

# 裝訂業機器摺紙精準度AI 影像辨識

---

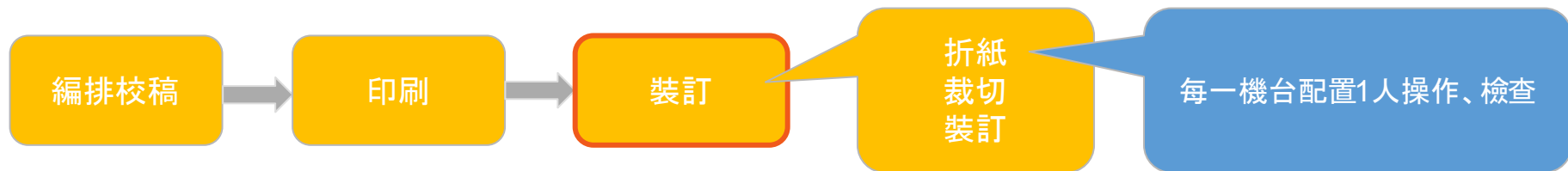
廖彥欽

# 課程資訊

技術分類	電腦視覺、智慧製造、瑕疵檢測
課程目標	<p>以印刷裝訂業折紙機瑕疵檢測為例說明：</p> <ul style="list-style-type: none"><li>● 學習訓練AI模型的資料處理技巧</li><li>● 缺乏大量訓練資料的應用情境神經網路的訓練技巧</li><li>● 選取適合網路結構，以符合裝訂業產線”少量多樣”的特色</li></ul>
適合對象	具備python 程式撰寫、機器學習、深度學習基礎之學生、AI工程師、社會人士

# 前言

一本書籍從內容編輯、印刷到最後裝訂成冊販售，經過了許多道的程序。在內容校稿完成到成品（書籍）製作完成主要可分為印刷與裝訂兩大流程。本次解題則著重於印刷完成後的裝訂流程的第一步：**折紙**。



# 單元內容

- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：變生網路與實作
- 單元6：結果與討論
- 單元7：重點回顧

# 單元內容

- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：孿生網路與實作
- 單元6：結果與討論
- 單元7：重點回顧

- ☐ 印刷裝訂流程與現況
- ☐ 目標
- ☐ 解題方向
- ☐ 訓練資料說明

# 印刷裝訂流程與現況

書籍的裝訂並非一張張、一頁頁的分別裝訂，而是在印刷時，將多頁內容印製成一張大張、雙面的印刷。接著由我們這次的主角“折紙機”，依照機器的設定，依序摺出正確的頁數。

一張多頁的大紙張能折出正確的樣式，除了選取正確的折法，最終的折紙品質還仰賴機器的物理參數的設定。不同書本、紙張、印刷方式，都有不同的設定，在此領域是完全仰賴操作人員的經驗。

然而折紙機的機械結構會隨著使用時間而產生偏差偏移，就造成了折紙的瑕疵。

要偵測折紙瑕疵，現有作法是每一折紙機台配置一個專責的工作人員，憑其經驗調控設定好參數後，先以少量的紙張試折，接著人工目測每一紙張的特殊對準標誌。當發生偏差時，則需要停止機器的運作，否則後續的成品因裝訂偏差造成裝訂的書本瑕疵將無法交付給客戶，而產生損失。

# 裝訂廠的摺紙機現場



裝訂業的工作人員相當辛苦，工作需要高度的專注，工作環境有相當大的噪音與粉塵。因此也期望能應用AI技術，將來能將此繁複的工作逐步交由AI處理。

# 目標

- 利用AI技術偵測折紙偏差, 即時通知現場人員前往校正折紙機參數
- 導入AI降低人力需求, 解決業主招工不易的困境



# 解題方向

折紙的瑕疵主要來自於參數設定的錯誤、或機械運作隨時間產生的偏移。要偵測瑕疵，可以從兩個方向著手：

1. 藉由機台參數或狀態，數值分析，預測瑕疵。機台狀態數據，可以經由機器的電控廠商的PLC控制電路讀取，或經由加裝感測器取得。
2. 藉由影像分析成品外觀，判定瑕疵。

作法1是藉由數值分析預測瑕疵，是現代化工廠導入智能生產流程的常見作法。難處在於設備提供廠商與需求方不易在技術與商務面取得合作共識。

而且折紙機容易有大振動，固定感測器不容易，加上電源與空間限制，**最後仍單純以視覺，藉由判斷成品的外貌來檢測瑕疵。**

# 訓練資料

每個印刷品有不同的對齊標誌。本次課程的訓練資料為實地拍攝取得，清洗後手工標註的資料。包含兩種以上的對齊符號的影像檔案 (936張)，以及各自對齊符號位置的標籤檔案 (xml格式)。



# 單元內容

- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：孿生網路實作
- 單元6：結果與討論
- 單元7：重點回顧

- ❑ 物件偵測與分類
- ❑ 輕量化模型
- ❑ 產品少量多樣

# 挑戰1: 缺乏機器學習需要的影像資料

1. 自行收集資料
2. 安裝攝影機、透過影像串流或定期拍照回傳影像
3. 工廠沒有連網設備(無WIFI)

使用樹莓派 搭配 4G/5G router 將影像回傳



鏡頭貼膠帶  
防塵

外掛設備(邊緣運算)

## 挑戰2: 單點操作環境, 無規劃流水線

- 本次施工的折紙機, 屬於裝訂流程中較前端的步驟。目前同業作法仍是以機器單點操作。
- 每一機器與機器之間需要靠人工搬運半成品與操作。
- 無規劃流水線, 因此設計上以輕量的邊緣運算為主。

## 挑戰3: 環境干擾

- 取像設備與邊緣運算裝置以外掛的方式固定在機台週圍，容易遭受機器振動脫落、或人員誤觸電源造成設備中斷。
- 空氣中的粉塵也造成取像干擾，照片偏於模糊。

## 挑戰5: 產品少量多樣

- 每本書都有自己的設定與對齊標記, 且每一次裝訂的數量也不多(1000本算很多)  
⇒ AI 模型需要快速調整
- 可能缺乏大量標註的資料訓練模型  
⇒ 選擇傾向俱備泛化能力的 AI 模型



# 解題構想

找出對齊符號（物件偵測）



異常對齊符號偵測（分類）

- 選擇“輕量型”的AI模型
- 借用image segmentation [\[7\]](#)與 anchor-free object detection [\[8\]](#) 概念實作物件偵測
- 以巒生網路[\[9\]](#) 訓練分類模型，判斷對齊異常

# 單元內容

- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：孿生網路與實作
- 單元6：結果與討論
- 單元7：重點回顧

- ☐ 收集資料
- ☐ 資料清洗與標註
- ☐ 建模訓練
- ☐ 模型佈署

# 收集資料

- 以樹莓派4B[\[1\]](#)加裝設像頭(pi-camera), 透過4G-LTE訊號分享器電信行動網路, 傳送資料與遠端開發程式。
- 額外加裝了監視器拍攝取像裝置, 輔助我們在設備數值異常時, 判斷是否因環境因素(例如工廠熄燈午休)或是樹莓派本身程式設定錯誤而造成。

# 資料清理與標註

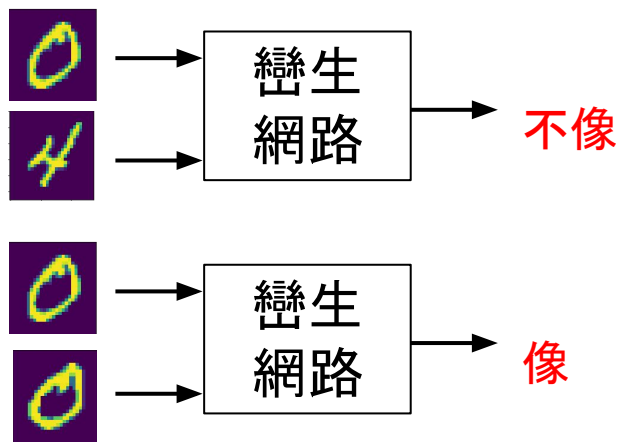
- 收集的資料屬於影像串流檔，為了節省儲存，只有在畫面有改變時，也就是機器實際在運轉時，才將影像傳出。
- 資料清洗：自動剔除太模糊之影像。由於照片繁多，以演算法自動挑選，作法可參考[\[2\]](#)
- 標註(labeling)：本次實作需要以物件偵測(object detection)偵測出畫面中的對齊標記(mark)並且分辨出”對齊”與”非對齊”。資料標註需花費人工，除了對齊與非對齊的類別外，還需要框出對齊標記在每張圖片出現的座標位置。

# 建模訓練 - 物件偵測

- option1 - 利用遷移學習(Transfer learning) [3] 技巧與預訓練模型微調。選用Tensorflow lite model maker [4] 調整，但訓練過程時常發生記憶體資源不足的窘境，在複雜度的考量下只能選擇稍微小一點的模型，然而準確度也無法滿足需求。
- option2 - 利用現有圖片，重新訓練較小的模型：上述的預訓練模型，可處理較複雜的任務(例如偵測1000種物件類別)。然而本次解題目標的每張畫面中，最多只會出現一個物件，也就是對齊標記，除了可用灰階影像表示外，任務相對單純
- 結果：option2 效果佳

# 建模訓練 - 分類

- 採用**孿生網路**<sup>[9]</sup>架構。對於輕量的深度學習網路，以少量的圖片訓練，快速調適並不容易。
- 模型輸入是兩張圖片，各自經過**結構與係數完全相同**的深度網路，將其影像特徵取出後，整併預測兩張圖片相似度。
- 可以將”正常對齊”的圖片與待測圖片作為輸入的**圖片組**，當相似程度較低，則可以推論此張圖片可能不符合折紙標準。



# 佈署

- 訓練後模型需要優化
- 利用Tensorflowlite 在無GPU的邊緣運算裝置(樹莓派)

# 單元內容

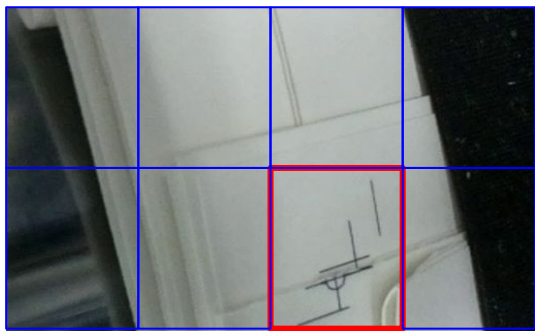
- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：孿生網路與實作
- 單元6：結果與討論
- 單元7：重點回顧

- ❑ 概念說明
- ❑ Python實作範例

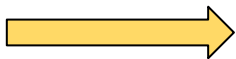


# 簡易物件偵測：只要找出“在哪裡”

- 每張圖只有一個對齊標誌
- 借用image segmentation [\[7\]](#) 與anchor-free object detection [\[8\]](#) 概念
- 影像轉化成二維矩陣，矩陣元素內容反應此區域是否含有目標物件
- 可以簡單convolutional neural network實現物件偵測



影像轉化二維矩陣

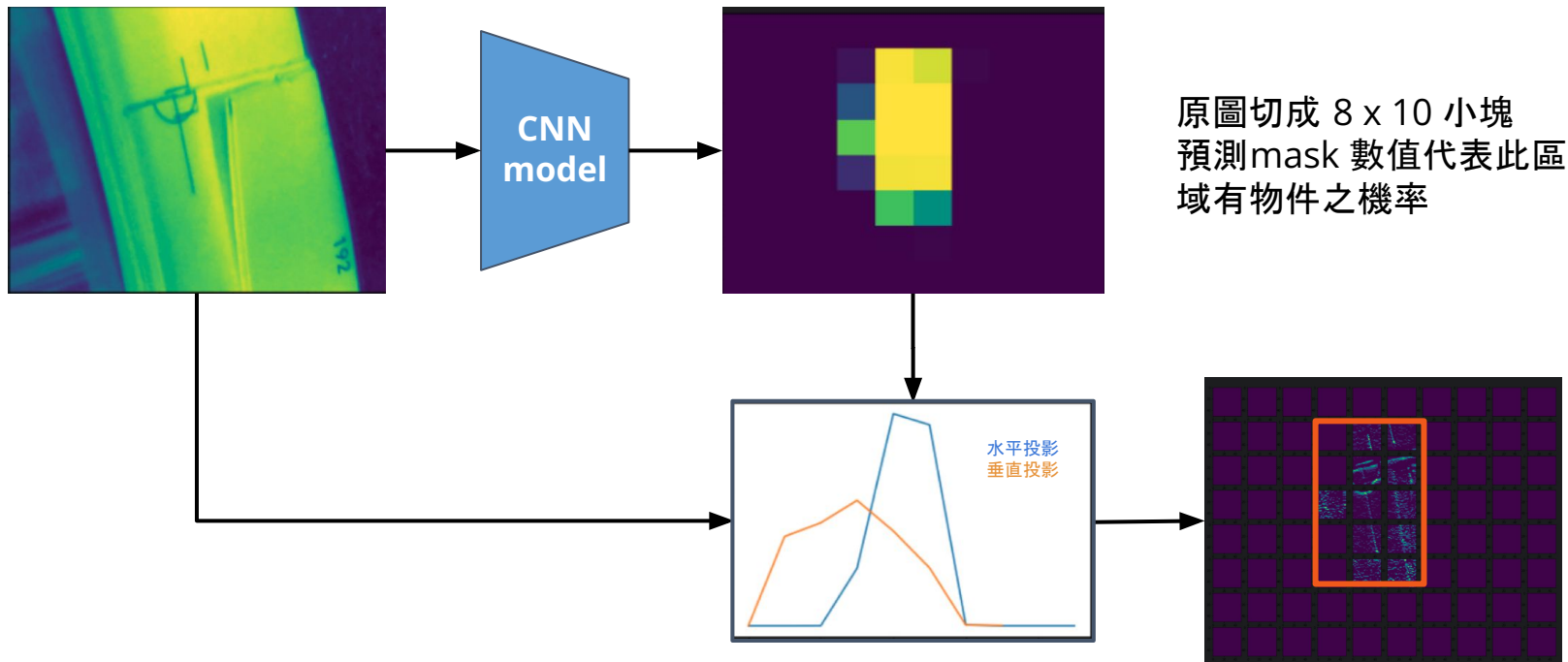


CNN模型預測  
二維矩陣內容

0	0	0	0
0	0	1	0

# 簡易物件偵測

CNN模型預測mask



從水平/垂直方向投影非零區域找出 bounding box

# Python 實作範例

## 1. 下載 資料夾


[https://drive.google.com/drive/folders/1KS2T9DfEkGpl8ks-JrXicbDVtsxXDSwO?usp=share\\_link](https://drive.google.com/drive/folders/1KS2T9DfEkGpl8ks-JrXicbDVtsxXDSwO?usp=share_link)

## 2. 執行 block\_segmentation\_0D.ipynb 訓練模型

## 3. 執行 inference\_tflite.ipynb 推論

# 單元內容

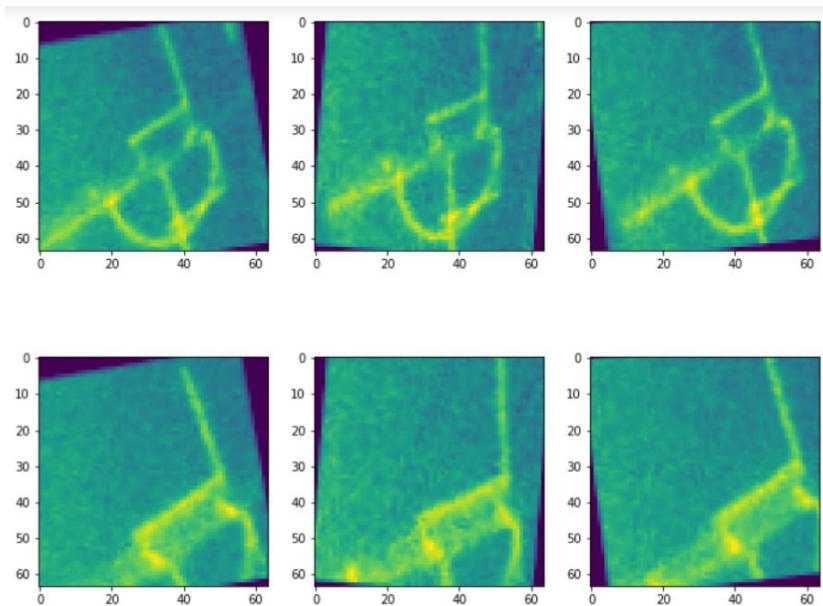
- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：孿生網路與實作
- 單元6：結果與討論
- 單元7：重點回顧

- 
- ❑ 概念說明
  - ❑ Python實作範例

# 偵測瑕疵 - 還有哪些可行方案？

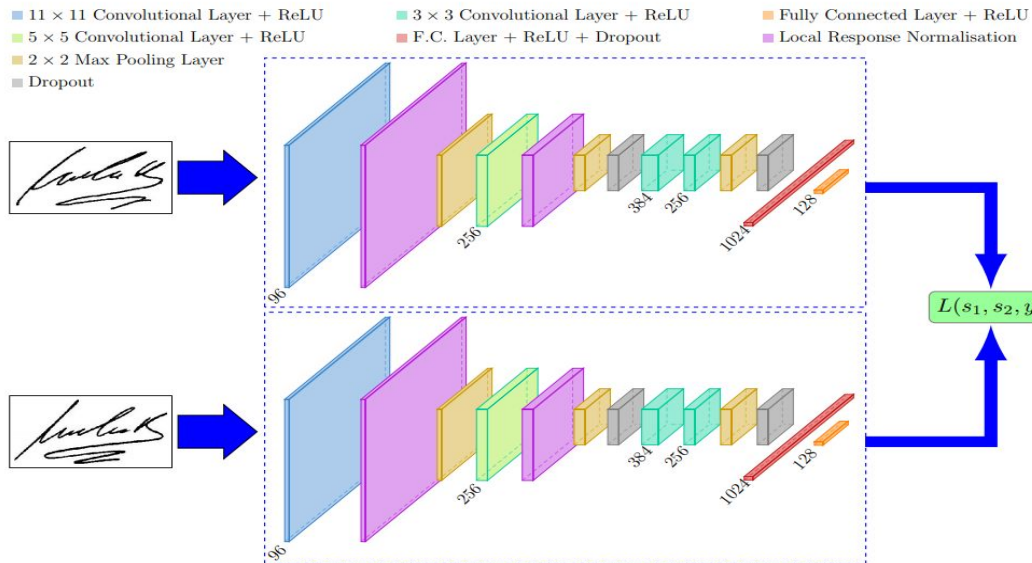
- 選項1: 傳統computer vision 作法
  - 偵測邊緣、線條、輪廓與形狀
  - 演算法需要的參數難設定, 也需要隨鏡頭拍攝角度、環境光線變化
  - 難以列舉出所有的對齊標記, 不易應付產品少量多樣的特性
- 選項2: 在預訓練好的網路上作遷移學習或fine tuning
  - 網路結構可能較複雜, 運算量多
- 選項3: 利用data augmentation增加訓練資料數量、**調整OK/NG比例**
  - 更換對齊標記(新產品)的時候可能需要重新訓練網路
- **選項4: 以孿生網路比對影像特徵與相似度**
  - 比較兩張圖片的**相似度**: 圖片比較像NG或是OK?
  - 新產品(模型未見過的新對齊標記)仍然可用

# 試試Data Augmentation解決 Data Imbalanced ?



- 利用 python **imgaug** 套件
  - random crop
  - random Affine transform
  - random noise
  - random contrast
- 調整後訓練資料共 490張
  - OK/NG 比例由 7% 調整至 61%
  - Validation accuracy 0.9688
  - Validation F1 score = 0.67
- **Testing data (new mark)**
  - **F1 score = 0**
- 沒看過的OK圖案都判成NG
- Augmentation的圖片不夠真實

# 孿生網路：像或不像？



不同書本有不同的對準符號

與其學習瑕疵特徵  
不如學習”像不像標準”

圖片來源: SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification (<https://arxiv.org/abs/1707.02131> )

# 訓練學生網路-1: 準備資料

- 分做三種類別: NG, OK, NotSure (模糊/對齊符號不完整)
- 只用NG, OK的資料訓練二分類分類器

訓練資料 (對齊標記樣式1)

NG : OK = 38: 3, 共41張圖片

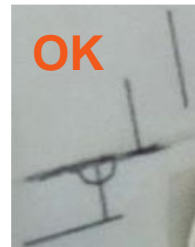
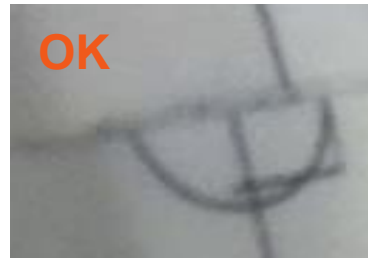
驗證資料 (對齊標記樣式1)

NG: OK = 30: 2, 共32張圖片

測試資料(對齊標記樣式2)

NG : OK = 50:2 共52張

(樣式2長寬比與樣式1不同)





# 學生網路實作

- **Step-1: 產生配對訓練資料**
  - {NG, NG}, {NG, OK}, {OK, OK}
  - 藉由配對圖片, 增加訓練資料數量
- **Step-2: 特徵抽取網路 (feature extraction)**
  - convolutional neural network
- **Step-3: 比較兩個輸入的特徵值差異**
- **Step-4: 定義損失函數**
  - binary cross entropy
  - distance: contrastive loss, triplet loss
  - 輸出層的激勵函數(output layer activation function)也要隨之調整

# Python 實作範例

<https://drive.google.com/drive/folders/1d77lQqGGSNzATQNFEBVbgPYh1SiWnt77Q?usp=sharing>

訓練資料 dataset1

執行sn.ipynb

# 單元內容

- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：孿生網路與實作
- 單元6：結果與討論
- 單元7：重點回顧

- ❑ 比較物件偵測效果
- ❑ 比較孿生網路效果

# 物件偵測測試

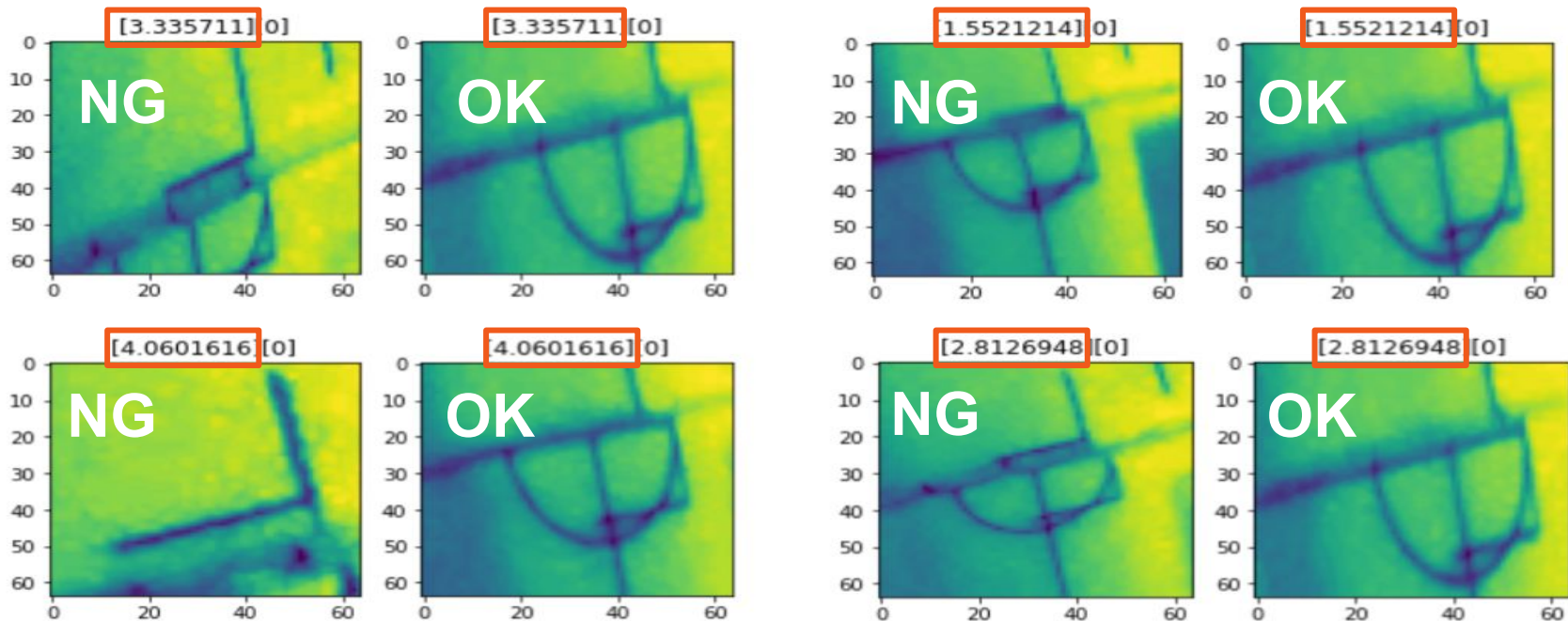
物件偵測模型IOU (intersection of Union)與detectio rate (偵測率)

測試資料	圖片數量(張)	平均IOU	最大IOU	最小IOU	偵測率 (%)
training set	553	0.4497	0.8523	0	99.8192
validation set	226	0.4648	0.8545	0	99.115
testing set	207	0.4747	0.8995	0.1171	100

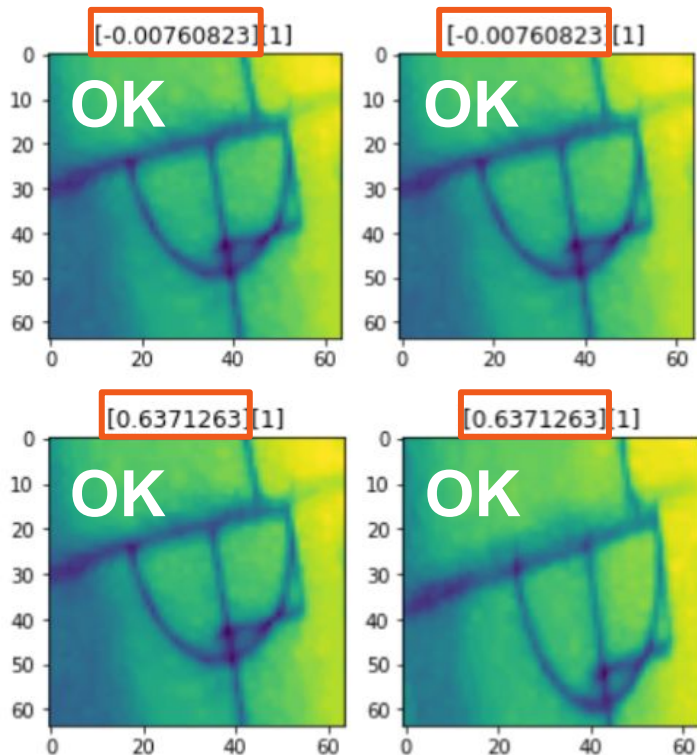
- IOU定義:  
(預測物件框與標註物件框重疊面積)/(預測物件框與標註物件框總面積)
- 偵測率定義: 正確框出圖片數 / 總圖片數

# 學生網路測試結果

Class-0: 不同類別配對資料 (NG-OK Pair), 越不相似、偏差越多距離越大



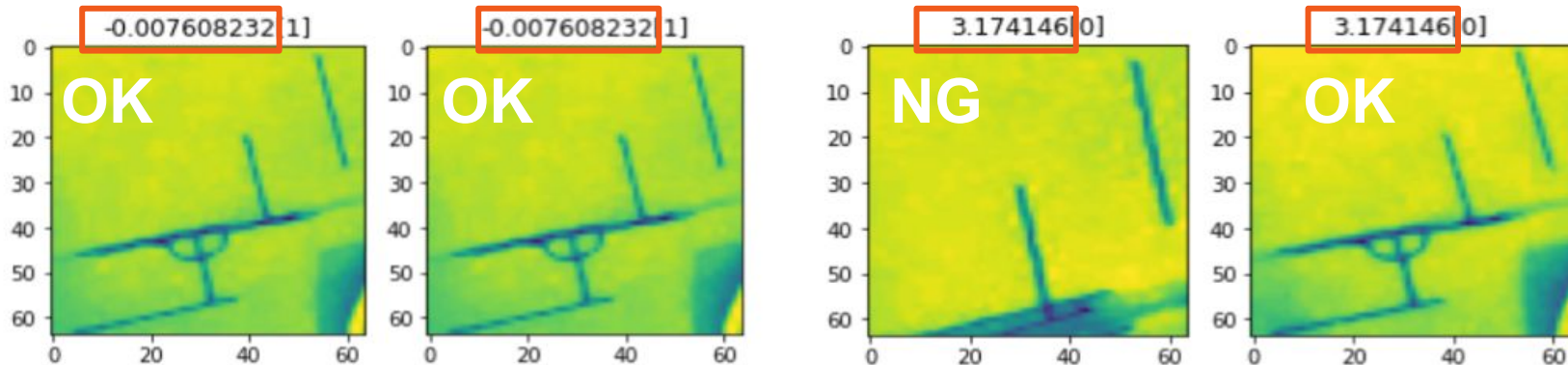
## Class-1: 相同類別配對資料(OK-OK Pair), 圖片越相似距離小



- threshold  $T$
- 若距離大於 $T$ :判定為相異配對資料

# 以模型沒見過的對齊標記作為測試資料

- 與訓練資料有不同的對齊符號
- 測試資料原始圖片與訓練資料有不同長寬比, 再resize成模型輸入端所定義的長寬比與大小
- 相同類別(OK,OK配對)距離小



# 單元內容

- 單元1：問題描述
- 單元2：解題構想與實務挑戰
- 單元3：解決方案實作流程
- 單元4：簡易物件偵測與實作
- 單元5：孿生網路與實作
- 單元6：結果與討論
- 單元7：重點回顧



# 重點回顧

- 利用AI影像技術，偵測印刷裝訂折紙瑕疵
  - **訓練資料不足**時如何訓練模型？
  - **產品少量多樣，生命週期短**
- 使用簡易物件偵測：簡單CNN model 即可找出對齊標記
- 利用孿生網路比對**成對**影像特徵差異

# 未來發展

台灣出版業**精品產品**生產比重逐年增加，每一訂單產品，量雖少但單價高，因此對產品的品質要求更為提高，將來更勢必加強生產的自動化與智能化，來提升、確保產品品質的一致性。

# 參考資料

- [1] <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>
- [2] <https://www.pyimagesearch.com/2015/09/07/blur-detection-with-opencv/>
- [3] A Comprehensive Survey on Transfer Learning (<https://arxiv.org/abs/1911.02685>)
- [4] Tensorflow Lite model maker ([https://www.tensorflow.org/lite/guide/model\\_maker](https://www.tensorflow.org/lite/guide/model_maker) )
- [5] Tensorflow hub (<https://www.tensorflow.org/hub/> )
- [6] Tensorflow model zoo ([https://github.com/tensorflow/models/blob/master/research/object\\_detection/g3doc/tf2\\_detection\\_zoo.md](https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md) )
- [7] Image Segmentation Using Deep Learning: A Survey (<https://arxiv.org/abs/2001.05566> )
- [8] anchor-free object detection(  
<https://medium.com/%E8%BB%9F%E9%AB%94%E4%B9%8B%E5%BF%83/cv-object-detection-1-anchor-free%E5%A4%A7%E7%88%86%E7%99%BC%E7%9A%842019%E5%B9%B4-e3b4271cdf1a> )
- [9] SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification (<https://arxiv.org/abs/1707.02131> )