教育科技與學習 7 : 2 (2019,7)：153-179

研究論文

以卷積神經網路實現口罩偵測

李其恩

博士研究生
國立中央大學企業管理學系
E-mail: rickyyenli@gmail.com

傅思齊

研究生
國立臺北科技大學資訊工程系
E-mail: es891010@gmail.com

林琛絜

研究生
國立臺北教育大學資訊科學系
E-mail: ckdcshadow@gmail.com

摘要

新冠肺炎病毒極高的傳播率使全球各國醫療資源供不應求，為了避免群聚感染而實施隔離，更對經濟、運輸、教育等方面都造成嚴重影響，疫情爆發至今仍未見擴散情況被控制，可預見防疫將是一項需長期進行且不容疏忽的日常工作。

鑒於戴口罩為目前行之有效的防疫方法，而當前的臉部偵測模型對於遮蔽了半張臉並戴著口罩的人臉成效不彰，且未依正確方法配戴口罩的行人偶爾可見，有潛在傳播疫情的可能性。

本研究將建立一具有三種標註之人臉數據集，並結合多種深度學習卷積神經網路架構與方法，設計可快速訓練和偵測出有戴、未戴與未戴好口罩的人臉偵測模型，希望能為防疫貢獻一份心力。

我們使用自適應算法調整圖像尺寸以減少不必要的操作，並修改 CIOU_LOSS 誤差函數以加快執行速度。實驗證實了我們的方法與相同精度的 YOLO v5m 相比，節省 70% 的時間。

關鍵詞：新冠肺炎、卷積神經網路、臉部偵測、遮蔽

壹、緒論

隨著科學技術的發展，機器學習在各領域有了長足的進展與多方面的應用，如資料探勘、搜尋引擎、自然語言的處理還有圖像影像及音訊的偵測與辨識等等，都有著許多技術上的突破，其中卷積神經網路（Convolutional neural network, CNN）的貢獻可謂功不可沒（Lecun et al., 1998）。

2019 年 11 月，一種全新冠狀病毒於中國武漢被首先發現，並導致嚴重特殊傳染性肺炎（COVID-19）疫情爆發。世界衛生組織（World Health Organization, WHO）宣布它為致命疾病。疫情構成全球大流行後，各國防疫宣導、規定、政策甚至封鎖措施紛紛出爐，如進出公務或非公務機關採實聯制、保持社交距離、入境隔離措施、出入高感染傳播風險場域需量體溫、戴口罩等等。隨著口罩被證明能確實減少飛沫和氣融膠傳播病毒的數量後（Howard et al., 2020），成為人們出門必備的物品，如不戴口罩連大眾運輸工具都無法搭乘，因為戴口罩而造成生活不便的情況也開始浮現，如呼吸不順暢、眼鏡起霧、部分具備人臉識別功能的設備因口罩的遮擋失靈等等。另外，研究者觀察行人發現部分民眾因各種原因雖有戴口罩卻並未依正確方式配戴，如口罩未攤開拉至鼻樑及下巴、鼻樑片未壓至與鼻樑貼合等，導致其口罩防護能力下降，具潛在染疫與傳播風險。

人臉識別技術被廣泛應用於如手機解鎖、監控系統、門禁管理、智慧零售等項目，但防疫方面目前多以紅外線熱像儀感應人體溫度，輔以人工方式判斷口罩是否配戴及是否正確配戴，於人潮眾多處難以一一發現及提醒，尤其在不得已必須與人群接觸的環境中，正確地配戴口罩成為非常重要之保護自身及他人的方法。在影像處理領域，戴口罩的臉部偵測對於當前的臉部偵測模型具有相當大的挑戰性，因為口罩遮蔽了部分的臉部特徵，且口罩的種類、款式與花色繁多，還有訓練樣本過少，都對現行的臉部偵測模型造成一定的阻礙（Wang et al., 2020）。

CNN 是目前深度神經網路的發展主力，在影像辨別方面卓有成效。隨著資訊設備及電腦性能的提升，常用的目標檢測方法通常可分為兩類：一階段網路（One-Stage）以 YOLO 系列網路為代表，雖然具有很高的實時性，但檢測精度有待提高；二階段網路（Two-Stage）以 faster-RCNN 為代表，存在的實時性差、模型規模大等問題。

在 COVID-19 大流行爆發期間，很少有研究人員（Roy et al., 2020; Loey et al. 2021）利用 YOLO v2、YOLO v3 和 tiny YOLO v3 來執行口罩檢測任務。而

YOLO v4 與 YOLO-V5 相比，YOLO-V4 檢測的準確率更高。YOLO-V5 主要側重於速度提升（Zhang et al., 2021）。Kumar（2021）也提到 YOLO 系列算法的一項缺點是無法檢測小物體，因此檢測人臉上的面罩目標是一項具有挑戰性的任務。遵循類似的策略，本研究採用了 YOLO 的變體，結合多種深度學習方法，以 YOLO v5 作為提取特徵的卷積神經網路架構，克服目前因口罩遮擋人臉之特徵點的困難，建立人臉數據集，並正確判斷目標是否配戴口罩以及口罩是否以正確方式配戴的人臉偵測模型，在常見解析度影片中維持一定的偵測速度與準確度，期望能對防疫工作做出貢獻。

貳、文獻探討

一、CNN

（一）簡介

卷積神經網路（Convolutional Neural Network, CNN）是目前深度神經網路（Deep Neural network）領域的發展主力，在影像辨別方面卓有成效，以模仿人類大腦的認知方式所建立而成，現行的許多影像辨識模型都基於卷積神經網路的架構做延伸，具有自動學習、歸納特徵的特性，解決了原先因影像所需要處理的數據量過大以及影像因數位化難以保留特徵的問題。

（二）架構

典型的卷積神經網路由卷積層、池化層與全連接層構成。卷積層將原始影像與特徵檢測濾波器做卷積運算，負責提取影像的局部特徵，其中特徵檢測濾波器隨機產生若干種；池化層用來降低維數，減少像素數量與網路計算的次數且保留特徵的關鍵訊息，目的為縮短訓練時間並防止過擬合；最後全連接層配合權重輸出預測的結果。

二、Faster-RCNN

（一）簡介

Region-based Convolutional Network method（R-CNN）先使用選擇性搜尋（Selective Search）預先篩選出約 2,000 個可能包含重要特徵的區域（region proposal），再將這些區域壓縮放入 CNN 提取特徵並分類提高準確性（Girshick et

al., 2014）。

Fast Region-based Convolutional Network method（Fast R-CNN）則將 R-CNN 中重複運算 CNN 的部分縮減至一次，再將擷取的特徵讓 2,000 個 region proposal 運用，再使用 Region of Interest Pooling（ROIpooling）對應到 Feature map 上，解決了 R-CNN 運算較慢的問題（Girshick, 2015）。

Faster Region-based Convolutional Network method（Faster R-CNN）使用 Region Proposal Network（RPN）來生成 region proposal，並整合 bounding box 與 regression 技術，大幅優化了 Fast R-CNN 的效能（Ren et al., 2017）。

（二）架構

Faster R-CNN 的架構大致可分為四個部分：卷積層（conv layers）在預處理時也會記錄特徵資訊給後面的池化層、Region Proposal Network 部分使用 RPN 生成 region proposals 的同時也用 bbox regression 校正 anchor 的位置、ROIPooling 整合 Feature Map 及 RPN 的資訊、最後的分類（Classification）則用 Softmax 函式計算被提出的區塊（Proposals）屬於哪個類別，並再次使用 bbox regression 技術取得更準確的區塊。

三、YOLO v1-v3

（一）簡介

相對於 Faster R-CNN 等 Two-stage 類型演算法將物件的位置偵測與分類分開進行，You Only Look Once（YOLO）演算法將物體偵測轉化為一個 regression 問題考慮（Redmon et al., 2016），開創了 One-stage 類型演算法的先河，大幅縮減了偵測所需時間，達到能即時偵測的程度。

YOLO 的主要思路是將影像切分成數個網格（grid），每格做兩個 bounding boxes 預測其屬於哪個類別，用卷積來判斷該格裡是否有物件的中心，並且同時也輸出每個 bounding box 屬於某物件的機率，最後利用 Non-Maximum Suppression（NMS）演算法選出最佳預測框，達成偵測影像中物件位置與類型的效果，但只能處理每個網格中最多只有一個物件的情況，且對影像中較小的物件偵測效果不彰。

YOLO 的作者在發表後又提出了 YOLO v2（Redmon & Farhadi, 2017），為了提高偵測定位的準確性，YOLO v2 將數據集的預訓練分為兩個步驟，先用較低解析度影像進行訓練，再改為輸入較高解析度影像繼續訓練，使得預訓練模型能夠適應高解析度影像，並且對影像資料多做了歸一化等預處理，另外借鑒 Faster R-CNN 中使用卷積層與 RPN 來預測 Anchor Box 的思路，作者使用 5 個大

小形狀不同的 Anchor Box 對物件邊框做偏移的預測來取代原先的全連接層，減少運算量的同時，達到可以在同一網格中進行多目標偵測的目的，以及保持運算速度的情況下對準確度的提升作出改進，並且在此基礎下提出一個可偵測約9,000類物件的偵測系統 YOLO9000，對影像中所佔比例較小的物件偵測雖有改善但仍有較容易忽略的情況。

YOLO v3 在 YOLO v2 的基礎上更進一步（Redmon & Farhadi, 2018），提出 Darknet-53 全卷積網路，捨棄池化層改用卷積中的 stride 降維，加入 Resnet 網路（Residual Network）解決加深網路層容易讓梯度消失或爆炸的問題，利用 feature map 做檢測並且設置了不同尺度的 bounding box 來對應不同大小的物件，另外 YOLO v3 把只能做單 label 多分類的 softmax 函數改為多個 logistic regression 分類器對 anchor 評分，可以對多個 label 進行多分類預測。

（二） 架構

目標偵測模型通常由輸入一張影像（Input）、預訓練骨架（Backbone）、用來提取不同層級的特徵圖層（Neck）還有用來預測對象類別和 bounding box 的檢測器 Head 所組成。YOLO v1 輸入為一張 448x448 的影像，經過多次卷積與池化層後由兩個全連接層輸出 7x7x30 的數據，其中 7x7 為網格大小，30 為兩個 bounding box 的預測座標及屬於各類別的機率。

四、YOLO v4

（一） 簡介

YOLO v4 結合了許多先進演算法的技術（Bochkovskiy & Liao, 2020），並將他們分別歸類為訓練時使用的 Bag-of-Freebies（BoF）和設計、測試神經網路時用的 Bag-of-Specials（BoS），在 YOLO v3 各方面都做了相當程度的改進，性能大幅提升的同時對硬體的需求也有所降低，在 MS COCO 數據集上與 YOLO v3 相比提高了 10% 的 AP 及 12% 的 FPS，是 YOLO 系列的一次重大更新。

（二） 架構

YOLO v4 的架構由輸入、Backbone、Neck、Prediction 組成。

1. 輸入

輸入想要偵測物件的影像或影片，統一縮放成 608x608x3 大小。

2. Backbone

YOLO v4 使用 CSPDarknet53 作為訓練的神經網路，CSPDarknet53 是用少量組的卷積將 Darknet53 與跨階段局部網路（Cross Stage Partial Network, CSPNet）組合（Wang et al., 2020），利用 CSPNet 獲取更多梯度融合信息降低計算量的特點來縮短預測時間，圖 1 為 CSPNet 的結構。

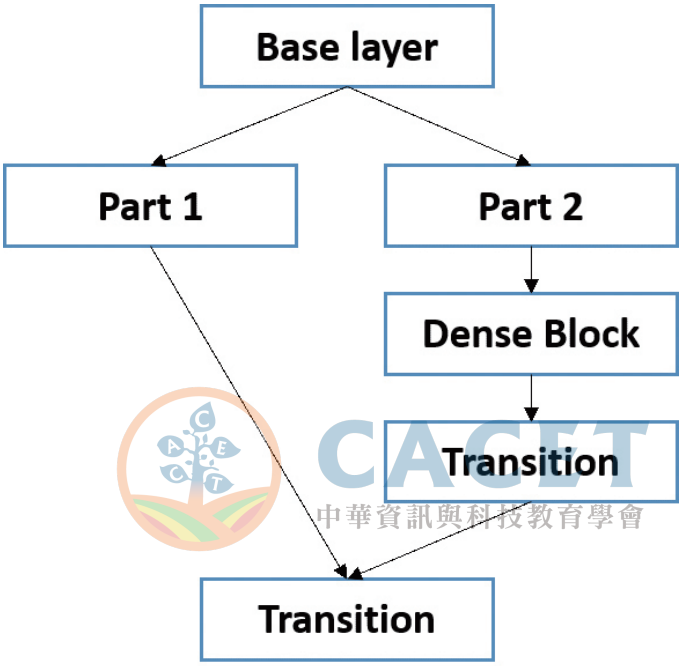

圖 1 CSPNet 的結構

YOLO v4 在數據增強方面使用 CutMix 之外也提出了 Mosaic 方法（Yun et al., 2019），使用四張影像以隨機裁剪、縮放、排放的方式拼接成新的訓練影像的方式，對於訓練的數據集做了增加訓練樣本變異性的處理，也藉此增加數據集中小目標的數量，另一個使用在 Backbone 中的 BoF 方法是 DropBlock，隨機將訓練影像的局部區域刪除，使神經網路不會只集中學習特定的某幾個特徵以降低過擬合（Overfitting）的問題（Ghiasi, Lin, & Le, 2018），最後是使用 Class label smoothing 正則化方法，降低正確類別的權重提高錯誤類別的權重，也是一種抑制 overfitting 的方法，Label smoothing 的公式如下：

$$y_n^{LS} = y_n(1-\propto)+\propto/N \qquad (2.1)$$

其中 N 為類別數，$\propto$ 在 y_n 為正確類別時為 1，錯誤為 0。

YOLO v4 作者在測試實驗時發現在 Backbone 中採用 Mish 這個連續可微的非單調激勵函數能提升準確度（Misra, 2019），更好的穩定神經網路梯度流，具有更好的泛化能力，其公式如下：

$$f(x) = x \tan h\left(\ln\left(1 + e^{x}\right)\right) \quad (2.2)$$

3. Neck

Neck 部分的主要目的是擴張感受野（Receptive Field），運用了如 Spatial Pyramid Pooling（SPP）技術使用同一影像做不同尺寸的池化後拼接（concat）成一張尺度不變的 feature map 進行特徵融合的方法（He et al., 2015），和 Feature Pyramid Network （FPN）技術（Lin et al., 2017），與 Path Aggregation Network （PAN）技術（Liu et al., 2018）結合提高提取特徵的能力，另外將原本 PAN 結構中直接相加的部分改為拼接，雖使通道數變多運算需求提高但效果也較佳。

4. Prediction

最後是對影像特徵做預測並輸出分類機率的部分，此前多數模型採用均方誤差（Mean square error, MSE）損失函數，而 YOLO v4 在比較 IOU_Loss、GIOU_Loss（Rezatofighi et al.,2019）、DIOU_Loss 和 CIOU_Loss（Zheng et al., 2020）等損失函數之後，選擇使用更為重視預測框寬高比例的 CIOU_Loss 作為神經網路的損失函數。

五、YOLO v5

（一） 簡介

YOLO v5 是仍在開發中的目標偵測模型，其架構與 YOLO v4 頗為相似，兩者優劣仍有許多爭議，YOLO v5 使用 pytorch 框架，相較於 YOLO v4 的 Darknet 框架更易於使用，並且支援更多元的使用方式，如對單張影像、多張影像、影片、視訊鏡頭與手機鏡頭等進行偵測。

（二） 架構

目前 YOLO v5 提供四種神經網路模型結構，由大到小分別為 YOLO v5x、YOLO v5l、YOLO v5m、YOLO v5s，主要差別在於各結構的深度與寬度。圖 2 為 YOLO v5s 架構。

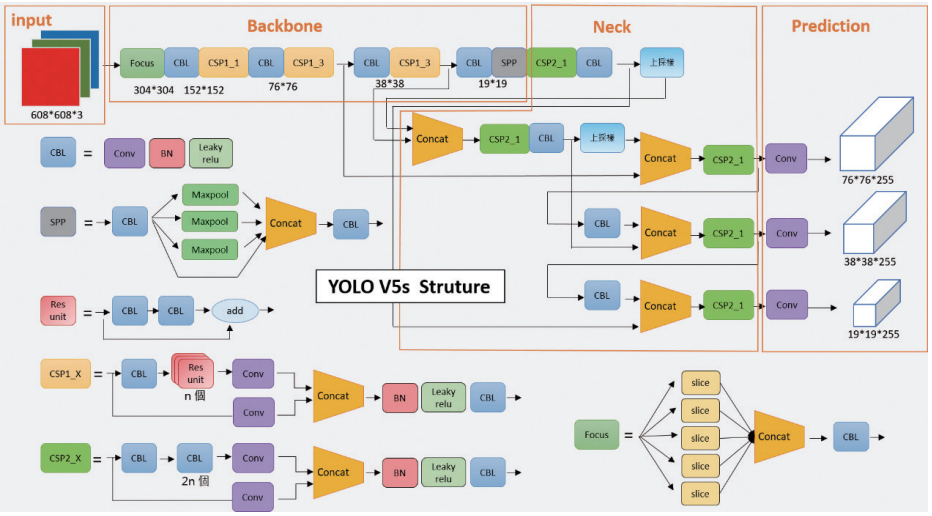

圖 2 YOLO v5s 架構

六、人臉偵測模型

隨著 GPU 和高端計算資源的出現，常用的目標檢測方法主要為 spatial pyramid pooling（He et al. 2014）、R-CNN（Girshick et al. 2014）、SSD（Liu et al. 2016）和 YOLO 系列（Redmon et al. 2016; Redmon & Farhadi 2017, 2018; Bochkovskiy et al. 2020）等基於深度神經網路啟發的卷積神經網路。

Alexey Bochkovskiy 於 2020 年 4 月發表論文並進行了 YOLO 系列的開發，獲得了 YOLO 的正式批准。在 YOLOV4 的熱度還在繼續的情況下，Ultralytics LLC 團隊於 5 月發布 YOLOV5。相比其他 YOLO 系列，YOLOV5 在 Tesla P100 快速檢測上可以達到 140 FPS，YOLOv4 只有 50 FPS。同時，YOLOV5 的大小只有 27 MB，而使用 Darknet 架構的 YOLOv4 的大小是 244 MB（Yang et al., 2020）。

YOLO v4 目標檢測器通過結合特徵提取和檢測層，使用 back-to-back 架構對對象進行定位和分類。與其他算法相比，YOLO v4 網路檢測速度最快，準確解決了回歸問題。與 YOLO-V5 相比，YOLO-V4 的準確率更高。YOLO-V5 主要側重於速度提升（Zhang et al., 2021）。YOLO 系列算法的缺點是無法檢測小物體，檢測人臉上的面罩目標是一項具有挑戰性的任務（Kumar et al., 2021）。

Wang（2020）對 YOLO-V5 算法進行了改進，然後使用 K-means++ 算法進

行 anchor 維度聚類，確定 anchor 參數，並將 CIoU 和 diounms 應用於 YOLO-V5 網路。Wang（2021）等人提出了一種基於改進的 YOLO-V4-tiny 的輕量級網路算法，增加了最大模塊結構以獲得更多目標的主要特徵，提高了檢測精度。提出了一種自下而上的多尺度融合；結合低層信息豐富了網路的特徵層次，提高了特徵利用率，並使用 CIoU 作為框架回歸損失函數，加快模型收斂。

本研究採用了 YOLO 的變體，以 YOLO v5 作為提取特徵的卷積神經網路架構，建立具有配戴口罩、沒有配戴口罩、未正確配戴口罩三種類別之人臉數據集，設計可以快速偵測及訓練的神經網路架構，並正確地判斷目標是否配戴口罩以及口罩是否以正確方式配戴的人臉偵測模型，在常見解析度影片中維持一定的偵測速度與準確度。期望能對防疫工作做出貢獻。

參、研究方法

一、神經網路架構

YOLO v5 提供比 YOLO v4 較小架構的神經網路，符合追求實時偵測的本研究需求，因此我們決定採用 YOLO v5 作為我們提取特徵的卷積神經網路架構，並比較兩者的差異，選擇針對我們自己蒐集、標籤的數據集而言，較適合的方式來構建神經網路，並且調整其中的參數以提升本研究的準確度與辨識速度。

我們的神經網路架構如圖 3 所示，在目前 YOLO v5 提供的四種結構中，我們選擇最小也最快的 YOLO v5s 進行改進，紅框處為我們改進的主要部分，此節將說明其中差異與我們使用的方法。

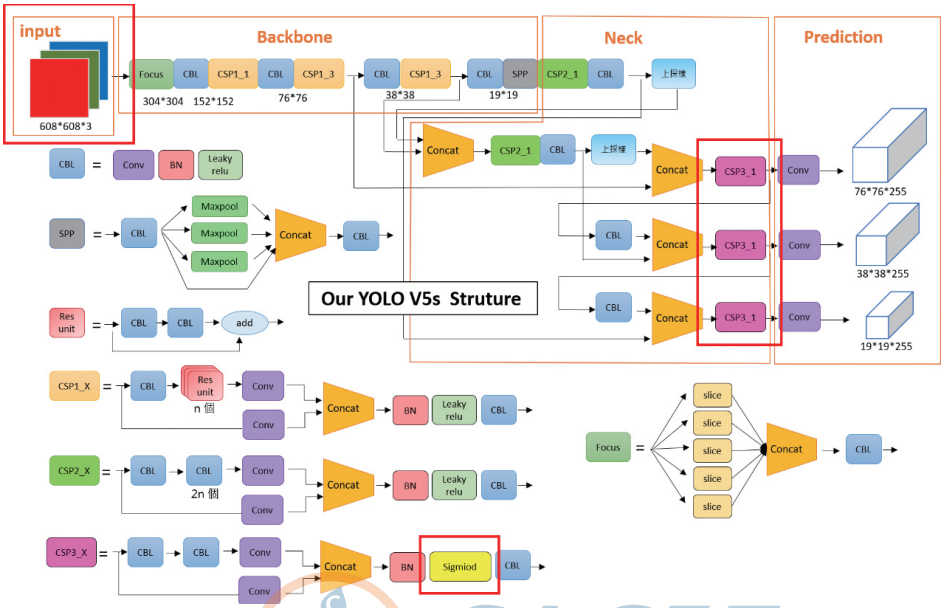

圖 3 神經網路架構

(一) 輸入

輸入測試影像時常用的做法是將影像調整至統一的大小，我們在此更進一步希望縮放後影像 padding 的邊框越少越好，以此減少不必要的運算，以下是我們使用的自適應影像縮放方法公式：

$$\alpha = \frac{|GhP - GwP| \bmod 32}{2} \qquad (3.1)$$

$$P = \min\ (Ph, Pw) \qquad (3.2)$$

$$Ph = \frac{eh}{Gh}\ ,\ Pw = \frac{ew}{Gw} \qquad (3.3)$$

其中 α 為需填充的像素行或列數。

(二)Backbone

除了使用 Mosaic 數據增強提升小目標偵測性能外，我們也使用旋轉影像、加入雜訊、調整亮度、調整對比度、縮放及傾斜旋轉等方式相互結合，盡可能擴充數據集的豐富度，降低類別不平衡的影響。

不同於 YOLO v4 訓練時使用的 Mish 激勵函數，我們採用更快速的 Leaky ReLU 激勵函數作為預訓練激勵函數，其公式如下：

$$LReLU\ (x) = \begin{cases} x \text{ if } x > 0 \\ \alpha x \text{ if } x \leq 0 \end{cases} \qquad (3.4)$$

其中 α 在訓練中以反向傳播進行調整。

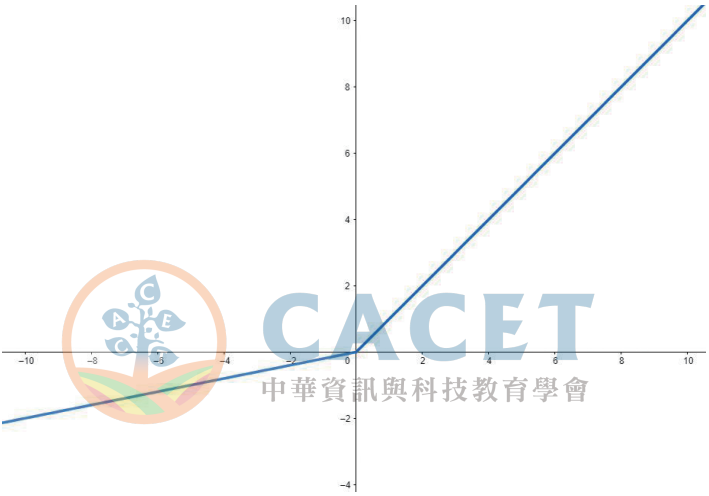

圖 4 Leaky ReLU 激勵函數

(三)Prediction

輸出 anchor box 時，YOLO v5 採用 GIOU_Loss 來更進一步提升運算速度，但因為我們調整過測試影像縮放的縮放方式，計算中考量了影像長寬比的 CIOU_Loss 更為適合，所以選擇其作為我們的損失函數。

$$\text{CIOU}_{\text{Loss}} = 1 - \text{CIOU} = 1 - \left(\text{IOU} - \frac{Distance_2{}^2}{Distance_{c^2}} - \frac{v^2}{(1-\text{IOU})+\text{v}}\right)$$

$$\text{v} = \frac{4}{\pi^2}\left(\tanh^{-1}\frac{w^{gt}}{h^{gt}} - \tanh^{-1}\frac{w^{p}}{h^{p}}\right)^2$$

$$(3.6)$$

在神經網路的隱藏層我們使用 Leaky RELU，測試時我們則選擇 Sigmoid 作為激勵函數。

$$\text{Sigmoid}(x) = \frac{1}{1+e^{-x}} \qquad (3.7)$$

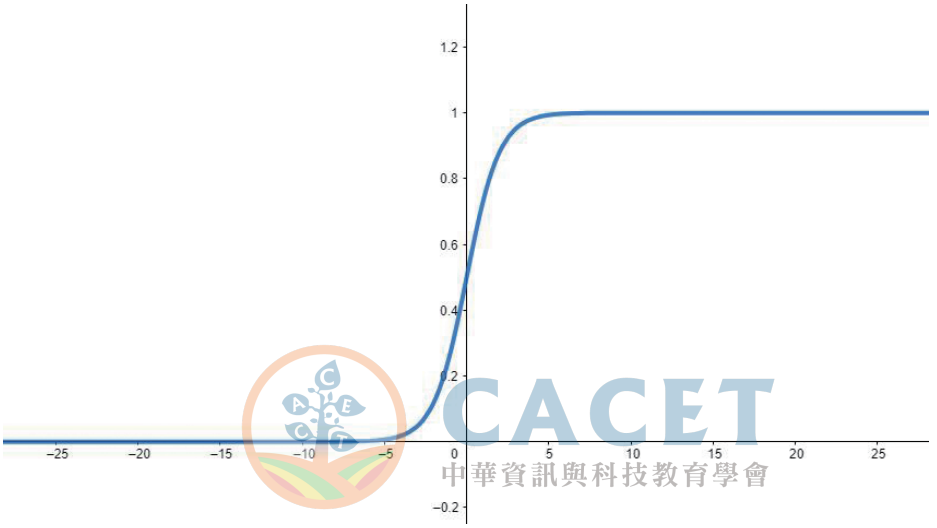

圖 5 Sigmoid 激勵函數

二、數據集

目前較為知名且包含戴口罩的公開人臉資料庫如下所列：

1. WIDER Face dataset 包含了 32,203 張人臉影像及 393,703 個人臉邊界標註框，測試子集有 3,226 張人臉影像與 12,880 個人臉邊界標註框，並分為三種偵測難度：容易、普通和困難，我們在其中挑選了 500 張影像，其中包含 221 張戴口罩人臉影像及 103 張人臉上有遮蔽物的影像（Yang et al., 2016）。
2. Kaggle Face Mask Detection dataset 包含了 853 張分別標註了有戴口罩、沒戴口罩、口罩沒戴好的影像。
3. Kaggle YOLO medical mask dataset 包含了 631 張標註了戴口罩人臉邊界框影像。

4. Real-World Masked Face Dataset（RMFD）收集了包含 525 人的約五千張戴口罩人臉影像及九萬張人臉影像，還有將一般人臉影像後製成戴口罩人臉影像約 50 萬張。

雖然人臉資料庫數量眾多，但適合本研究的影像，尤其是口罩配戴不當的影像難尋，且其中有許多影像是人工創造而非自然影像，或是標註錯誤、影像只有人臉而沒有背景等等無法使用或訓練效果不佳的情況，因此除了使用現存的資料庫，我們也藉由拍攝、爬蟲取得更多的訓練影像並標註，然後使用前文所述之數據增強方法對其擴充，最後一共是 4,431 張影像，其中包含 6,448 個戴口罩人臉標註框、1,679 個沒有戴口罩人臉標註框、470 個錯誤配戴口罩人臉標註框，並將其隨機以 4:1 比例分配給神經網路的訓練集與驗證集，圖 6 為其中一張包含三種類別的影像及其標註。

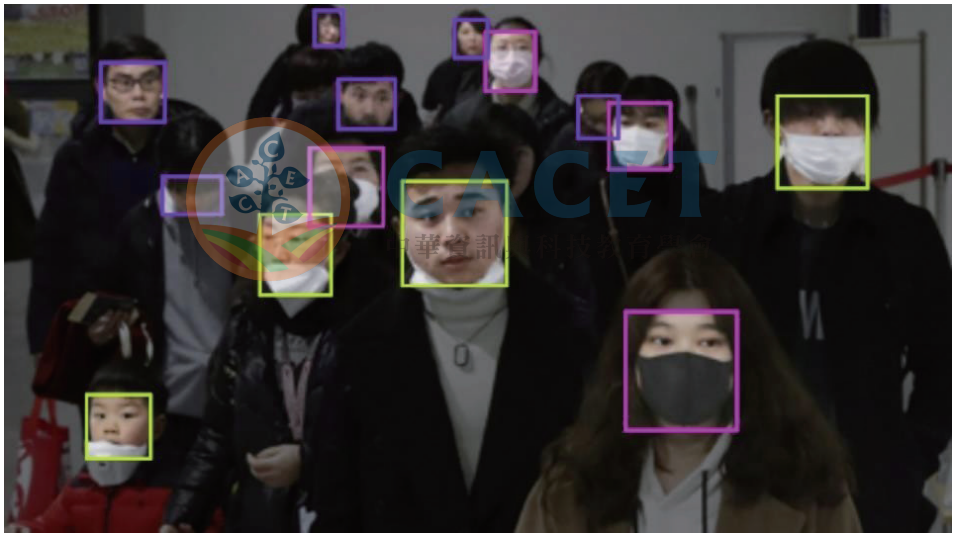

圖 6 三種類別的影像及其標註

三、演算法

本節將闡述我們的特徵點遮蔽人臉偵測神經網路的演算法，我們將整個過程分為兩個演算法。

首先對數據集影像進行預處理，然後對數據集進行訓練，接著使用訓練出來的模型偵測測試影像中是否有配戴口罩的人臉及其是否配戴正確。如表 1 所示，在訓練的部分將影像及像素值作為輸入，對影像調整大小並歸依化後進行數據增強以擴展數據集，降低過擬合的可能，然後將數據集隨機分為訓練集與驗證集，以 Adam 優化函數編譯整個模型後存檔。

接著如表 2 所示，將之前存檔的模型部屬到要偵測的影像或影片中，如果偵測到人臉，將會顯示一個邊框將其包圍，並顯示該人臉屬於正確配戴口罩、錯誤配戴口罩、沒有配戴口罩中哪一類。

表 1 數據集預處理與訓練演算法

演算法 1：數據集預處理與訓練
INPUT：已標註的影像
OUTPUT：訓練模型
STEP1：讀取影像和他們的像素值
STEP2：對影像做預處理
STEP3：讀取影像檔名及對應的標註檔
STEP4：對影像應用數據增強技術
STEP5：將影像分成訓練集和驗證集
STEP6：用 Adam 優化函數編譯模型
STEP7：將模型儲存

表 2 部屬偵測模型

演算法 2：部屬偵測模型
INPUT：要偵測的影像或影片
OUTPUT：顯示邊框包圍影像或影片中出現的人臉並顯示其是否配戴口罩與配戴正確
STEP1：讀取訓練好的模型與要偵測的影像或影片
STEP2：如果輸入的是影像
讀取影像
STEP2.1：應用偵測模型並偵測影像中是否有偵測目標
STEP2.2：取得並儲存偵測結果
STEP3：如果輸入的是影片：
讀取影片的每一幀影像
STEP3.1：應用偵測模型並偵測影像中是否有偵測目標
STEP3.2：取得並儲存偵測結果

肆、研究結果

一、研究設備

本研究使用了配備 Intel i5-6500 處理器（3.2GHz）、2 x 8GB DDR3 記憶體及 NVIDIA GeForce GTX 1060 6GB 平行處理器的桌上型電腦，並且使用 Python3.7 環境下的 Jupyter Notebook 軟體實現與訓練神經網路。

二、評估指標

目標偵測神經網路常見的評估指標有以下幾種：

$$\text{Precision} = \frac{\text{Tp}}{Tp+Fp} \quad (4.1)$$

$$\text{Recall} = \frac{\text{Tp}}{Tp+Fn} \quad (4.2)$$

$$\text{F1score} = 2 * \frac{Recall+Precision}{Recall+Precision} \quad (4.3)$$

$$\text{AP} = \int_0^1 p_{smooth}\ (r)\ dr \quad (4.4)$$

其中 *Tp*= True positive 表示標註為真且預測結果為真，*Fp*= False positive 表示標註為假但預測結果為真，*Fn*= False negative 則表示標註為真且預測結果為假，預測結果的真假則通過 CIOU_Loss 損失函數與一個閾值（threshold）來判斷。Precision 為某一類別預測結果正確的準確率，結果如圖 7 所示，Recall 為某一類別預測結果正確佔所有此類樣本的比例，結果如圖 8 所示，F1 分數則為 Precision 與 Recall 的調和平均數，結果如圖 9 所示。如圖 10 所示，以 Precision 為縱軸、Recall 為橫軸畫出一 PR 曲線圖後，我們可以看出兩者呈現負相關，Average Precision（AP）為對此曲線平滑化後做積分做為評估指標，mAP 則為所有類別的 AP 值平均。

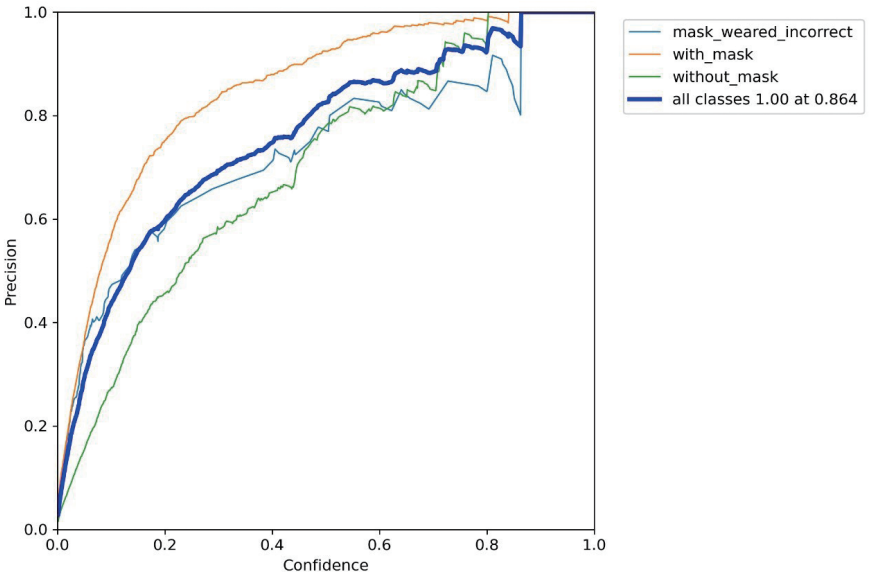

圖 7 Precision 曲線圖

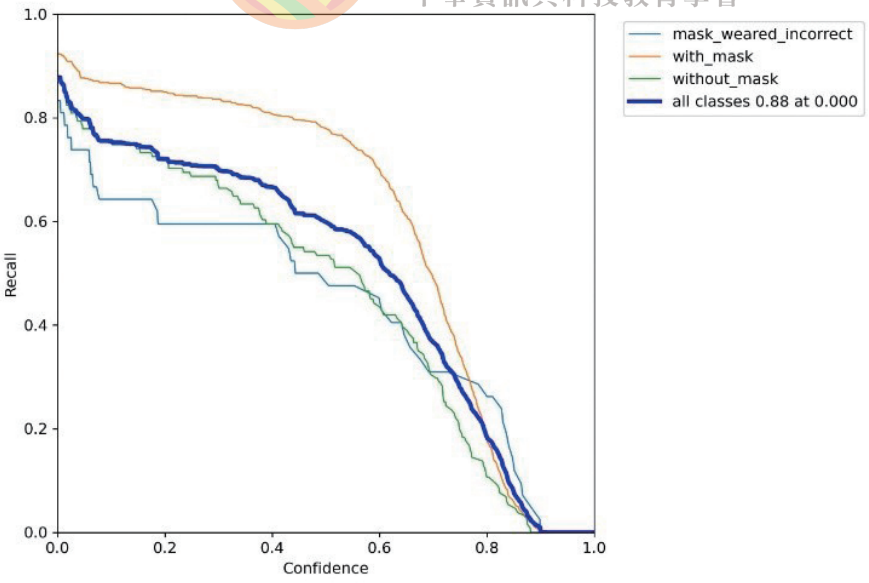

圖 8 Recall 曲線圖

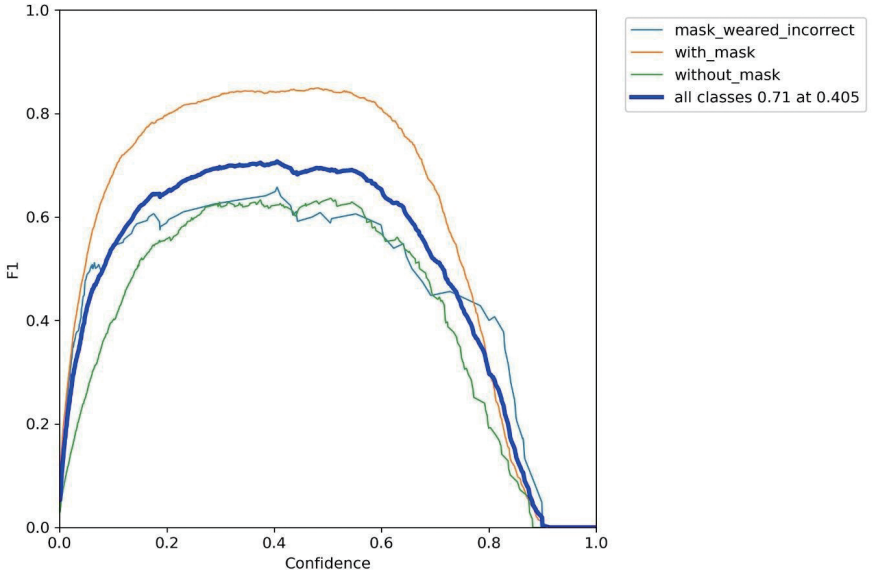

圖 9 F1score 曲線圖

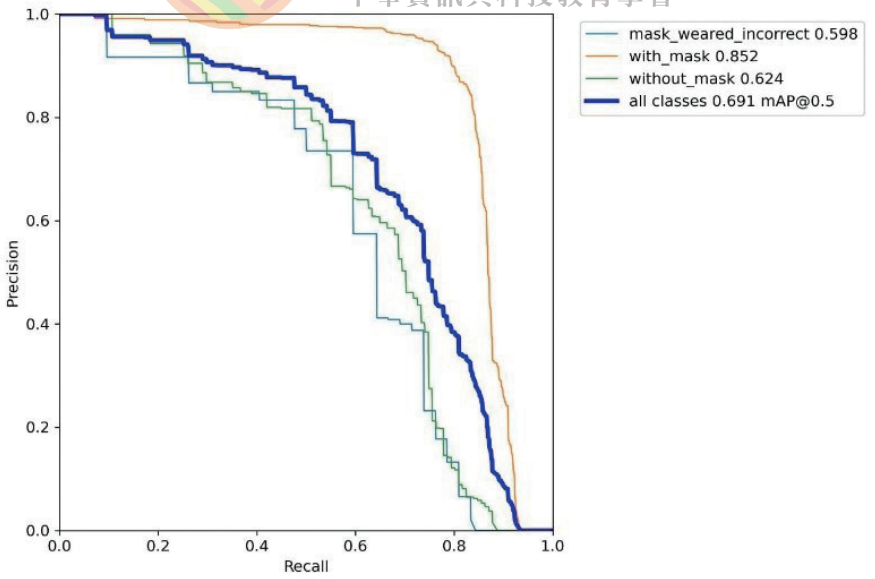

圖 10 PR 曲線圖

我們的架構與使用 YOLO v4、YOLO v5s、YOLO v5m 架構對相同數據集之評估指標比較結果如表 3 所示。

表 3 不同架構的評估指標比較結果

	Precision（%）	Recall（%）	F1 score	AP@0.5
YOLO v4	**78.85**	57.83	66	64.1
YOLO v5s	75.68	62.64	69	67.68
YOLO v5m	77.96	**69.63**	**73**	**69.69**
Ours	78.08	66.53	71	69.13

我們的架構與使用 YOLO v4、YOLO v5s、YOLO v5m 架構對相同數據集之訓練時間比較如表 4 所示。

表 4 不同架構的訓練時間比較結果

	300epochs（hours）
YOLO v4	5.428
YOLO v5s	**4.804**
YOLO v5m	16.504
Ours	4.875

我們的架構與使用 YOLO v4、YOLO v5s、YOLO v5m 架構對同一部 1920x1080 解析度的影片之偵測速度比較如表 5 所示。

表 5 不同架構的偵測速度比較結果

	FPS
YOLO v4	41.66
YOLO v5s	46.53
YOLO v5m	33.44
Ours	**47.61**

圖 11 為多目標影像測試結果，結果為少偵測八個目標，其餘正確。

圖 11 多目標影像測試結果

圖 12 為僅露出鼻孔的影像測試結果，結果為正確。

圖 12 僅露出鼻孔影像測試結果

圖 13 為常見行人的影像測試結果，結果為少偵測一個目標，其餘正確。

圖 13 常見行人影像測試結果

圖 14 為用手遮口罩的影像測試結果，結果為正確。

圖 14 用手遮口罩影像測試結果

圖 15 為戴口罩側臉的影像測試結果，結果為正確。

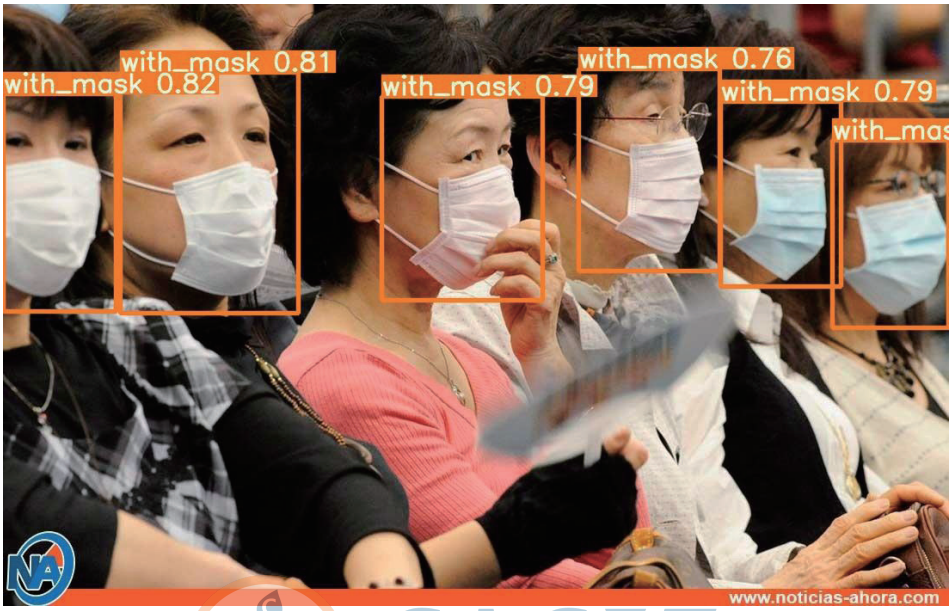

圖 15 戴口罩側臉影像測試結果

圖 16 為不同造型口罩的影像測試結果，結果為少偵測一個目標，其餘正確。

圖 16 不同造型口罩影像測試結果

由以上測試結果，我們的神經網路模型比 YOLO v5m 節省了 70.5%、比 YOLO v4 節省了 10.2% 的訓練時間，準確度方面在四種評估指標中也都有不錯的成績，偵測時間則在四種架構中取得最佳的 47.61 FPS 數據。

模型的適用範圍為少於五十個偵測目標、光線良好之影像，對於口罩的各種顏色與款式大多能有效地辨識出來，且對於影像的解析度並無太高要求，我們認為與我們資料集中樣本採用了來自各資料集與我們自行拍攝、爬蟲收集之影像包含了上述各種類型之影像有關；較為不足的部分則有透明的口罩、以人體彩繪方式畫在人臉上之口罩這種影像，也許可以試著在資料集多加入這些影像，或是針對這類型影像做更多的數據增強來改善。

伍、結論

本研究針對戴口罩人臉的數據集經過嚴格的篩選與重新標註，完成了區分為三個類別的訓練及開發，可適用於輸入各種解析度的影像與影片，參考了 YOLO v4、YOLO v5 與其他神經網路架構，我們的神經網路模型以 4.875 小時運行 300 次 epoch 完成訓練的速度，於 1920x1080 解析度的影片下可達到平均每秒偵測 47.61 幀影格的結果，並且對於正臉、側臉、小目標、五十個以下的多目標、不同顏色口罩、不同款式口罩、口罩被遮擋、口鼻被非口罩遮擋之影像皆能達到一定的準確度與偵測率。實驗證實我們的方法與相同精度的 YOLO v5m 相比，節省 70% 的時間。

在疫情肆虐全球的現在，每個人都需要提高警覺，出門戴口罩更是必不可少，本研究訓練之深層神經網路模型可適用於車站、學校出入口等人流密集的地方，以其判斷快速的特性做到實時區分出對於戴口罩做得不夠確實的人群，提高大家的警惕心，降低群聚感染發生的可能，可望對於防疫做出些許貢獻。

目前的數據集若使用更深的神經網路架構，容易因類別不平衡導致過擬合的情況發生，如果加入更多的真實影像數據應該可以使訓練模型的準確度更上一層樓。另外也許可以結合紅外線熱像儀產生的紅外線影像，開發一個能同時判斷是否配戴口罩、體溫是否正常之警報系統，讓防疫工作更加完善。

參考文獻

Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

Ghiasi, G., Lin, T. Y., & Le, Q. V. (2018). Dropblock: A regularization method for convolutional networks. *Advances in Neural Information Processing Systems, 31,* 10727-10737.

Girshick, R. (2015). Fast r-cnn. *Proceedings of the IEEE International Conference on Computer Vision,* 1440-1448.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 580-587.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(9), 1904-1916.

Howard, J., Huang, A., Li, Z., Tufekci, Z., Zdimal, V., van der Westhuizen, H. M., ... & Rimoin, A. W. (2020). Face masks against COVID-19: an evidence review.

Kumar, A., Kalia, A., Sharma, A., & Kaushal, M. (2021). A hybrid tiny YOLO v4-SPP module based improved face mask detection vision system. *Journal of Ambient Intelligence and Humanized Computing,* 1-14.

Lecun, Y., Bottou, L. Bengio, Y. & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278-2324. https://doi.org/ 10.1109/5.726791.

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2117-2125.

Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path aggregation network for instance segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 8759-8768.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European Conference on Computer Vision,* 21-37.

Loey, M., Manogaran, G., Taha, M. H. N., & Khalifa, N. E. M. (2021). Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection. *Sustainable Cities and Society, 65,* 102600.

Misra, D. (2019). Mish: A self regularized non-monotonic activation function. *arXiv*

preprint arXiv:1908.08681.

Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 7263-7271.

Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767.*

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 779-788.

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans., 39*(6), 1137-1149.

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., & Savarese, S. (2019). Generalized intersection over union: A metric and a loss for bounding box regression. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 658-666.

Roy, B., Nandy, S., Ghosh, D., Dutta, D., Biswas, P., & Das, T. (2020). MOXA: A deep learning based unmanned approach for real-time monitoring of people wearing medical masks. *Transactions of the Indian National Academy of Engineering, 5*(3), 509-518.

Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops,* 390-391.

Wang, F. (2020). Improved yolov5 artificial intelligence detection and recognition algorithm for wearing masks and helmets. *Architecture and Budget, 11*(11), 67-69.

Wang, S., Wu, Z., He, G., Wang, S., Sun, H., & Fan, F. (2021). Semi-supervised classification-aware cross-modal deep adversarial data augmentation. *Future Generation Computer Systems, 125*, 194-205.

Wang, Z., Wang, G., Huang, B., Xiong, Z., Hong, Q., Wu, H., ... & Liang, J. (2020). Masked face recognition dataset and application. *arXiv preprint arXiv:2003.09093.*

Yang, S., Luo, P., Loy, C. C., & Tang, X. (2016). Wider face: A face detection benchmark. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 5525-5533.

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., & Yoo, Y. (2019). Cutmix: Regularization strategy to train strong classifiers with localizable features. *Proceedings of the IEEE/CVF International Conference on Computer Vision,* 6023-6032.

Zhang, S., Sun, J., Kang, J., & Wang, S. (2021). Research on Recognition of Faces with Masks Based on Improved Neural Network. *Journal of Healthcare Engineering, 2021*.

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., & Ren, D. (2020). Distance-IoU loss: Faster and better learning for bounding box regression. *Proceedings of the AAAI Conference on Artificial Intelligence, 34*(7), 12993-13000.

Facial Mask Detection Using Convolutional Neural Networks

Chi-Yen Li

PhD student
Department of Business Administration,
National Central University
E-mail: rickyyenli@gmail.com

Su-Chi Fuh

Master student
Department of Computer Science & Information Engineering,
National Taipei University of Technology
E-mail: es891010@gmail.com

Chen-Jie Lin

Master student
Department of Computer Science,
National Taipei University of Education
E-mail: ckdcshadow@gmail.com

Abstract

The extremely high transmission rate of the COVID-19 has made the supply of medical resources in countries around the world in short supply. In view of the fact that wearing masks is currently an effective method of epidemic prevention, and the current face detection models are not effective for masked faces that cover half of their faces, and pedestrians who have not worn masks in the correct way are occasionally visible. It may spread the epidemic.

This research will establish a face data set with three kinds of annotations, and combine a variety of deep learning convolutional neural network architectures and methods to design a face detection model that can quickly train and detect wearing a mask, not wearing a mask, and wearing a mask incorrectly faces. Hope to contribute to the epidemic prevention and control of the epidemic.

We use an adaptive algorithm to adjust the image size to reduce unnecessary operations, and modify the CIOU_LOSS error function to speed up the operation.

Experiments have confirmed that our algorithm saves 70% of the time compared to YOLO v5m with the same accuracy.

Keywords: COVID-19, Convolutional Neural Network, Face Detection, Masked