食品藥物研究年報. 16：354-368 2025
Ann. Rept. Food Drug Res. 16：354-368 2025

# 研析影像辨識技術應用於食品廣告之探索

郭岳翰　吳孟縈　王兆儀

衛生福利部食品藥物管理署食藥戰情中心

## 摘　要

影像辨識技術為人工智慧與計算機視覺領域的重要發展，特別是在結合物件辨識技術YOLO (You Only Look Once)與光學字元辨識OCR (Optical Character Recognition)技術後，展現出廣泛之應用潛力。本研究利用食品廣告之錄影影像進行測試，結果顯示前揭技術所建置之模型能於多數場景中穩定且準確地判別疑似違規內容，具備良好之辨識效能。在測試過程中，YOLO模型能快速定位目標區域，OCR技術則針對精確區域提取文字，兩者結合可降低背景干擾，提升運算效率與辨識精度，特別適用於場景複雜或資訊密集之影像辨識任務，另再透過蒐集公開之食品違規廣告裁處案件及法規敘述及違規誇大字詞等建立之辭庫導入比對，能有效識別出疑似違規用詞等特徵，展現出良好之場景適應性，顯示未來應用於網路巡查疑似違規訊息及影片等領域頗具潛力。然本次測試亦仍存在一些限制，如背景過於複雜、光線極端條件下，物件定位及文字擷取準確性下降，專有名詞、特殊字體之辨識，於快速移動物件或低解析度影像中表現不穩，特別是在高度創意或非標準化字體設計之影像中，辨識結果易受影響。未來可持續針對應用需求進行模型優化，提升對特殊字體及複雜背景之適應能力，並確保影像輸入之品質標準。整體而言，本研究驗證影像辨識技術於提升影片巡查效率之可行性，惟實際應用時仍需考量硬體運算資源等因素，建議可透過升級高效能硬體(如GPU)與技術優化，以進一步拓展應用場景，實現更高效、精準且穩健之影像辨識能力，滿足多元場域之需求。

**關鍵詞：影像辨識、食品安全管理、食品廣告、OCR、YOLO**

## 前　言

面對資訊快速傳播的時代，影像已成為日常傳播中最常見的形式之一，不肖業者將疑似違規之內容潛藏於影像資料中的現象層出不窮，且手法日趨隱蔽，傳統人工巡查方式在效率與準確性上皆面臨挑戰，隨著影像資料量呈指數型成長，倚賴人力進行全面監管已顯不足，導入自動化辨識技術以提升巡查之即時性與全面性已成趨勢。因此，如何快速且準確地辨識影像中之關鍵內容，成為巡查與監管工作中的重要課題；近年隨著人工智慧與深度學習技術的發展，影像辨識技術已逐步成熟，尤以物件偵測(Object Detection)與文字辨識(Optical Character Recognition, OCR)為核心應用，展現出強大的影像處理能力與廣泛應用潛力。本研究主要探討並實作物件檢測技術(You Only Look Once, YOLO)與光學字元辨識技術OCR於影像辨識之應用，藉由蒐集電商平台與食品廣告中之實際影像進行測試與驗證，評估其於疑

似違規廣告自動辨識上之可行性與準確性，並針對現行技術之成效與限制進行分析。透過本次試作，期提供具實用性且可延伸應用之智慧巡查輔助規劃，以提升行政效率，並為未來擴大應用場景及優化模型策略奠定基礎。

## 影像辨識技術概述

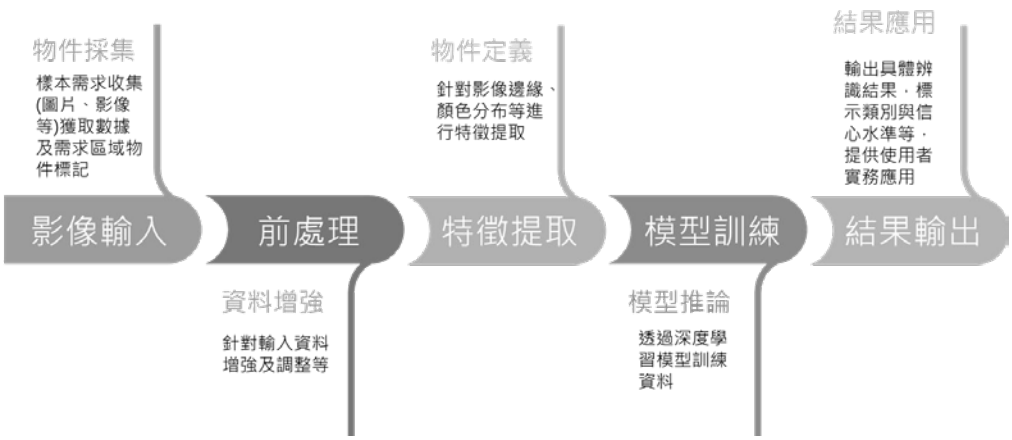
影像辨識(Image Recognition)是人工智慧和計算機視覺領域中的一項技術，主要功能是分析和處理影像，將影像中的內容或特徵轉換為使用者可理解的資訊，使其能夠辨識和解釋影像內容中的數據。影像辨識的原理主要為模仿人類視神經的運作方式，從辨識邊界開始，逐步組合出圖像，進而完成識別[1]，在深度學習的影像辨識中，這一過程是通過多層次的處理實現的；起始模型會將圖片分解為大量的小像素，作為輸入資料，並透過多層演算法逐步從像素中進行特徵提取，並將特徵進一步組合，最後在輸出層生成最終的辨識結果[2]；影像辨識技術的基礎仰賴於深度學習模型，尤其是卷積神經網絡(Convolutional Neural Networks, CNN)和轉換器模型(Transformers)。這些模型能夠學習影像中的空間和語義特徵，實現自動化特徵提取，並進行高效分類[3]。另為了進一步提升影像辨識系統之性能，通常結合多項技術加以強化，如資料增強(Data Augmentation)技術透過對訓練影像進行旋轉、翻轉、裁剪、縮放等操作，以增加資料多樣性，減少過擬合(Overfitting)現象，進而提升模型的泛化能力[4]。

在物件偵測方面，常用技術如YOLO與Faster R-CNN (Faster Region with Convolution Neural Network)，YOLO可將物件偵測視為單一問題，實現即時且高速的偵測效果，適合大量即時處理需求；Faster R-CNN則結合區域提議網路(Region Proposal Network, RPN)，兼顧高準確率與良好運算效率，適用於精度要求較高的應用場景[5,6]；語意分割(Semantic Segmentation)領域，常見的UNet與SegNet (Segmentation Network)技術，則可將影像分割為具有特定意義的區域，實現像素級別的精細分析，其中，UNet具備編碼器-解碼器結構，能在小樣本條件下有效進行像素分類，而SegNet則強調解碼過程中保留空間資訊，適合複雜場景下之邊界辨識需求[7,8]，以上技術在提升辨識效能、增強模型適應性與提高系統穩健性方面，皆扮演關鍵角色，並可依據不同應用情境靈活搭配使用，以滿足多樣化的影像辨識需求。

綜上所述，影像辨識流程大致分為五個步驟：影像輸入、前處理、特徵提取、模型訓練和結果輸出(圖一)；一般透過圖片、攝影機或其他影像獲取數據，針對輸入的影像進行影像增強、尺寸調整等前處理，以提高辨識準確性，再將影像進行特徵提取(如邊緣、顏色分布、紋理模式)，並透過深度學習模型(如CNN)進行訓練，最終輸出具體的識別結果，供使用者或應用程序進一步處理[9]。影像辨識主要應用於多項場景，如醫療影像分析、智慧城市、零售與電子商務、農業領域、自動駕駛技術、文化保護與創意應用等[10,11]。然影像辨識技術仍面臨一些挑戰，包括數據隱私保護、數據多樣性與模型偏差的改進，以及設備性能限制[12]。未來影像辨識可進一步與多模態技術(結合語音、文本、影像等數據)和生成式人工智慧(如生成對抗網絡，Generative Adversarial Network, GAN)融合，強化使用者與數位世界的互動[13]。

## 即時物件偵測與文字辨識技術於食品廣告之整合應用

本研究將透過YOLO[14]與OCR[15]結合應用，並針對實際違規廣告進行影像辨識及驗




**圖一、影像辨識流程**

證；應用兩項技術組合將大幅提升疑似違規廣告之辨識、分析效率與準確性，為影像相關巡查工作提供精準輔助。YOLO是一種即時物件偵測技術，能快速定位影像中多種類別的目標，例如交通標誌、車牌、廣告牌等，具備高準確性和高速處理的特點，非常適合應用於動態或大範圍場景中的違規廣告檢測，YOLO能協助使用者在即時影像中鎖定可能違規的目標，即使在複雜的街景或高密度廣告環境中也可以快速定位，大幅提升檢視影像的時間；另OCR主要能提取目標中的文字內容，於影像中提取文字資訊，能識別多語言、多字體，並具備處理不規則字體或手寫文字的能力。對於疑似違規廣告之巡查，OCR可從影像中提取文字內容，例如不當用詞等，提供明確依據。將YOLO與OCR技術結合後，能在疑似違規廣告巡查中的多個環節實現物件與文字辨識，提升作業效率與精準度，同時與法規、違規關鍵字等進行比對，檢視是否存在不當用詞或不實廣告用語等的情況。上述兩個技術的整合將可為未來食品等廣告巡查提供一個智慧化的方法，能協助提升巡查效率並降低人事成本，亦可輔助相關人員，集中精力於判斷其他細節，提高整體工作效率。

YOLO結合OCR應用實作需建立以下步驟：

## 一、環境建置

針對訓練環境需由下列條件建立：

1. 深度學習的訓練對於電腦顯示卡要求較高，若電腦沒有獨立顯示卡，無法使用GPU (Graphic Processing Unit)進行訓練，需用CPU進行訓練，效能將與GPU訓練有一定差異。
2. 應用YOLO模型並使用GPU進行訓練，安裝CUDA[16](使程式能直接調用GPU的計算能力)與cuDNN[17](專為深度學習設計的GPU加速資料庫，包含YOLO模型使用的常見操作，如卷積等)，可直接支援GPU加速運算，使YOLO模型在訓練時提升運算效能。
3. 建立一個虛擬環境及安裝套件pytorch[18](深度學習框架，主要用於神經網路建構與訓練)、ultralytics[14](YOLO模型工具，用於目標檢測及追蹤)、easyocr[19](光學識別(OCR)，用於文字識別)、open-cv[20](視覺與影像處理，用於圖像分析、物件追蹤及邊緣檢測)。

4.為進行物件偵測任務，需下載並使用YOLO系列模型作為基礎架構，依照實驗需求，選擇合適之模型版本(如YOLOv5、YOLOv8或YOLOv11)，模型將應用於食品廣告錄影影像之物件偵測流程，負責快速框選影像中潛在廣告或違規物件區域，作為後續文字辨識(OCR)之輸入依據。

5.透過爬蟲技術擷取電商平台之圖片300張，作為YOLO物件偵測模型訓練之基礎資料，此步驟有助於建立符合實際應用場景之資料集，提升模型於日後辨識影像中疑似違規內容之準確性與穩定性。

6.蒐集公開之食品違規廣告裁處案件及法規敘述、違規誇大字詞等，建立違規關鍵字辭共3,272個。

以上為本研究整體應用YOLO結合OCR所需之相關套件、模型及資料。

## 二、模型訓練

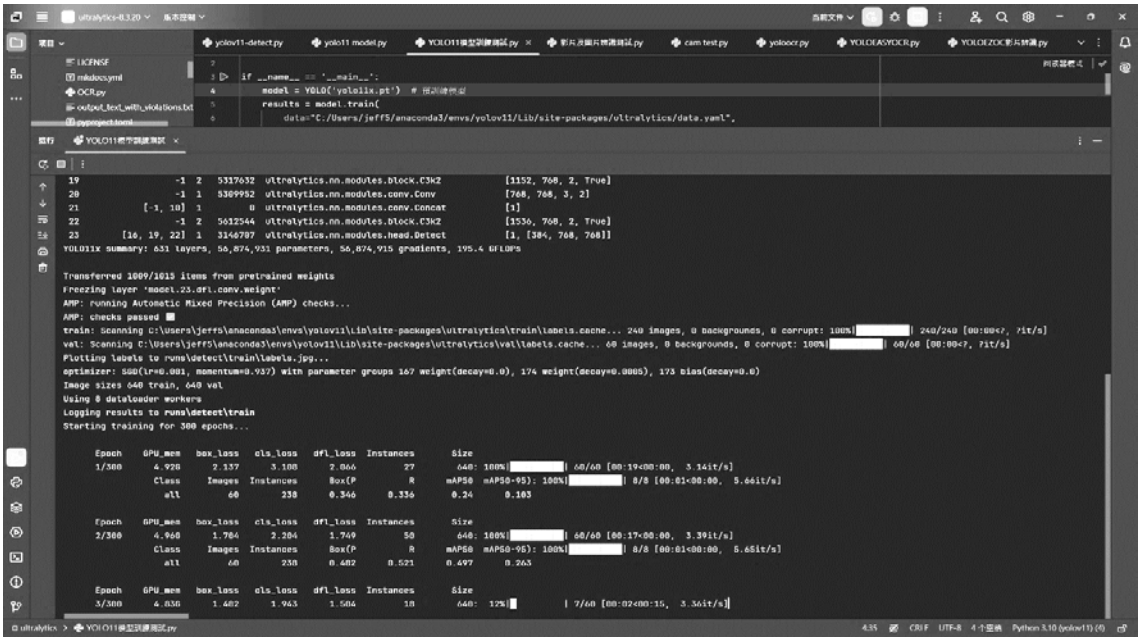
透過爬蟲方式收集電商平台、廣告等多樣化圖片並透過標註工具(MakeSense)[21]進行標註，生成YOLO格式的邊界框和類別標籤，並按照需求進行模型訓練，另透過數據增強技術(如旋轉、翻轉等)進一步提升模型泛化能力，並劃分合理比例的訓練集、驗證集和測試集，標註完整的數據集能；並應用官方提供的預訓練權重進行訓練，本研究使用YOLOv11x版本模型進行訓練，如圖二。

YOLO模型在訓練過程中的各種損失函數和評估指標(例如損失函數、精確度、召回率、mAP等)變化趨勢，幫助我們理解模型訓練整體表現情況[22]。在整體趨勢圖中，train代表訓練集，val代表驗證集；訓練集用於訓練模型，相當於「答案」；驗證集則用於評估模型的性能，相當於「考題」，但學得好不一定考得好，這主要取決於考題與學習內容的相關性，兩者的共同點在於都涉及loss (損

```python
from ultralytics import YOLO

if __name__ == '__main__':
    model = YOLO('yolo11x.pt')  # 預訓練模型
    results = model.train(
        data="C:/Users/jeff5/anaconda3/envs/yolov11/Lib/site-packages/ultralytics/data.yaml",
        epochs=300,  # 訓練回合數
        imgsz=640,  # 圖像尺寸
        device=0,  # 使用 GPU 訓練
        optimizer='SGD',  # 可選 Adam 或 SGD。SGD 通常有更好的效果
        workers=8,  # 加載數據數
        batch=4,  # 批次大小
        amp=True,  # 使用混合精度訓練
        lr0=0.001,  # 初始學習率
        lrf=0.1,  # 最終學習率
        momentum=0.937,  # 動量
        weight_decay=0.0005,  # 權重衰減，防止過擬合
        warmup_epochs=3.0,  # 過渡的回合數
        warmup_momentum=0.8,  # 過渡動量
        warmup_bias_lr=0.1,  # 過渡初始學習率
        cos_lr=True,  # 學習率調整
    )
```

圖二、應用yolov11x模型訓練圖片



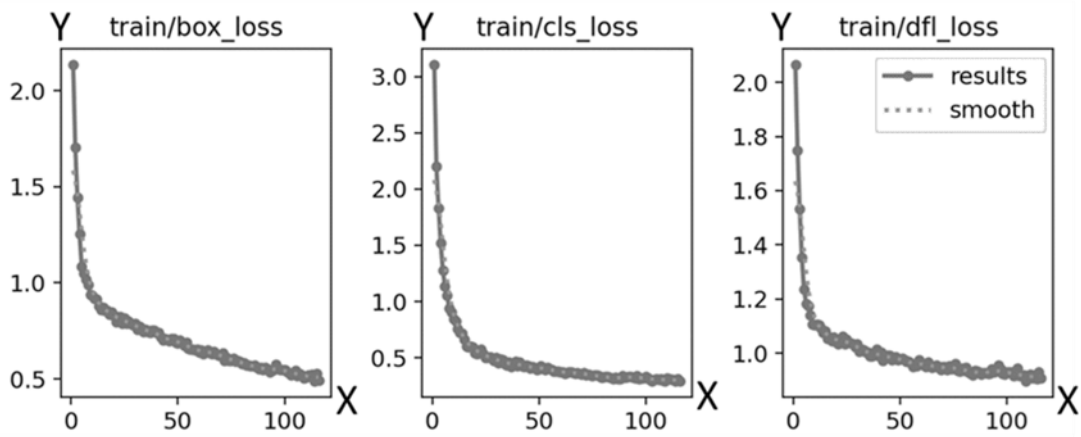

**圖二(續)、應用yolov11x模型訓練圖片**

失)[23]；在機器學習演算法中，損失是一個常見的概念，它反映了模型預測結果與真實答案之間的差距。例如，一輛汽車將100單位的燃油能量轉化為70單位的動能，損失了30%，若能實現100%能量轉化，損失即為0。同理，在機器學習中，若模型的預測結果與真實標籤完全一致，損失值為0，差距越大，損失值就越高，因此，損失函數的目標是盡可能減少損失值，從而提高模型的預測準確性[24]；在有監督的學習中，模型基於已標記的數據進行學習，類似於「打明牌」，問題和答案都為已知情形，模型通過對訓練集和驗證集數據進行推理(或預測)，並將其結果與正確答案進行比較，計算出損失值，損失值越小，表示模型的推理結果與正確答案的差異越小；當損失值為0時，表示模型的推理與答案完全一致，預測達到100%準確[23,24]。以下逐一介紹整體模型訓練指標：

㈠訓練損失(Training Loss)

由上圖三可見，本次訓練過程train系列的loss都是降低的，X軸表示訓練輪次，Y軸表示損失的值。這表示為有不錯的訓練成效。

1. train/box_loss為框損失(Box Loss)用於衡量模型對目標框(Bounding Box)定位的準確性。可看到圖三中在訓練過程中，box_loss從大約2.0快速下降至0.5，並逐漸趨於平穩，這表明模型對目標框的定位學習效果逐漸提升並達到穩定狀態。正常情況下，隨著訓練的進行，損失表現應該是要越降越低的，如是長期忽高忽低，或一直不明顯收斂，那就表示訓練可能出現問題。如果box_loss的損失不斷降低，而後持續穩定，表示訓練沒有明顯問題，可視情況減少或不再投入訓練資源。
2. train/cls_loss為分類損失(Classification Loss)用於評估模型在預測類別和真實




圖三、模型訓練損失

類別之間的差異。可看到圖三中在訓練過程中cls_loss初始值約為3.0，隨著訓練輪次的增加迅速下降至0.5，顯示模型分類誤差顯著減少，分類準確率提升。

3. train/dfl_loss為分佈式焦點損失(Distribution Focal Loss)用於提升邊界框回歸的精確性，特別是目標框的邊界位置。可看到圖三中在訓練過程中dfl_loss從2.0下降至1.0，說明模型對邊界框的學習變得更加精確。簡單說它是輔助box_loss，提供額外的資訊，透過對邊界框位置的機率分佈進行最佳化，進一步提高模型對邊界框位置的細化和準確度。

㈡驗證損失(Validation Loss)

從規範上講，驗證集和訓練集是永遠不見面的，這麼做是為了驗證模型是否真正學到了資料的特徵和精髓，而非是靠死背所見過的資料，也就是說模型在經過幾輪的訓練集資料學習後，將從未見過的驗證集資料，給出預測答案，並再去對照標準答案後，兩個答案的差異，就是驗證集的損失。由圖四可見相較於訓練集的平滑趨勢，驗證集似乎是有些反覆，這是一種常見現象，只要驗證集損失沒有顯著上升，整體趨勢正在變好，與訓練集損失的差距不是特別大，這一般是正常的。不過，可能要留意以下細節：1.樣本資料的變異：驗證集可能包含一些與訓練集不同風格的樣本，這會導致損失不穩定。例如拿著熊做辨識訓練，最後讓模型去認識熊貓，模型會有點迷糊無法準確判斷。2.模型的過擬合：如果驗證集的樣本資料正常，模型在訓練集上的損失表現很好，但驗證集表現不穩定，可能是模型記住了訓練集的細節，也就是過於死背，也就是過度擬合[25]。另如果遇到比較嚴重的問題，可以調整超參數(Hyperparameter)，例如調小學習率，或使用提前停止策略(Early Stopping)來防止過度擬合[26]，也可以調整批次大小(batch size)，增加一個批次數量，讓它見多識廣。同時，增加訓練資料量或使用資料增強技術，可以使模型更好地泛化(Generalization)，減少驗證損失的波動[27]。

1. val/box_loss為驗證集上的框損失，反應模型對未見數據的目標框定位能力。可

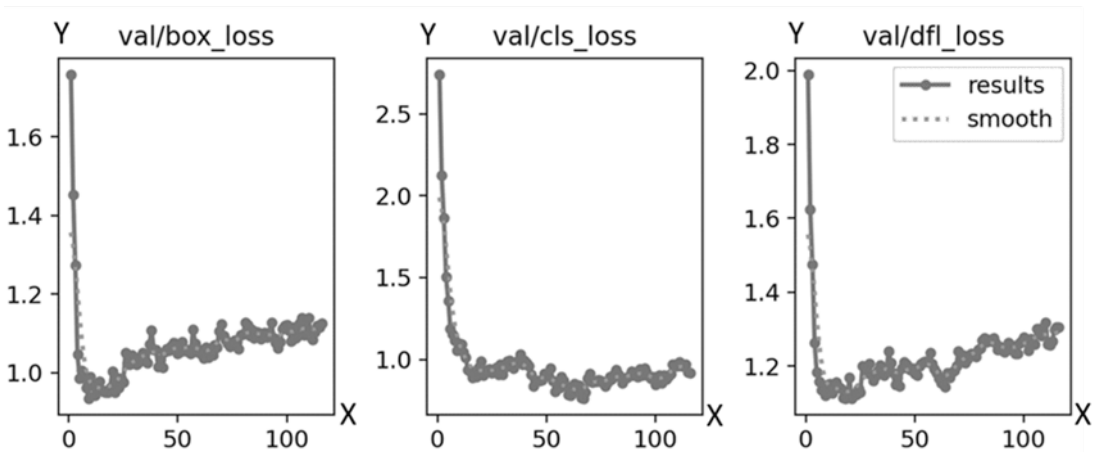
看到圖四中在驗證過程中val/box_loss從1.6下降至1.0，表明模型在驗證集上的框定位誤差減少，具備一定的泛化能力(Generalization Ability)。

2. val/cls_loss為驗證集上的分類損失，評估模型在未見數據上的分類能力。可看到圖四中在驗證過程中val/cls_loss從2.5降至1.0左右，顯示模型對驗證集目標分類的準確性不斷提高。
3. val/dfl_loss為驗證集上的分布式焦點損失，評估模型在未見數據上的目標框邊界處理能力。可看到圖四中在驗證過程中val/dfl_loss逐漸下降至1.2，顯示模型在驗證集上的邊界框處理能力與訓練集一致。

㈢性能指標(Performance Metrics)

在YOLO模型中，訓練目標是準確識別和定位影像中的物體，為了評估模型的表現，需透過量化指標來衡量其準確性，該指標通常於驗證集上計算，以了解模型在未知資料上的預測能力；性能指標(metrics)表示模型在驗證集上的評估指標，(B)在目標偵測任務中表示Bounding Box，也就是邊界框的偵測結果。

1. metrics/precision(B)為精確率(Precision)表示模型預測為正的樣本中，真正為正的比例，衡量模型檢測結果的準確性，可看到圖五中看到precision從約0.1開始迅速提升至0.8以上，表明模型能有效地減少誤檢(False Positive)。
2. metrics/recall(B)為召回率(Recall)表示實際正樣本中被模型正確檢測到的比例，衡量模型檢測的完整性，可看到圖五中看到recall從約0.1穩步增長至接近0.8，表示模型能夠檢測到大多數真實目標。
3. metrics/mAP50(B)為平均精度(Mean Average Precision, mAP)在IoU(Intersection over Union，重疊度)閾值為50%時的值，反映模型在單一閾值下的目標檢測效果。可看到圖五中看到mAP50從0.1開始快速增長至穩定在0.8左右，顯示模型的檢測準確性高。
4. metrics/mAP50-95(B)為平均精度(mAP)在多個 IoU 閾值(50%至95%)下的平均值，衡量模型對多種檢測條件的綜合能力。可看到圖五中看到mAP50-95從0.1增長至0.6，表明模型在不同目標尺度和位置下的表現良好；簡單來說mAP50



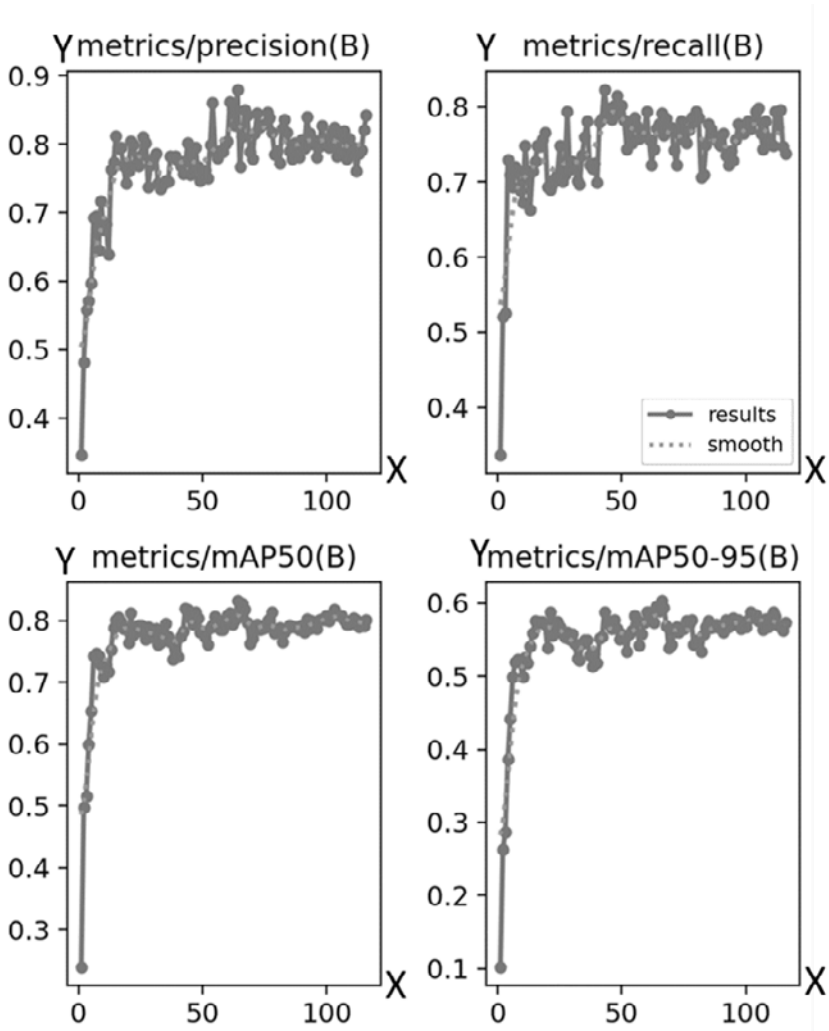

圖四、模型驗證損失

這個指標相對寬鬆，能夠展示模型在較低嚴格度下的整體表現，它更適用於那些對定位要求不是特別嚴格的應用場景；另mAP50-95則表示在嚴格的IoU條件下也能準確偵測和定位目標，適用於那些對定位要求較高的應用場景，如自動駕駛、醫療影像分析等。

綜合本次訓練指標所呈現之結果：訓練損失與驗證損失均快速下降並趨於穩定，表明模型逐漸收斂，性能提升；精確率與召回率(Precision&Recall)皆為0.8，表示模型對目標的檢測準確性和完整性顯著提升；mAP指標顯示模型在不同條件下的檢測性能穩定且良好。整體模型的檢測表現穩定，能夠有效完成目標檢測任務。

## 四、影像辨識程式實作

本次實作為結合YOLO目標檢測和




圖五、模型性能指標

EasyOCR光學文字辨識的應用，針對影片進行逐幀分析以辨識廣告中的物體與文字內容，並檢測是否存在違規詞語。分析過程 如圖六，以下逐一說明：

㈠匯入建立好的違規詞庫文字檔案，這些詞語用於判斷影片中是否出現需要注意的敏感內容，再透過EasyOCR模型，設置為支援繁體中文和英文的文字辨識，並載入事先訓練好的YOLO模型，並透過GPU加速運算，以顯著提高文字和目標檢測的效率。

㈡功能：在影片分析過程中，程式逐幀讀取影片畫面，利用YOLO模型檢測畫面中的物體並劃出邊界框(Bounding Boxes)，這些框用於鎖定畫面中需要進行辨識的區域，提取邊界框內的畫面(ROI，感興趣區域)，並將這些畫面輸入EasyOCR模型進行光學文字辨識，識別出每一框內的文字內容。

㈢詞句檢查與標記：每次從畫面中檢測到文字後，程式會將辨識到的文字與違規詞列表進行逐一比對，如文字內容中包含違規詞，程式會對該內容進行標記，添加「[疑似違規]」的標籤，以便使用者快速識別影片中的敏感或潛在問題內容。

㈣戳記與重複檢查：為了方便使用者追蹤問題，程式會為每段檢測到的文字附加一個精確的時間戳記，時間格式為「時:分:秒:毫秒」，對應影片中出現該文字的位置；另為了避免重複記錄相同的內容，程式設計了一個機制，檢查當前幀中辨識到的文字是否與上一幀相同，只有當檢測到新文字或內容變更時，才會將結果儲存下來，確保輸出結果的整潔性與精確性。

㈤儲存與即時顯示：所有的檢測結果，包括時間戳記、文字內容、違規標記，以及邊界框的座標位置，均會儲存到指定的文字檔案中，以便後續進行分析；另程式還會即時在影片畫面上繪製檢測框，並在框內標註文字內容，將分析結果以視覺化的形式展示於螢幕中，方便用戶檢視處理進度。

㈥應用場景：這段程式碼適用於廣告審核、影片合規性檢查、文字內容提取等多種場景，尤其在需要檢測敏感詞語或追蹤影片文字內容時，能有效提高處理效率，結合GPU加速與多步驟的自動化流程，該工具具有高效性和實用性，能顯著減少人工審核的工作量。

## 智慧影像辨識於食品廣告探索之應用實例

本研究透過食品廣告錄影影像進行測試，整體影像辨識結果顯示，於多數場景中可穩定且準確地判別影像中之疑似違規內容，展現良好之辨識效能；測試過程中，應用YOLO模型進行物件定位，能迅速偵測並框選影像中物件之位置；同時結合光學字元辨識(OCR)技術，擷取框選區域內之文字資訊，並與建立之違規辭庫進行比對，以識別疑似違規特徵；即便在光線變化劇烈、不規則字體呈現或背景複雜等條件下，仍可維持高辨識精度及穩定性，顯示其具備良好之場景適應性，未來可作為巡查作業之輔助工具。影像辨識之具體結果詳見圖七至圖九。

## 影像辨識應用之限制與展望

儘管在影像辨識過程中展現了穩定性與高準確性，但仍存在一些需要克服的限制。主要對於光線條件極端的影像，例如在背景過於複雜或干擾物過多的情況下，可能出現誤判，將非廣告物件錯誤標記或將無關文字辨識為重點內容；OCR在面對字體高度變形、不規則排版

```python
import cv2
from ultralytics import YOLO
import easyocr

# 讀取違規詞列表
restricted_phrases = []
with open("廣告違規詞.txt", "r", encoding="utf-8") as f:
    restricted_phrases = [line.strip() for line in f if line.strip()]

# 初始化 EasyOCR 和 YOLO 模型，並啟用 GPU
reader = easyocr.Reader(lang_list=['ch_tra', 'en'], gpu=True)
model = YOLO('C:/Users/jeff5/anaconda3/envs/yolov11/Lib/site-packages/ultralytics/runs/detect/train3/weights/best.pt')
model.to("cuda")

# 設定影片路徑
video_path = "C:/Users/jeff5/anaconda3/envs/yolov11/Lib/site-packages/ultralytics/113TP3034CUT.mp4"
cap = cv2.VideoCapture(video_path)

if not cap.isOpened():
    print("無法開啟影片")
    exit()

# 建立儲存結果的文字檔
text_output_path = "output_text_with_violations.txt"
previous_texts = {}

with open(text_output_path, "w", encoding="utf-8") as text_file:
    while cap.isOpened():
        ret, frame = cap.read()
        if not ret:
            print("影片結束")
            break

        # 使用 YOLO 模型進行物體偵測
        results = model.predict(frame, save=False, device="cuda")
        annotated_frame = results[0].plot()
        # OCR 文字辨識與違規詞檢查
        current_texts = {}
        for i, box in enumerate(results[0].boxes):
            x1, y1, x2, y2 = map(int, box.xyxy[0])
            roi = frame[y1:y2, x1:x2]
            result = reader.readtext(roi)
            text = " ".join([res[1] for res in result])

            # 檢查是否包含違規詞
            is_violation = any(phrase in text for phrase in restricted_phrases)

            # 若偵測到新文字且與上一幀不同，才進行紀錄
            if previous_texts.get(i) != text and text not in current_texts.values():
                if is_violation:
                    text = f"[疑似違規] {text}"

                # 獲取時間戳記並轉換為時:分:秒格式
                total_seconds = cap.get(cv2.CAP_PROP_POS_MSEC) / 1000
                hours = int(total_seconds // 3600)
                minutes = int((total_seconds % 3600) // 60)
                seconds = total_seconds % 60
                timestamp = f"{hours:02}:{minutes:02}:{seconds:06.3f}"
                # 儲存結果
                text_file.write(f"時間 {timestamp}, 區域 [{x1}, {y1}, {x2}, {y2}]: {text}\n")
                print("識別到的文字：", text)
            # 更新目前幀的文字
            current_texts[i] = text
        # 更新上一幀的文字
        previous_texts = current_texts
        # 顯示畫面（可選）
        cv2.imshow(winname='Video Detection with OCR', annotated_frame)
        if cv2.waitKey(1) & 0xFF == 27:
            print("檢測結束")
            break
cap.release()
cv2.destroyAllWindows()
print(f"處理完成，文字結果保存於 {text_output_path}")
```

圖六、YOLO結合OCR影像辨識


圖七、食品廣告影像辨識結果


圖八、食品廣告影像辨識結果


**圖九、食品廣告影像辨識結果**

或手寫體時，識別準確率可能降低，尤其是在廣告字體設計特殊或創意形式極具挑戰時，模型可能無法完整提取文字資訊，低解析度或模糊影像更是另一挑戰，特別是當影像來自遠距離拍攝或影片中快速移動的物件時，辨識結果可能受到影響，增加漏判與誤判的風險。

在應用層面，對規範和法規的依賴也可能成為限制因素，如違規辭庫未能即時更新或未涵蓋之特殊規範等，可能無法準確判斷違規內容，對於新型廣告形式的識別，也可能需進行額外訓練與優化，才能適應更多元的應用場景。

另本研究應用影片作為主要辨識來源，其硬體效能也是一大挑戰，若未使用GPU進行運算，影片處理的效能及即時性可能大幅下降，單靠CPU的單純運算能力在處理大量影像數據時可能顯得不足，因此在未來的應用中，需將硬體配置(如GPU的應用)納入考量，以確保系統能在高效能需求的情境下穩定運作。

綜合測試結果，在實際操作中建議仍須有人工介入監督與輔助，尤其於疑似違規內容之判定階段，應由專業人員進行確認，以降低因模型技術誤差而產生誤判或遺漏的風險，儘管YOLO與OCR技術展現出良好的影像辨識效能，在光線變化劇烈、不規則字體或背景複雜等挑戰條件下仍具備一定的適應性，然於多變的實際應用環境中，純技術辨識仍存在一定侷限，需透過持續優化及人機協作方式以進一步提升整體之準確性與可靠性。

未來研究可聚焦於技術優化與人機互動策略之結合，並將模型訓練延伸至新型態廣告表現手法(如動態特效字幕、疊圖式標語、轉場文字等)，也將有助於更全面的在實際應用中，實現更高層次之影像辨識自動化與穩定

性；特別是在食品、醫療等對廣告規範要求嚴格的領域，期望透過本技術作為輔助工具，達到「先自動偵測、再人工複核」的流程優化，有效提升違規廣告判讀之準確性與時效性，並可進一步整合資訊來源管理、時間標記與違規紀錄等回報機制，建立完整的智慧巡查模式，協助於大量網路與電視廣告中，自動化監控、通報與追蹤潛在違規內容，提升整體巡查能量與覆蓋範圍。若持續透過資料擴增、模型精緻化及跨模組整合等方式，強化技術穩定性與場景適應性，並搭配制度面與法規面的即時更新，可促進智慧辨識技術於實務場景中的深化應用與制度化推動。

## 總結與建議

本次研究透過食品廣告錄影影像進行測試，整體影像辨識結果顯示，於多數場景中能穩定且準確地判別疑似違規內容，展現出良好之辨識效能。

在測試過程中，YOLO模型能快速定位目標區域，OCR技術則針對精確區域提取文字，兩者結合可降低背景干擾，提升運算效率與辨識精度，特別適用於場景複雜或資訊密集之影像辨識任務，另再透過蒐集公開之違規食品違規廣告裁處案件及違規法條敘述、違規誇大字詞等建立之辭庫導入比對，能有效成功識別出疑似違規用詞等特徵；即使面對光線變化、不規則字體及背景複雜之影像條件，仍維持較高之辨識精度與穩定性，展現出良好之場景適應性，顯示未來應用於巡查疑似違規訊息及影片等領域頗具之潛力。

然本次測試亦發現影像辨識技術目前仍存在一些限制，例如於背景過於複雜影像、有特殊字體或快速移動及低解析度影像條件下中，可能無法穩定提取關鍵細節，物件定位及文字提取之準確性可能受影響，導致部分內容無法完整或正確識別。

在實際應用中，影像品質及提升對特殊字體和複雜影像的適應能力將成為影響辨識效能之關鍵因素，未來除針對特殊字體辨識、低品質影像處理及異常條件下之辨識精確度進行模型優化並於影像辨識流程中納入前處理與品質評估機制外，持續蒐整違規廣告樣態之資料及更新機制，強化對新興廣告形式及潛在違規語句之辨識能力，同時亦可導入多模態數據(如語音轉文字、字幕辨識等)結合訓練架構，將有助於提升整體偵測廣度與精準度。

本次測試顯示導入影像辨識技術在巡查工作中具有良好的輔助作用，但同時仍需考量硬體條件等實際問題，未來如有挹注其經費，完善硬體設備、升級(如應用更高效的GPU)及技術優化等，預期可進一步拓展於更廣泛應用場景；透過人機協作機制與回饋學習設計，可逐步建立更具自我修正及實用性之模型，以因應日益多元與複雜的廣告呈現手法，並增強實務運作流程中之應用價值與長期發展潛力。

## 參考文獻

1. LeCun, Y., Bengio, Y. and Hinton, G. 2015. Deep learning. Nature, 521(7553), 436-444.
2. Krizhevsky, A., Sutskever, I. and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.
3. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D. *et al.* (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.
4. Shorten, C. and Khoshgoftaar, T. M. 2019. A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 1-48.
5. Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. 2016. You only look once: Unified,

real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
6. Ren, S., He, K., Girshick, R. and Sun, J. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in Neural Information Processing Systems, 28.
7. Ronneberger, O., Fischer, P. and Brox, T. 2015. U-Net: Convolutional networks for biomedical image segmentation. International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 234-241). Springer.
8. Badrinarayanan, V., Kendall, A. and Cipolla, R. 2017. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12), 2481-2495.
9. Szeliski, R. 2022. Computer vision: Algorithms and applications (2nd ed.). Springer.
10. Esteva, A., Kuprel, B., Novoa, R. A., Ko, J. *et al.* (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115–118.
11. He, K., Gkioxari, G., Dollár, P., Girshick, R. *et al.* (2017). Mask R-CNN. Proceedings of the IEEE International Conference on Computer Vision, 2961–2969.
12. Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M. *et al.* (2018). ImageNet-trained CNNs are biased towards texture. arXiv preprint arXiv:1811.12231.
13. Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B. *et al.* (2014). Generative adversarial nets. Advances in Neural Information Processing Systems, 27.
14. UltralyticsYOLOv11。[https://docs.ultralytics.com/models/yolo11/]
15. Memon, J., Sami, M. and Khan, R. A. 2020. Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR). *arXiv preprint*, *arXiv:2001.00139*.
16. NVIDIA CUDA。[https://developer.nvidia.com/cuda-toolkit]。
17. NVIDIA cuDNN。[https://developer.nvidia.com/cudnn]。
18. Pytorch。[https://pytorch.org/]
19. EasyOCR。[https://github.com/JaidedAI/EasyOCR]
20. Opencv。[https://opencv.org/]
21. Makesense。[https://www.makesense.ai/]
22. Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779–788.
23. Goodfellow, I., Bengio, Y. and Courville, A. 2016. Deep Learning. MIT Press.
24. Géron, A. 2019. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems (2nd ed.). O'Reilly Media.
25. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. *et al.* 2014. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929-1958.
26. Prechelt, L. 1998. Early Stopping — But When? In Neural Networks: Tricks of the Trade (pp. 55-69). Springer.
27. Shorten, C. and Khoshgoftaar, T. M. 2019. A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 1-48.

# An Exploratory Study on the Application of Image Recognition Technology in Food Advertising

YUE-HAN KUO, MENG-YING WU AND CHAO-YI WANG

Decision Support Center, TFDA , MOHW

## ABSTRACT

Image recognition technology represents a significant advancement in the fields of artificial intelligence and computer vision. In particular, the integration of object detection technology—YOLO (You Only Look Once)—with Optical Character Recognition (OCR) demonstrates broad application potential. This study utilized video footage of food advertisements to evaluate the performance of these technologies. Results showed that the constructed model using the aforementioned technologies could reliably and accurately detect suspected non-compliant content in most scenarios, demonstrating strong recognition performance. During testing, the YOLO model rapidly localized target regions, while the OCR component extracted textual content from specific areas. The combination of these two technologies reduced background interference and improved both computational efficiency and recognition accuracy. This made the system especially suitable for complex or information-dense visual environments. Additionally, by incorporating a lexicon constructed from publicly available food advertising violation cases, relevant legal descriptions, and exaggerated promotional terms, the system was able to effectively identify suspicious or non-compliant language, demonstrating strong adaptability across various visual scenarios. This suggests its promising potential for applications in online monitoring of suspected violations in both textual and video content. However, several limitations were observed in this study. Recognition accuracy declined under extremely complex backgrounds or poor light intensity conditions. The system also struggled to identify proper nouns and special fonts when dealing with fast-moving objects or low-resolution images. In particular, images featuring highly creative or non-standard typography affected recognition stability. Future development should focus on optimizing models to enhance adaptability to special fonts and complex backgrounds, while also ensuring consistent input of image quality. Overall, this study validates the feasibility of applying image recognition technology to enhance the efficiency of video-based inspection processes. However, practical implementation must also consider hardware computational requirements. It is recommended that high-performance hardware (e.g., GPUs) be deployed alongside further technical optimization to expand application scenarios and achieve more efficient, accurate, and robust image recognition capabilities, meeting the demands of diverse environments.

Key words: Image Recognition, Food Safety Management, Food Advertising, OCR, YOLO