

# NKUST 機器學習期末報告

隊伍：TEAM\_9305

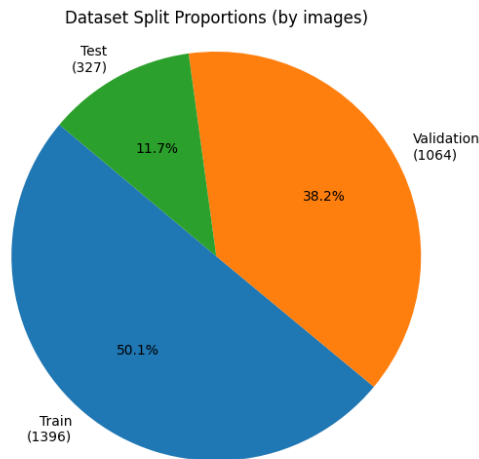
隊員：魏司諺

Private leaderboard：0.93375012/0.9279647988481136

## 壹、資料處理與程式環境

### 一、資料前處理

本研究使用之資料來源為AI Cup官方提供之影像資料集(競賽規定資料集)，影像格式為 PNG，並包含對應之標註檔。研究過程中，先將官方提供之訓練資料與標籤上傳至雲端空間，並下載至 Google Colab 環境進行後續處理與模型訓練。資料切分方式採用YOLO 建機制，將資料依比例劃分為**訓練集**、**驗證集**、**測試集**，比例為**5:4:1**，以確保模型訓練與評估之穩定性。為提升模型泛化能力，訓練階段加入資料增強方法，包括影像水平翻轉、垂直翻轉、影像縮放，以及 HSV 色彩空間調整，以模擬不同影像條件並增加資料多樣性。



### 二、原始程式碼概述

本研究之程式碼架構係以 AI Cup 官方提供之範例程式碼作為基礎，主要流程包含資料下載、資料檢查與模型訓練等步驟。程式於資料準備階段會先下載影像資料與對應標註，並自動**剔除未包含有效標註之影像**，以避免無效資料影響訓練結果；隨後透過 YOLO 架構進行模型訓練，訓練完成後會產生模型權重檔 (.pt)，並可於後續實驗中載入該權重進行延續訓練或參數微調，以提升模型效能與訓練效率。同時亦建立IoU等模型評估指標，確保最終模型準確性。

### 三、執行環境

本研究之模型訓練與實驗主要於 Google Colab 平台上進行，並作為唯一之成果產出與效能評估環境。研究中期曾短暫於本地端透過 Visual Studio Code 進行環境測試，硬體為 AMD Ryzen 7 5800X 與 NVIDIA RTX 3080、作業系統為 Windows 11，惟未產生任何可用成果，後續即完全捨棄本地端訓練，並統一以 Google Colab 作為正式實驗平台。程式語言採用 Python，整體架構以 AI Cup 官方提供之範例程式 (ipynb) 為基礎進行操作，主要使用 ultralytics 套件進行 YOLO 模型訓練與推論，並搭配 gdown、os、shutil 與 locale 等函式庫進行資料下載與檔案管理。模型方面，本研究採用 YOLOv9-m(研究初期亦曾採用 12n、9s 等模型) 之預訓練權重進行遷移學習，以加速模型收斂並提升整體偵測效能。

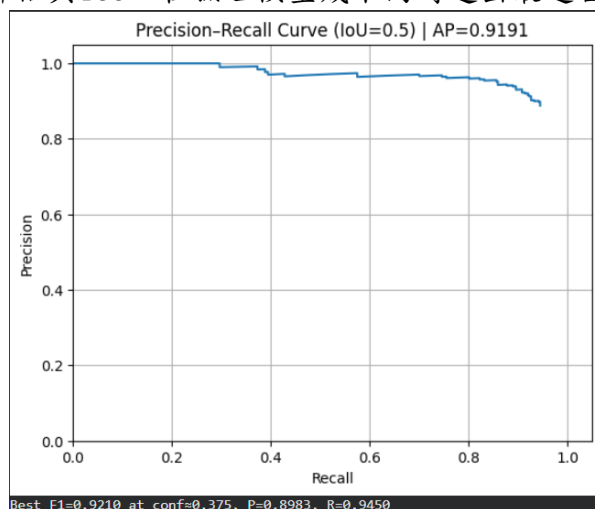
## 貳、模型設計與訓練策略

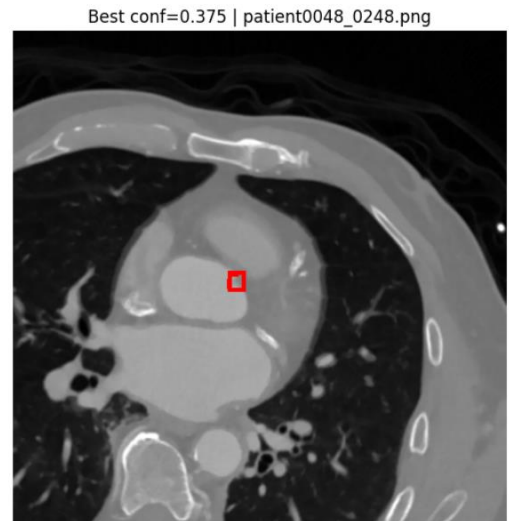
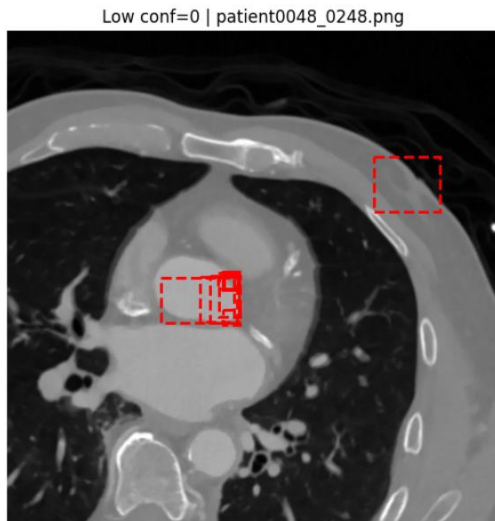
### 一、模型訓練原始程式碼與說明

本研究採用基於深度學習之物件偵測方法，使用 Ultralytics YOLO 架構進行主動脈瓣影像之自動定位與辨識。所有訓練與推論程式皆於 Google Colab 平台撰寫與執行，透過雲端 GPU 加速模型訓練流程。程式內容涵蓋資料前處理、資料分割、模型訓練、模型評估以及最終推論輸出，形成一套完整且可重現之物件偵測實驗流程。github連結：[c112154141-collab/AI\\_cup\\_TEAM\\_9305: NKUS\\_T\\_EE\\_ML\\_期末報告&AI\\_cup\\_競賽報告](https://github.com/c112154141-collab/AI_cup_TEAM_9305: NKUS_T_EE_ML_期末報告&AI_cup_競賽報告)

### 二、模型訓練流程

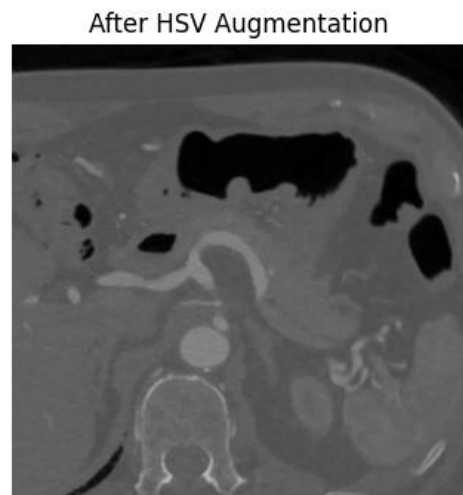
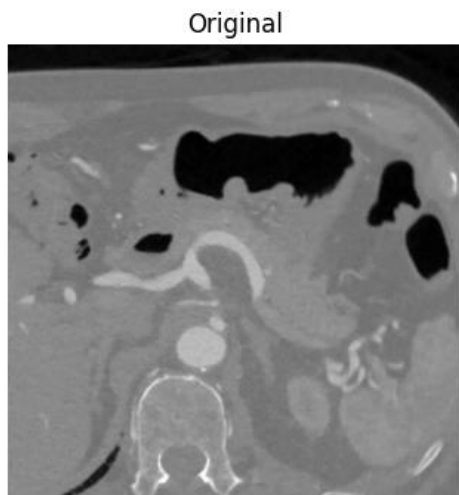
模型訓練流程首先由程式自動下載並解壓縮影像與標註資料，接著僅保留具有標註之影像進行訓練，以避免無效資料影響模型學習。資料集以病人 (patient) 為單位進行切割，比例設定為訓練集、驗證集與測試集分別為 5:4:1。完成資料分割後，使用 YOLO 預訓練模型進行微調訓練，並於訓練結束後直接對測試集進行模型效能評估與 IoU，在驗證模型成果同時選出最適合 conf。





### 三、參數設定

模型訓練時，訓練輪數設定為 100 至 150 epochs，以確保模型充分收斂，批次大小設定為 8 以符合 Colab Tesla T4 GPU 記憶體限制，輸入影像尺寸為 640x640。資料增強策略以醫學影像合理性為原則，僅採用低比例 Mosaic (0.05)、水平與垂直翻轉、影像縮放及 HSV 色彩調整，避免過度扭曲影像結構。損失函數權重設定為 box=7.5、cls=0.5、dfl=1.5，並使用預設自動最佳化之 optimizer 以提升訓練穩定性。



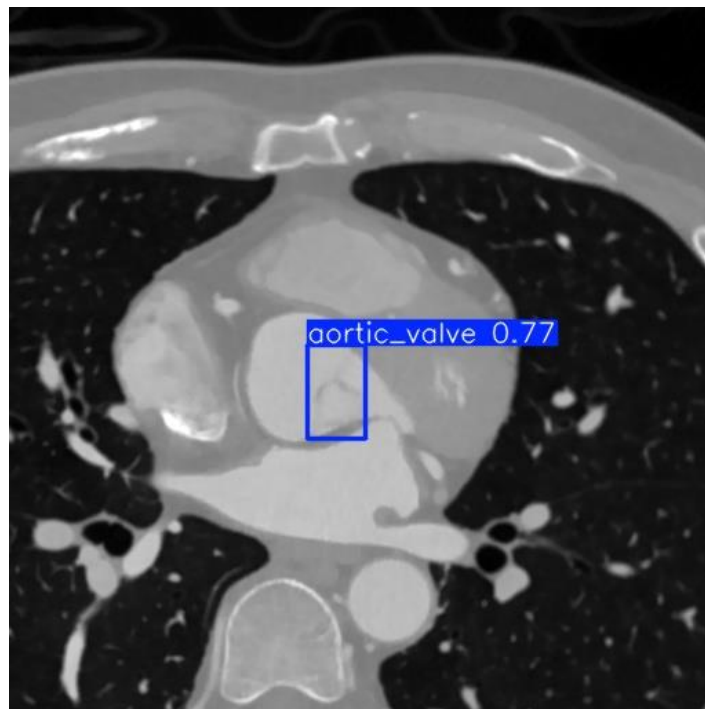
### 參、分析與結論

本研究完成以 YOLO 為核心之心臟主動脈瓣物件偵測模型訓練流程，並實作完整的資料前處理、模型訓練、測試集評估與 IoU 指標分析。整體流程皆於 Google Colab 平台上進行，並以 Tesla T4 GPU 作為主要運算設備。在實際訓練過程中，雖嘗試調整多項超參數與資料增強策略以提升模型表現，但最終成果仍未優於 AI CUP 官方所提供之基準範例模型。

經分析後，主要影響因素來自運算資源的限制。由於顯示卡記憶體容量有限，訓練過程中需在批次大小、影像解析度與模型規模之間進行取

捨，導致無法同時採用較大的 batch size 與較複雜的資料增強策略。此外，在顯存壓力下，部分訓練設定（如較高訓練輪數或進階增強方法）容易導致訓練不穩定，進而影響模型收斂品質。

相較之下，官方範例模型以相對簡潔的訓練設定，即可在有限輪數內達到穩定且具競爭力的效能，顯示其模型結構與預設參數已高度優化。本研究雖未在最終評分指標上取得優勢，但透過完整實作訓練與測試流程，實際體會到硬體資源對深度學習模型效能之關鍵影響。整體而言，本研究成果突顯出模型表現不僅取決於演算法與參數調整，更高度依賴運算設備條件，未來若能於較充足的硬體環境下進行訓練，模型效能仍具進一步提升之可能。



## 肆、使用的外部資源與參考文獻

- Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A review of YOLO algorithm developments. *Procedia Computer Science*, 199, 1066 – 1073. <https://doi.org/10.1016/j.procs.2022.01.135>
- AI Cup. (2025). [Competition name/task name]: Official baseline code. [Website]. <https://tbrain.trendmicro.com.tw/Competitions/Details/42>
- OpenAI. (2024). ChatGPT [Large language model]. <https://chat.openai.com>
- DeepSeek AI. (2024). DeepSeek: Large language model for reasoning and coding. <https://www.deepseek.com>
- xAI. (2024). Grok: A generative AI assistant. <https://x.ai>

使用chat GPT、deepseek、Grok進行訓練參數推薦與語法偵錯，同時使用GPT輔助進行報告潤稿、github上傳及格式微調