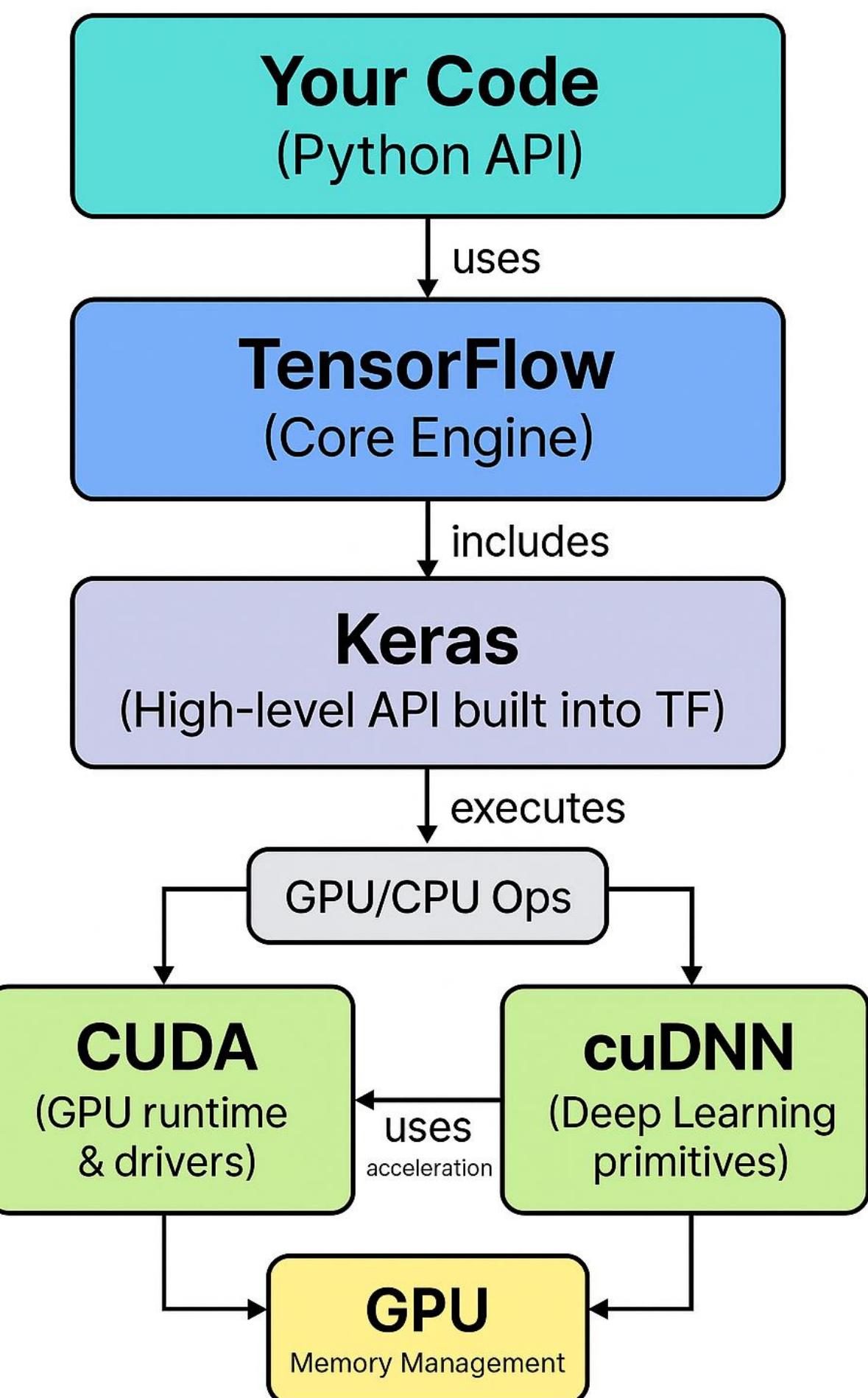
Keras 轉成 TensorFlow 的計算圖與運算節點（ops）。

3. 裝置分派

TensorFlow 檢查可用硬體：若有 GPU 且正確安裝 CUDA/cuDNN，就把符合條件的 ops 派到 GPU；其餘在 CPU 上執行。

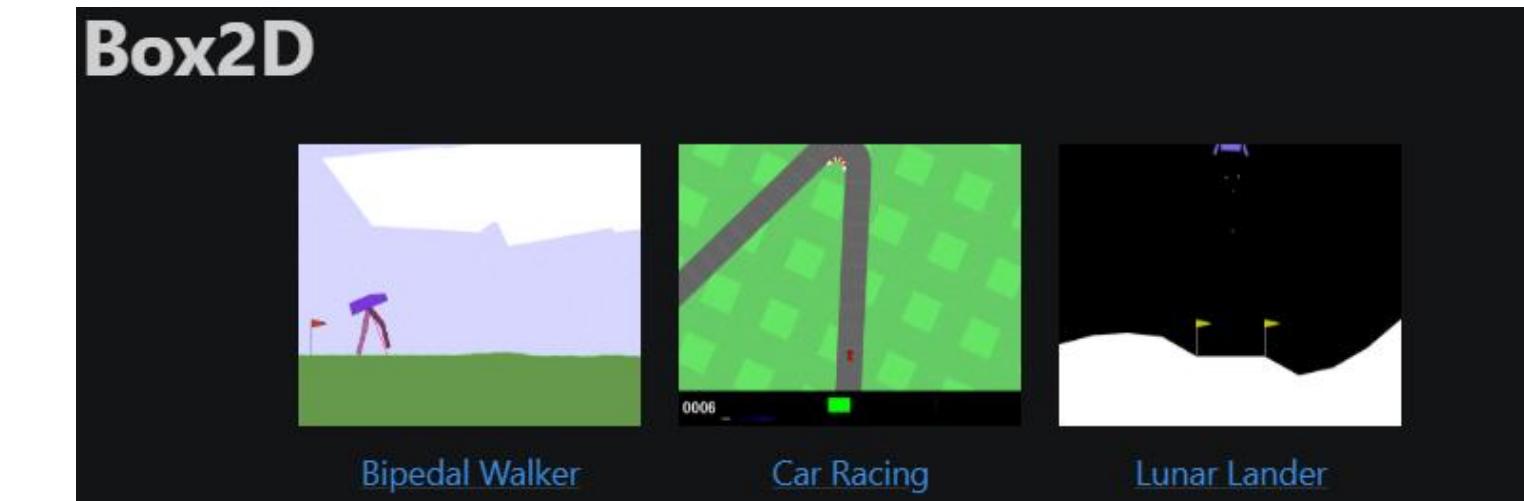
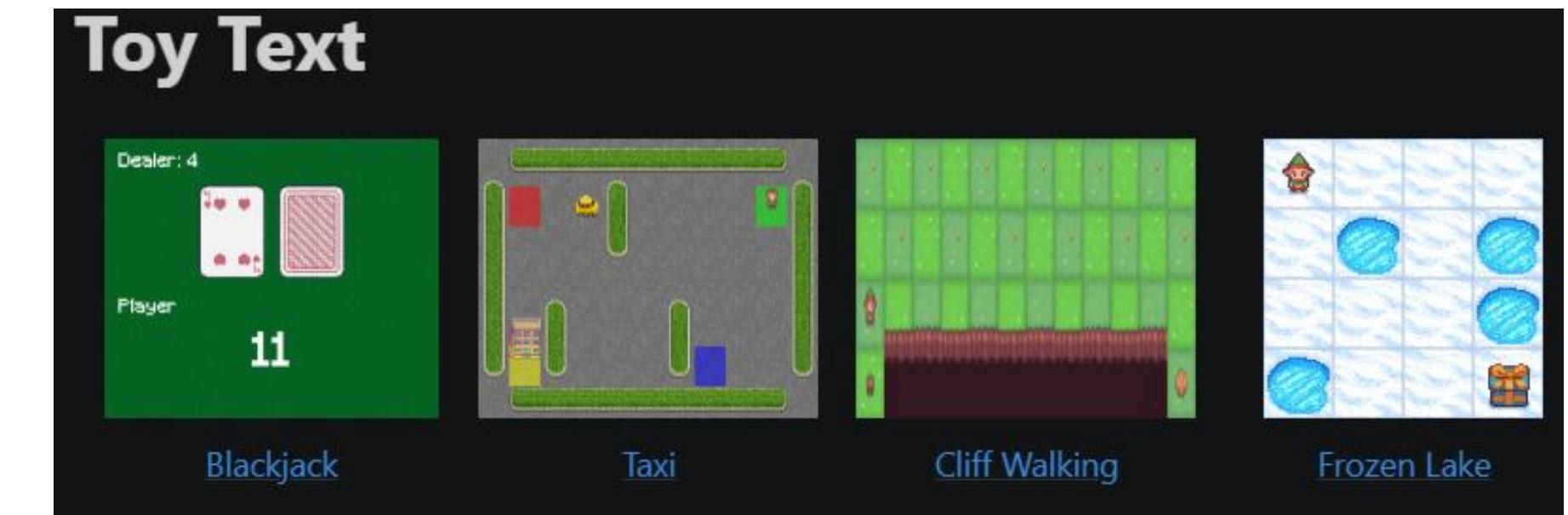
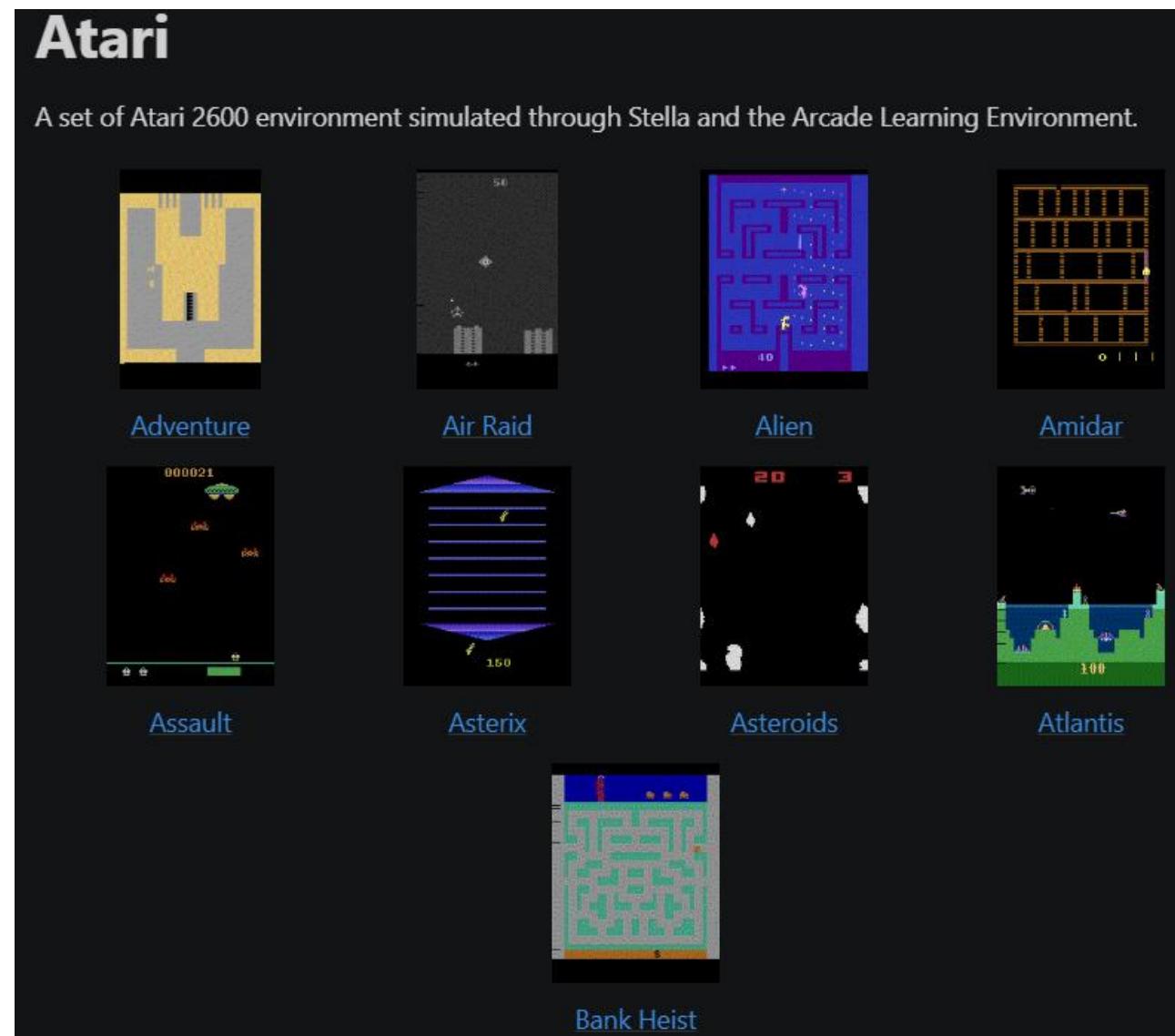
4. 底層運算

在 GPU 上時，TensorFlow 會呼叫 cuDNN（或其他 CUDA 底層庫）完成高效矩陣與卷積運算。



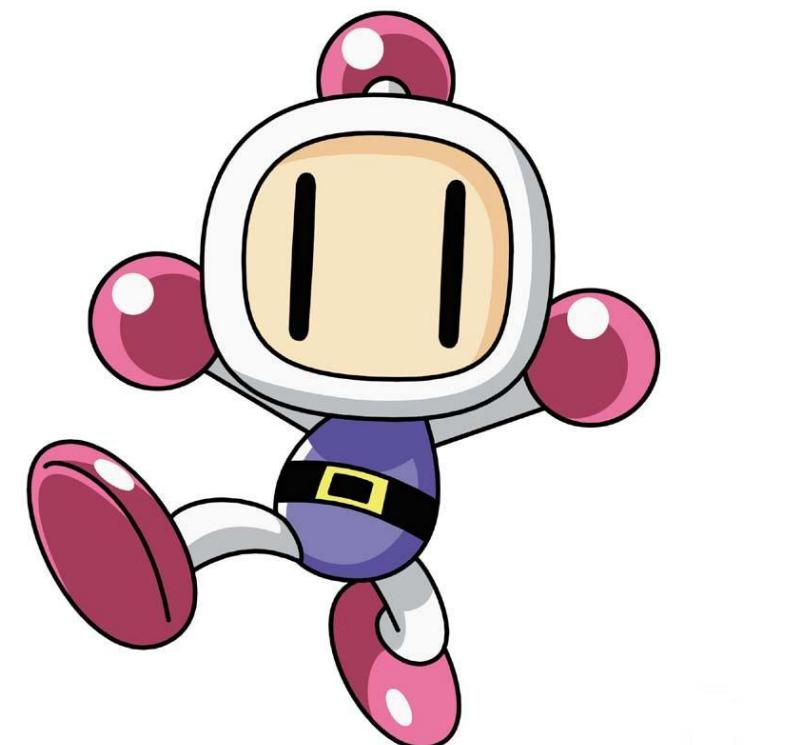
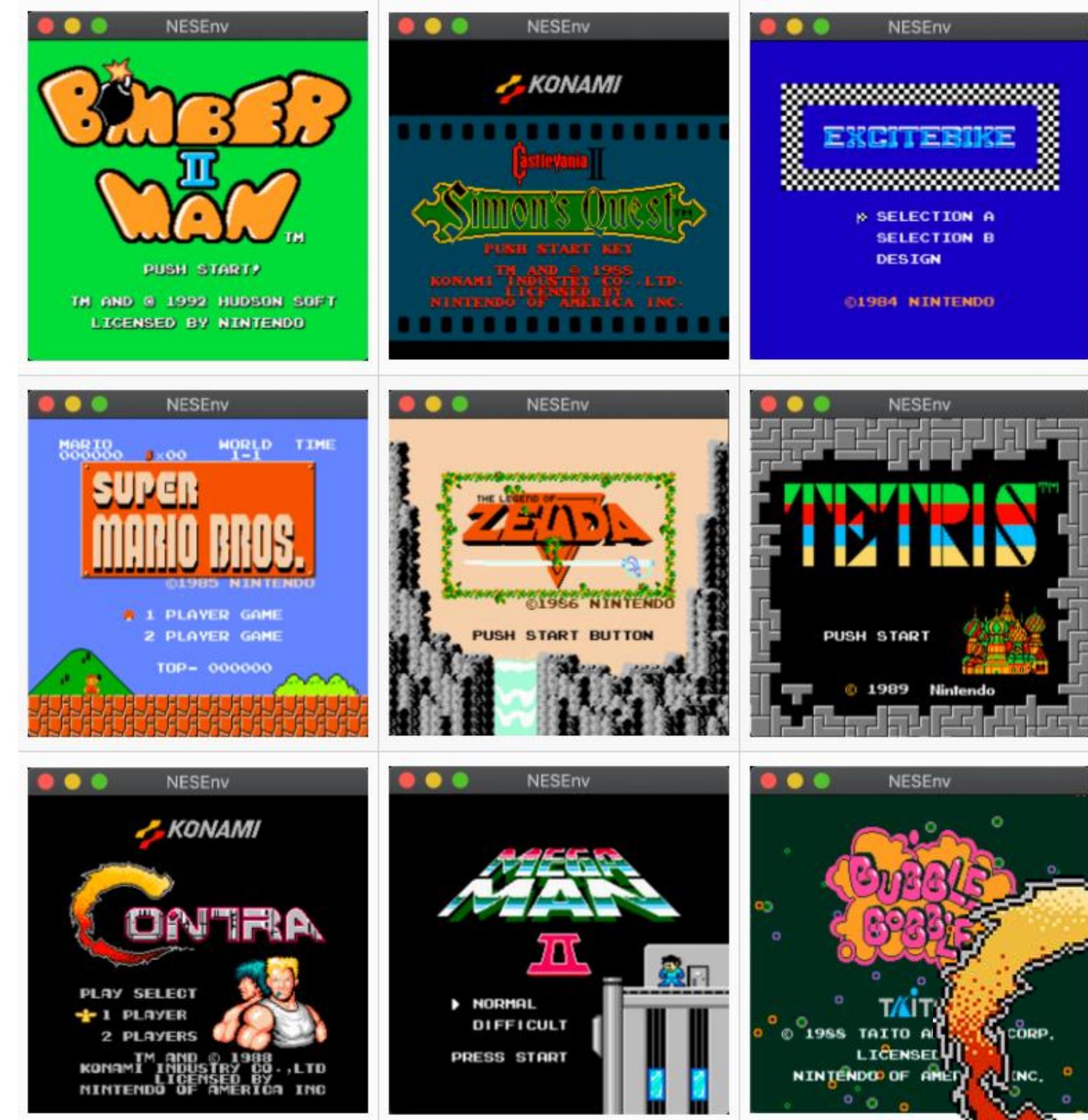
OpenAI Gym

OpenAI Gym是由Open AI開發的開源函式庫，是一個可以提供多種預設環境來讓使用者練習跑RL的模組。也可以根據需求來自己創造環境



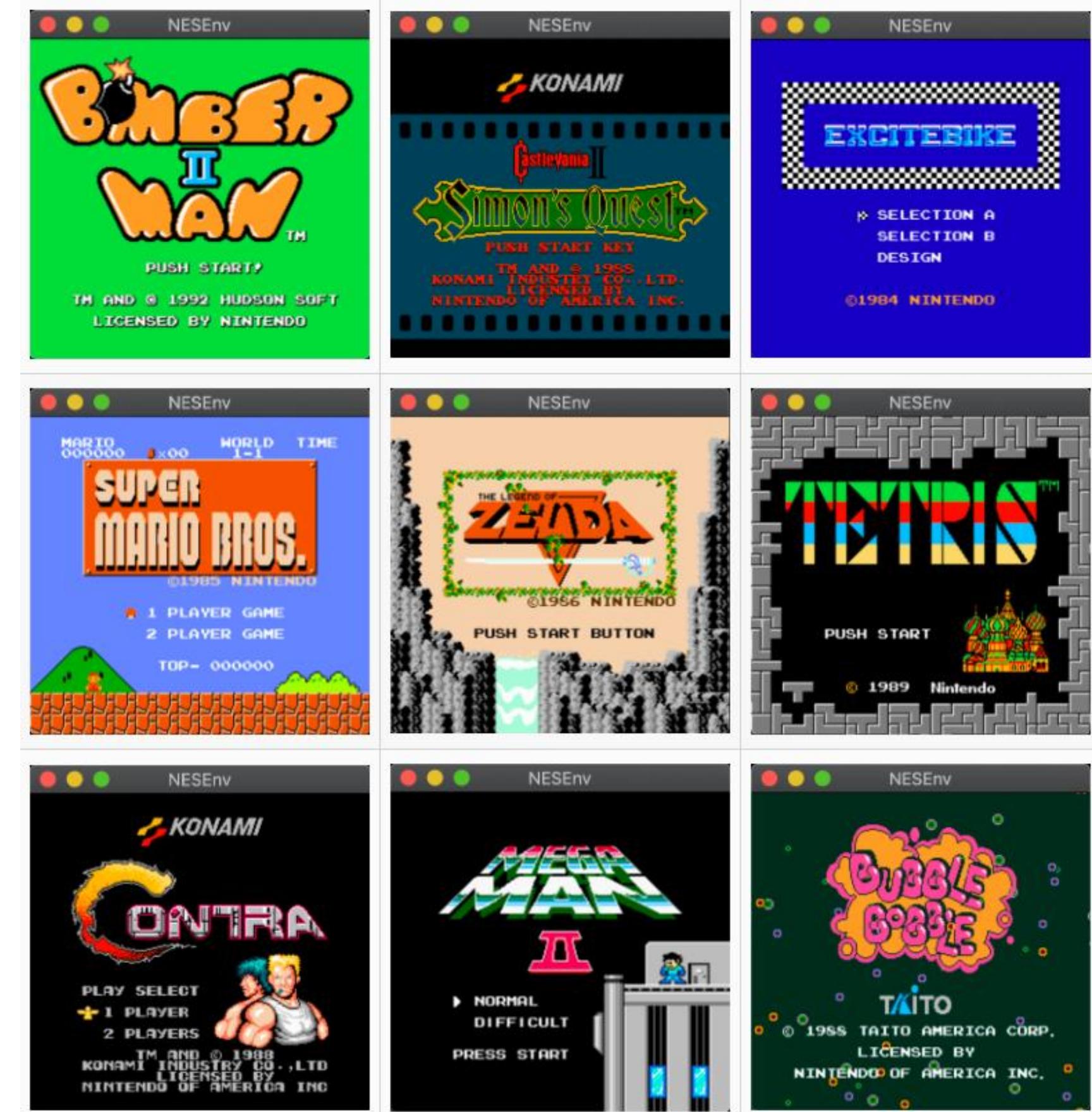
基本介紹與環境建置

Nes-py



Nes-py介紹

Nes-py是一個基於 Python 的 Nintendo Entertainment System (NES) 模擬器框架。它提供了一個高效、易用的模擬環境，適合用於強化學習的研究和開發。能夠模擬經典的 NES 遊戲，例如《Super Mario Bros》、《Contra》和《The Legend of Zelda》



Nes-py環境建置

Visual Studio 工具

Visual Studio 2022 遠端工具

Visual Studio 2022 遠端工具可實現在未安裝 Visual Studio 的電腦上進行應用程式部署、遠端偵錯、遠端測試、效能分析以及裝置測試。使用此工具需要有效的 Visual Studio 授權。

下載

繁體中文 (Chinese, Traditional) ▾

AMD64 ARM64 x86

IntelliTrace Standalone Collector for Visual Studio 2022

IntelliTrace 獨立收集器可讓您為實際執行伺服器的應用程式收集診斷資料，而無須安裝 Visual Studio 或重新部署應用程式。使用此工具需要有效的 Visual Studio 授權。

下載

Agents for Visual Studio 2022

Agents for Visual Studio 2022 可用於負載、功能及自動化的測試。使用此工具需要有效的 Visual Studio 授權。

下載

Agent Controller

Build Tools for Visual Studio 2022

這些 Build Tools 可讓您從命令列介面建置 Visual Studio 專案。支援的專案包括 ASP.NET、Azure、C++ Desktop、ClickOnce、容器、.NET Core、.NET Desktop、Node.js、Office 和 SharePoint、Python、TypeScript、單元測試、UWP、WCF 以及 Xamarin。除非您正在建置專案的開放原始碼相依性，否則使用此工具需要有效的 Visual Studio 授權。如需詳細資訊，請參閱[建置工具授權](#)。

下載

您要尋找其中一個 Visual Studio 2022 [長期維護通道 \(LTSC\)](#) 嗎？您可以在[這裡](#)找到。

各版本資訊：
Python=3.9.20
Numpy=1.25
Pytorch=2.5.1
gym=0.26.2
gym-super-mario-bros=7.3.0

[Pytorch網站](#)

[Build Tools for Visual Studio 2022](#)

[Nes-py安裝](#)



Super Mario Test

排除萬難執行run.py

✓ 今天			
	 run	2025/12/12 下午 01:38	Python 來源檔案 6 K
	 reward	2025/12/12 下午 01:35	Python 來源檔案 2 K
	 __pycache__	2025/12/12 下午 01:39	檔案資料夾
✓ 上週			
	 DQN	2025/12/1 下午 06:33	Python 來源檔案 4 K
	 eval	2025/12/1 下午 06:33	Python 來源檔案 3 K
	 model	2025/12/1 下午 06:33	Python 來源檔案 3 K
	 utils	2025/12/1 下午 06:33	Python 來源檔案 1 K
✓ 很久以前			
	 ckpt	2024/12/11 下午 12:01	檔案資料夾

Q & A

