國立中興大學資訊科學與工程學系碩士學位論文

使用串級深度學習網路於即時 PVC 皮革表面瑕疵偵測之研究

A New Real-Time Defect Detection Method for The PVC Leather Surface

Using Cascaded Deep Learning Neural Networks

指導教授：吳俊霖 Jiunn-Lin Wu

研 究 生：張立 Li Chang

中華民國 一一○ 年 八 月

*l*e*L*r A. JtifL# ..LfI. i

lJi± {ft.*�*x.



A New Real-Time Defect Detection Method for The PVC

Leather Surface Using Cascaded Deep Leaming Neural Networks 







**誌謝**

當初毅然決然地決定從職場回來進修，在研究所這段期間，得到了許多貴人的幫助，不僅是精進了學術上的知識，更結交到的志同道合的好夥伴，時光飛逝，兩年的研究所時光過得非常充實，準備邁入人生的下一個階段，非常感激在這段期間給予我幫助的老師、同學以及廠商。

首先，要感謝我的指導教授吳俊霖老師，除了在這兩年裡傳授我專業的知識之外，也給予我很多分析問題的建議與技巧，在產學合作遇到困難時，讓我能夠有條理的思考並解決問題。除此之外，也很感謝老師給予實驗室很好的資源和設備，讓我們能夠更有效率地進行實驗和測試。老師也常會邀請實驗室的成員參加爬山、運動、聚餐……等活動，增進實驗室成員的感情與凝聚力。

再來我想感謝實驗室的學長姐、學弟妹和同學，除了在修課遇到問題時， 大家會一起討論並提出各自的見解，在準備口試和寫論文這段期間大家一起挑燈夜戰，互相幫忙，祝福大家畢業後能夠一帆風順，並找到自己理想中的工作。在這邊也要特別感謝碁仕科技的陳暐諺，在進行產學合作的這段期間，給予了我很多關於相機以及鏡頭操作問題的協助。

最後，我想感謝我的家人，在我決定要回來攻讀研究所時也表示支持，在就讀期間不忘對我關心、鼓勵及包容，沒有你們給我這麼好的成長環境，也不會有今天的我，謝謝一路上有你們默默地陪伴和付出，讓我能順利完成碩士學位，並邁入人生的下一個里程碑。

**使用串級深度學習網路於即時 PVC 皮革表面**

**瑕疵偵測之研究**

# 摘要

目前使用機器視覺在檢測任務上非常普遍，像是透過機器視覺搭配影像處理來找出生產環節中的瑕疵，在檢測的能力上也優於人工，它的好處是不會影響欲檢測設備的運作和有判斷標準不一致的問題，但是以影像處理的方法通常都只能侷限在比較固定的環境，如果檢測的環境一改變，檢測的演算法就需要重新調整，而且如檢測目標的背景或是紋理更為複雜，也比較難被影像處理的方法檢測出來，相較之下，深度學習的方法針對這些複雜背景、紋理和環境變化能比影像處理的方法更加強健。

本研究採用深度學習的方法搭配可以跟上捲送機運行速度取像的彩色工業用相機對 PVC 膠皮進行瑕疵檢測，並將檢測結果傳送至可程式化邏輯控制器 (PLC)進行後續的處理。由於廠商紀錄的瑕疵資料有限，提供的瑕疵影像也與實際使用工業用相機拍攝出來的視野不同，因此在研究中使用了資料增量，來生成更多類似工業用相機取像視野的影像來訓練深度學習模型，實驗結果顯示所提方法在瑕疵偵測上有良好的表現。

**關鍵字： 深度學習、物件偵測、物件辨識、機器視覺、瑕疵檢測、工業4.0**

**A New Real-Time Defect Detection Method for The PVC Leather Surface**

**Using Cascaded Deep Learning Neural Networks**

**Abstract**

Machine vision is very common in inspection tasks. Combine machine vision and image processing to find defects in the production process, its ability is more accurate and more effective than manual method, it will not affect the operation of the equipment and there will be no different judgment standards. However, the detection environment changes, the detection algorithm needs to be modified, if the texture of the detection target is more complex, the image processing method is also more difficult to detect. For complex textures and environmental changes, deep learning methods are much more robust than image processing methods.

In this study, we proposed a deep learning method to detect the defects on the PVC leather and send the detection results to a programmable logic controller for subsequent processing. Since the defect images are limited and the field of view is different from the industrial camera, data augmentation is needed to get similar to defect images taken by industrial camera to train deep learning model. Experiment results show that the proposed method has better performance in defect detection.

**Keywords**: *Deep learning, Object detection, Object classification, Computer vision, Defect detection, Industry 4.0*

# 目錄

[緒論 1](#_gjdgxs)

* 1. [研究背景與動機 1](#_30j0zll)
  2. [論文架構 3](#_3znysh7)

[文獻探討 4](#_2et92p0)

* 1. [人工檢測 4](#_tyjcwt)
  2. [機器視覺搭配影像處理 5](#_3dy6vkm)
  3. [機器視覺搭配深度學習 7](#_26in1rg)

[研究方法 10](#_44sinio)

* 1. [瑕疵資料增量 10](#_z337ya)
     1. [影像資料增量流程 11](#_3j2qqm3)
     2. [資料增量方法論述 11](#_4i7ojhp)
  2. [深度學習物件偵測 14](#_qsh70q)
     1. [卷積層(Convolutional layer) 16](#_49x2ik5)
     2. [激勵函數(Activation function) 17](#_147n2zr)
     3. [批次正規化(Batch Normalization) 18](#_ihv636)
     4. [跨階段局部網路(Cross Stage Partial Network) 19](#_32hioqz)
     5. [殘差網絡架構(Residual block) 20](#_41mghml)
     6. [多尺度預測(Multi-scale prediction) 21](#_vx1227)
  3. [分類器(Classifier) 25](#_19c6y18)
     1. [模型訓練流程 25](#_3tbugp1)
     2. [分類器網路架構 25](#_28h4qwu)

[實驗結果與討論 27](#_1mrcu09)

* 1. [工廠環境與實驗裝置 27](#_46r0co2)
  2. [資料集 32](#_sqyw64)
  3. [實驗結果 35](#_2r0uhxc)
  4. [評估方法 39](#_1jlao46)
  5. [比較結果 41](#_3vac5uf)

[結論與未來發展 43](#_39kk8xu)

[參考文獻 44](#_1opuj5n)

**圖目錄**

[圖 1 膠皮捲送機示意圖。 2](#_1fob9te)

[圖 2 織布瑕疵影像處理流程圖[6]。 5](#_1t3h5sf)

[圖 3 初始化閥值最大值與最小值流程[6]。 6](#_2s8eyo1)

[圖 4 影像處理過後結果[6]。 7](#_3rdcrjn)

[圖 5 ResNet101 與優化過的 ResNet101 架構示意圖[7]。 8](#_lnxbz9)

[圖 6 透過語意分割逐像素抓出不同光源下的瑕疵[7]。 9](#_35nkun2)

[圖 7 模型運行速度[7]。 9](#_1ksv4uv)

[圖 8 本研究方法偵測流程圖。 10](#_2jxsxqh)

[圖 9 瑕疵資料增量流程圖。 11](#_1y810tw)

[圖 10 膠皮廠商提供的瑕疵影像。 12](#_2xcytpi)

[圖 11 架設於捲送機上方約 0.71 公尺之相機拍攝的影像。 12](#_1ci93xb)

[圖 12 像素填補方式 (a)常數值填補 (b)鄰近像素填補 (c)重複 (d)鏡射。 13](#_3whwml4)

[圖 13 不同明亮變化的瑕疵影像。 14](#_2bn6wsx)

[圖 14 YOLOv4 網路架構圖。 15](#_3as4poj)

[圖 15 ImproveYOLOv4 網路架構圖。 16](#_1pxezwc)

[圖 16 卷積運算示意圖。 17](#_2p2csry)

[圖 17 ReLU。 18](#_3o7alnk)

[圖 18 (a)Leaky Relu (b)mish。 18](#_23ckvvd)

[圖 19 CSPResNet 架構圖。 20](#_1hmsyys)

[圖 20 (a)原始網路架構 (b)殘差網絡架構。 21](#_2grqrue)

[圖 21 FPN 架構圖。 22](#_3fwokq0)

[圖 22 PANet 架構。 23](#_1v1yuxt)

[圖 23 (a)原始影像(1280 ×1024) (b)調整大小後的影像(608×608)。 24](#_4f1mdlm)

[圖 24 所改善之多尺度預測架構圖。 24](#_2u6wntf)

[圖 25 Inception 網路架構。 26](#_nmf14n)

[圖 26 GoogLeNet 網路架構圖。 26](#_37m2jsg)

[圖 27 捲送機設置環境樣貌。 27](#_2lwamvv)

[圖 28 Allied Vision Mako G-131C 工業相機[29]。 28](#_111kx3o)

[圖 29 (a)Kowa’s LM4NCL[30] (b)Fuji’s DF6HA-1S[31] (c)Kowa’s LM12NCL[32]。](#_3l18frh)

[. 29](#_3l18frh)

[圖 30 焦距 3.5 毫米鏡頭所拍攝影像。 29](#_206ipza)

[圖 31 焦距 6 毫米鏡頭所拍攝影像。 30](#_4k668n3)

[圖 32 焦距 12 毫米鏡頭所拍攝影像。 30](#_2zbgiuw)

[圖 33 相機建置環境示意圖。 31](#_3ygebqi)

[圖 34 偵測系統架構圖。 32](#_2dlolyb)

[圖 35 各種瑕疵影像 (a)變形 (b)損傷 (c)髒污 (d)色差 (e)褶皺 (f)壓痕。 33](#_3cqmetx)

[圖 36 瑕疵影像資料增量結果示意圖。 34](#_1rvwp1q)

[圖 37 瑕疵本體影像。 34](#_4bvk7pj)

[圖 38 (a)、(b) YOLOv4 偵測結果 (c)、(d) Improved YOLOv4 偵測結果。 35](#_1664s55)

[圖 39 (a)、(b) YOLOv4 誤判結果 (c)、(d) 所提方法辨識結果。 36](#_3q5sasy)

[圖 40 (a)變形 (b)損傷 (c)髒污 (d)色差 (e)摺皺 (f)壓痕。 37](#_25b2l0r)

[圖 41 (a)、(b)損傷 (c)變形 (d)色差。 38](#_kgcv8k)

[圖 42 (a)ground truth (b)偵測結果。 38](#_34g0dwd)

[圖 43 混淆矩陣。 39](#_43ky6rz)

[圖 44 IoU 示意圖 (a)IoU = 0.5 (b)IoU = 0.7 (c)IoU = 0.9。 40](#_4h042r0)

# 表目錄

[表 1 各鏡頭可拍攝範圍。 31](#_1egqt2p)

[表 2 資料增量對各個指標的影響比較。 41](#_2afmg28)

[表 3 各深度學習模型於瑕疵偵測辨識的評估比較。 42](#_pkwqa1)



 **緒論**

## 研究背景與動機

膠皮廣泛的被應用在我們生活中，在生產過程，膠皮可能會有上色不均、褶皺、損傷……等瑕疵產生，這些瑕疵品若沒被檢驗出來，繼續被送往後面的加工流程，會造成工廠產能與成本的浪費，倘若出貨產品內含有瑕疵，可能會導致被客戶退貨修改，更可能會需要重製新的膠皮，此舉嚴重甚至會直接影響到工廠的生產成本跟排程，因此，在生產過程中，瑕疵的檢測是很重要的一環，不僅能避免工廠成本的浪費，也能確保出貨產品的品質。

傳統的瑕疵檢測仰賴有經驗的專業人士，在檢測的過程中會受到檢測者的主觀判斷、個人經驗影響，導致檢驗標準不一致，此外，長時間的工時會導致偵測準確率下降，在生產時，機台運轉速度非常快，以人工偵測的速度很難去達成即時偵測的需求。膠皮工廠現階段就是以人工的方式在捲送機前進行瑕疵的檢測，如[圖 1](#_1fob9te)

所示，而在捲送機運轉速度每分鐘 33 公尺的轉速下，檢測產品的寬度為 1.63 公尺，以人眼的方式必然無法跟上機台運行的速度，除了捲送的速度快之外，要檢視的面積也非常廣，檢驗者只能夠概略性的檢測，這樣的方式非常不具效率，除了人工方式很難維持長時間高專注度來檢測之外，若增加人力也會對於整體生產成本有不少的影響。



圖 1 膠皮捲送機示意圖。

而自動化檢測的方式，目前大部分都是基於機器視覺搭配影像處理的方法來進行檢測，它的好處是不會影響檢測機台的運作，藉由影像中的統計特徵、結構特徵、光譜特徵來設計演算法，這些方法可以達到不錯的效果，也能節省人力，因此，目前的膠皮工廠需要比人工更有效率的檢測方式。以自動檢測的方式來進行卷送機上的瑕疵檢測，自動化除了能減少人力需求外，亦可以透過工業用相機達到在高速運行機台下的檢測，檢測能力也不會像人工方式因人員狀況而有疏忽、誤判和標準不一致的情況。

目前有許多以機器視覺搭配影像處理來檢測瑕疵的案例，像是 Liu 等人[[1]](#_48pi1tg)提出光譜減法用於積體電路上的瑕疵偵測、Tian 等人[[2]](#_2nusc19)提出基於小波轉換(Wavelet transform)的方法找出玻璃器皿上的裂痕或是 Kumar 等人[[3]](#_1302m92)提出以加伯濾波器 (Gabor filter)紋理材料進行瑕疵的檢測，這些都是使用影像處理方法來達到自動化的檢測目的。但是影像處理的方法容易受到外在環境影響，導致無法處理不同環境

下的影像，只要取像的環境因子一改變，這些方法中的參數就需要重新被調整，若影響嚴重甚至會需要重新設計檢測的方法。

由於膠皮瑕疵的種類、大小非常多樣，隨著產品的不同，膠皮紋理的差異性也非常大，這會導致非常困難在不同紋理膠皮上偵測瑕疵，因為工廠內機台處於不同的位置及環境，拍攝出來影像的照明皆會有所不同，相較於影像處理的方法，參數設定容易受到環境因素的影響，使用深度學習處理不同環境的影像能有更好的適應性。因此，在本研究中，採用機器視覺搭配深度學習的方式，來學習各種不同樣式瑕疵的樣貌特徵，並在卷送機上方架設工業用相機，進行膠皮瑕疵的偵測。

## 論文架構

本論文的架構主要分成五個章節，各個章節的內容如以下所述。第一章為緒論，介紹關於膠皮工廠相關的瑕疵偵測背景與研究動機。第二章為文獻探討，本研究會敘述過去針對瑕疵檢測的方式，並分析各個方法的優點與缺點。第三章為本篇論文實驗的研究方法，介紹本研究針對瑕疵訓練影像的資料增量處理、以及採用的深度學習架構。第四章為實驗結果與討論，說明本研究的實驗環境、使用的相關設備與設備架設的方式，並將所提方法實測結果進行各種指標的比較。第五章為結論，並提出未來期望與發展。

# 文獻探討

目前膠皮工廠是以人工的方式對捲送機上的膠皮進行檢測，而針對瑕疵檢測的方法種類繁多，因此本研究將利用下列章節來介紹人工檢測、使用機器視覺搭配影像處理自動化檢測、及機器視覺搭配深度學習(Deep Learning)自動化檢測的相關文獻，並探討各自的優點與缺點。

## 人工檢測

在工廠的產線，使用人力的方法來對產品進行品質的控管是最常見的方式，在機台運轉下進行檢測，需要很高的專注度，隨著檢測的時間拉長，檢視的人員勢必很難一直維持高專注度，所以檢測的人員會常需要輪替以確保檢測的品質，因此使用人力檢測是非常高成本且沒效率的方法。

Nisha 等人[[4]](#_3mzq4wv)提到，傳統纖維工廠是以人眼的方式進行瑕疵檢測，而一個具經檢的檢測者大約只能檢測出 70%的瑕疵。See 等人[[5]](#_2250f4o)針對 82 名檢測人員進行實驗，這些受測者需要對精密製造零件外觀進行可靠性的檢察，實驗結果有 85%的瑕疵物品被受測員檢驗出來，而當中有 35%的良品被受測員誤判為不良品，實驗中發現多數受測者的表現與壓力及工作量無直接關係，但檢測的環境因子，像是照明會對人工檢測準確率有很大的影響，另外年長受測者相較於年輕受測者的準確率來的低，並且實驗也說明隨著檢測的時間增加及精力的消耗，誤判的比例也會隨之提高。

在對物品進行檢測時，容易受到每個人不同主觀判斷，導致檢測標準不同，檢測的環境的變化也會影響到判斷的準確率，而檢測者的表現會因為個人狀況而會有疏忽、誤判的可能性，除此之外，人員的汰換必定會是個問題，在短時間很難訓

練出一個具有豐富判斷經驗的檢測員，所以使用人力的方式進行檢測是沒有效率且高成本的。

## 機器視覺搭配影像處理

Cho 等人[[6]](#_haapch)提出了基於機器視覺搭配影像處理的方法，他們的影像處理流程如下[圖 2](#_1t3h5sf) 所示，透過在捲送機上方架設線掃描相機進行取像，來自動化檢測織布產品。

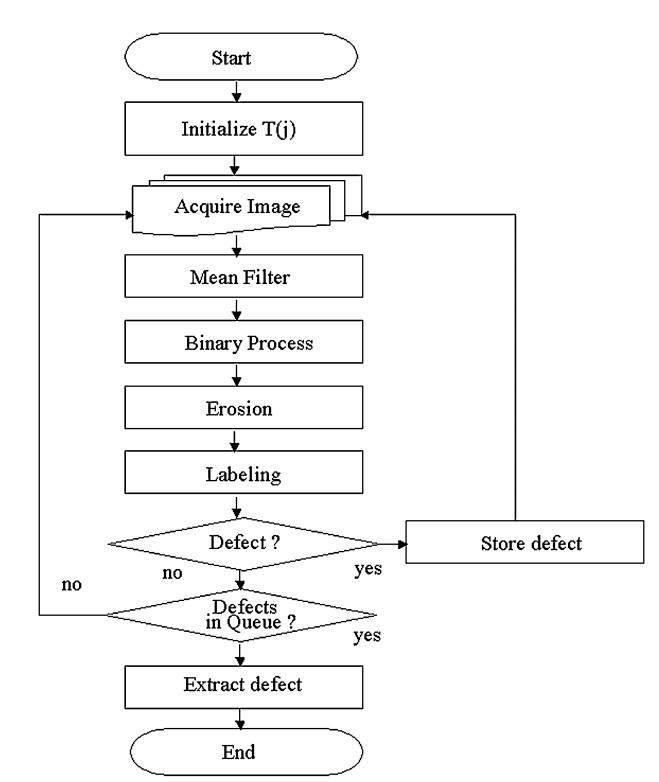


圖 2 織布瑕疵影像處理流程圖[[6]](#_haapch)。

由於在取像的過程中會受到環境的照明變化、亮度分配不平均還有織物的紋理等因素影響而產生雜訊，因此要先對這些取得的影像進行前處理，此篇論文使用中值濾波器(Median filter)藉由調整閥值來降低這些雜訊的干擾，使用中值濾波器的

好處在於它在消除雜訊的同時能夠保留影像中的邊界資訊。由於影像中的亮度分配不平均，如果使用固定的閥值於整個影像上會發生各種錯誤，所以這篇論文使用自適應閥值來解決此問題，定期更新閥值並應用在整個影像上，藉由每行 x 軸方向上的每個像素計算平均值和標準差找出最大𝑇1和最小𝑇2的閥值，如下公式[( 1 )](#_4d34og8)所示，其中𝛼∗為權重，並定期更新。

| 𝑇1,2(𝑗) = mean[𝐿(𝑗)] ± 𝛼∗mean[𝑠𝑡𝑑(𝐿(𝑗))] | ( 1 ) |
| --- | --- |

L(𝑗)是一條無包含瑕疵的 x 軸像素亮度值，[圖 3](#_2s8eyo1) 所示，當偵測開始時𝑇1和𝑇2會被計算出來，並透過公式[( 2 )](#_17dp8vu)的限制，將輸入像素 f(𝑖, 𝑗)進行二值化處理，符合在閥值規範內的設為 1，超出閥值的設為 0，其意義代表 1 為沒有瑕疵，而 0 為有瑕疵或是雜訊。

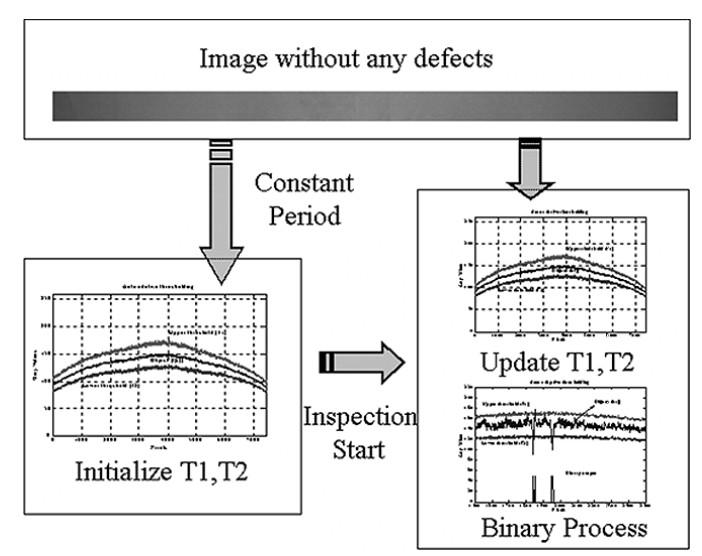
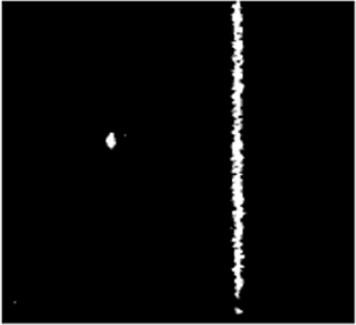
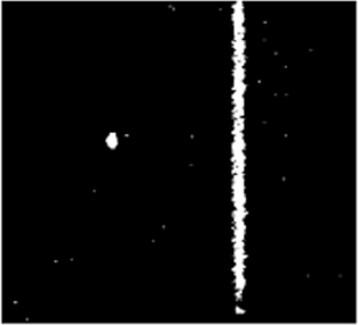
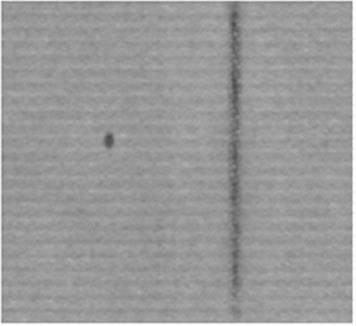


圖 3 初始化閥值最大值與最小值流程[[6]](#_haapch)。

| 𝑖𝑓 𝑇1 < 𝑓(𝑖, 𝑗) < 𝑇2 𝑡ℎ𝑒𝑛 𝐵(𝑖, 𝑗) = 1 𝑒𝑙𝑠𝑒 𝐵(𝑖, 𝑗) = 0 | ( 2 ) |
| --- | --- |

雖然使用中值濾波器可以過濾掉高頻的雜訊，影像中仍有可能會殘留其它雜訊，這些剩餘的雜訊可以使用侵蝕(Erosion)來消除，每個步驟處理的結果如下[圖 4](#_3rdcrjn)所示。



| (a) | (b) | (c) |
| --- | --- | --- |

圖 4 影像處理過後結果[[6]](#_haapch)。

使用影像處理的方法，可以達到即時的運算而且設備成本也不高，但是針對閥值參數的設定可能會忽略掉一些型態較為不同的瑕疵，譬如形狀較小的瑕疵和與背景顏色相近的瑕疵，此外，若是環境因子一改變，有的演算法也需重新做對應的設計，而且偵測出來的瑕疵也無法去判斷它屬於什麼類別的瑕疵，也無法適用在紋理較為複雜的織品上，因此，使用影像處理的方式來抓取瑕疵適應性並沒有非常高。

## 機器視覺搭配深度學習

使用影像處理的方式雖然能檢測出瑕疵，但是如果瑕疵種類繁多且背景結構複雜，就無法很精確的被影像處理方法偵測出來。隨著近年來深度學習的崛起，卷積神經網路(Convolutional neural network, CNN)的出現對於機器視覺有很重大的影響，它可以透過輸入影像資料來進行訓練，藉由卷積層萃取出影像中的特徵，

並將這些特徵作為輸入至後一層的資料，直到整個網路的末端，透過損失函數 (Loss function)來計算與真實答案的誤差，並透過反向傳播(Backpropagation)來更新各層網路神經的權重，CNN 常見的應用有影像辨識、物件偵測(Object detection)或是影像的語意分割(Semantic segmentation)。

深度學習的發展在一開始，透過影像辨識，能讓我們從中了解影像內所含的物件是什麼，然而，當影像中出現了多個物件，就必須對每個物件進行區分和辨識，進而發展出了物件偵測和語意分割。語意分割和物件偵測相比，雖然兩者都能尋找出物件的位置(Localization)，但是以語意分割更為精確，因為它能達到像素級別(Pixel-level)的辨識，Zhang 等人[[7]](#_319y80a)提出一個基於 Mask RCNN 來對車輛損傷進行語意分割偵測的方法，並優化了 Mask RCNN 中 ResNet101 的架構，如[圖](#_lnxbz9) [5](#_lnxbz9)，藉由優化過的模型來進行車輛的損傷語意分割，如[圖 6](#_35nkun2)，它對於不規則形狀的目標和不同環境光源的影像有較好的適應性，但是運行速度卻無法達到即時，如下所[圖 7](#_1ksv4uv) 示。

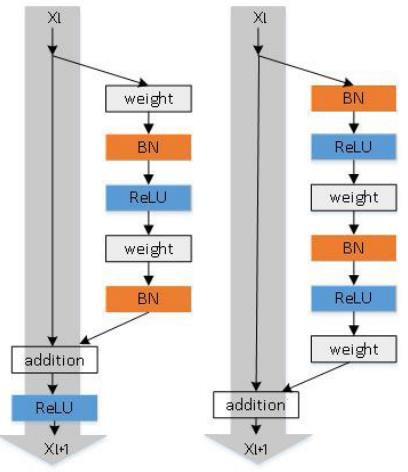


圖 5 ResNet101 與優化過的 ResNet101 架構示意圖[[7]](#_319y80a)。

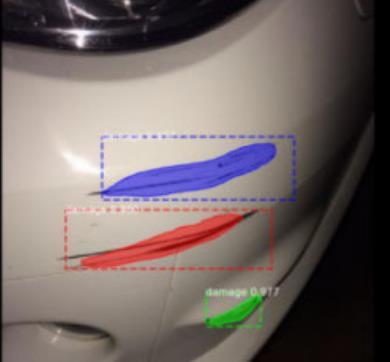
 

圖 6 透過語意分割逐像素抓出不同光源下的瑕疵[[7]](#_319y80a)。

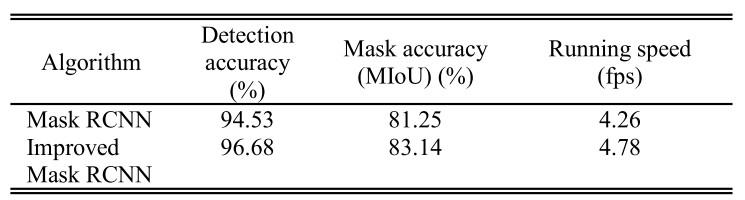


圖 7 模型運行速度[[7]](#_319y80a)。

使用深度學習的方法在訓練時可能存在過擬合(Overfitting) 、梯度消失 (Vanishing gradient)、梯度爆炸(Exploding gradient) 等問題，因此需要使用大量的

資料來對模型進行訓練，深度學習的方法能克服背景複雜或是物件型態不固定導致無法被偵測出來的問題，相較於影像處理的方式能有更好的適應性。雖然語意分割相較於物件偵測的定位能更加精確，但是大多數的語意分割模型很難實現即時 (Real-time)的偵測，而物件偵測的方法在運行速度上相較於語意分割還來的快，因此，本研究採用物件偵測的模型 YOLOv4，來對運轉中的捲送機進行膠皮瑕疵的位置偵測，接著把偵測到的瑕疵主體影像輸入給第二個深度學習模型 GoogLeNet，進行瑕疵主體類別的辨識，以提高偵測的準確率。

# 研究方法

本研究是針對在運轉中的 PVC 捲送機進行即時的瑕疵檢測，由於瑕疵有各式各樣的形態和大小，若只單純使用物件偵測模型，形狀較小的物件會很難被偵測到，而且存在容易分類錯誤的問題，因此，在研究中使用了物件偵測模型 YOLOv4[[8]](#_1gf8i83)並優化針對不同尺度的瑕疵偵測能力，再結合分類器 GoogLeNet[[9]](#_40ew0vw)進行瑕疵的辨 識，藉此提高準確率。[圖 8](#_2jxsxqh) 為本研究在進行偵測的流程圖，本章 3.1 節會先介紹在

有限的原始資料下，增加可使用的瑕疵資料，接著 3.2 節會介紹使用優化過後的深度學習方法進行瑕疵的偵測，並於 3.3 節介紹以GoogLeNet 進行瑕疵的類別辨識。

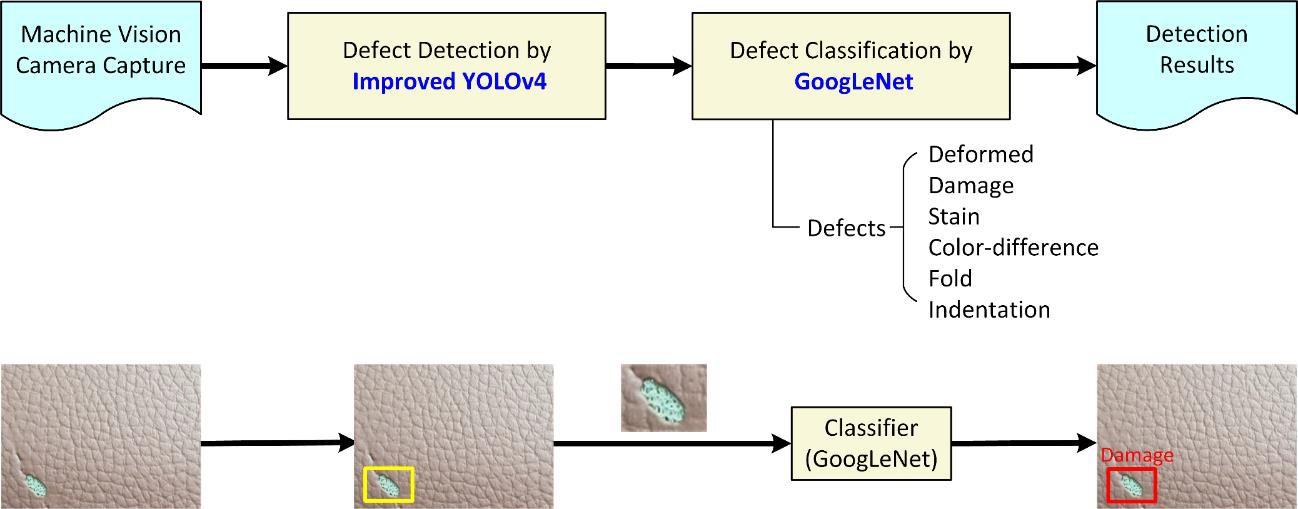


圖 8 本研究方法偵測流程圖。

## 瑕疵資料增量

膠皮工廠過去的作法，皆是在捲送機前以人工的方式來檢視是否有瑕疵，而瑕疵樣本通常是在檢測時發現才會記錄，所以廠商僅能提供目前手邊現有的瑕疵影像，數量非常有限，而這些影像中大多數都是以手持裝置進行拍攝，每張影像的拍攝角度、距離、照明都不相同，且提供的影像大多都是針對瑕疵進行局部的拍攝，

因此在視野上也與實際架設的工業用相機拍攝的畫面差異非常大，本章節的目的，是使用有限的資料來生成更多近似工業用相機拍攝的影像，克服訓練資料量不足的問題。

### 影像資料增量流程

下[圖 9](#_1y810tw) 為瑕疵增量的流程圖，首先本研究會針對瑕疵影像進行隨機的水平垂直移動、旋轉、形變，由於廠商提供的影像都是以近距離進行拍攝，與實際使用工業用相機所拍攝到的影像視野非常不同，所以須要再對影像進行縮小。而由於形變和縮小，影像中部分地方會有空白的區域產生，因此需要針對這些空白區域進行填補，另外，工業用相機實際拍攝的影像與廠商提供影像的照明也有所差異，所以最後必須再針對這些影像進行亮度的調整，藉此獲得不同光源下的各種瑕疵模樣。

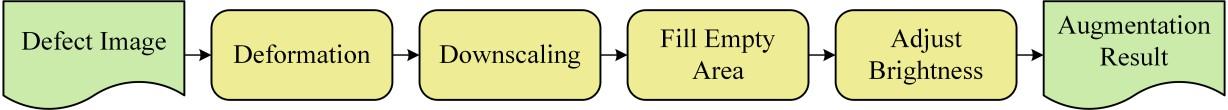


圖 9 瑕疵資料增量流程圖。

### 資料增量方法論述

廠商所提供的原始瑕疵影像大多都是以小範圍近距離的拍攝，如[圖 10](#_2xcytpi)，與實際架設在捲送機上方約 0.71 公尺的工業用相機進行拍攝的視野差異非常大，如[圖 11](#_1ci93xb)，於是本研究透過將變形後的瑕疵影像縮小 8 至 16 倍，使得瑕疵影像趨近於工業用相機拍攝到的影像視野。

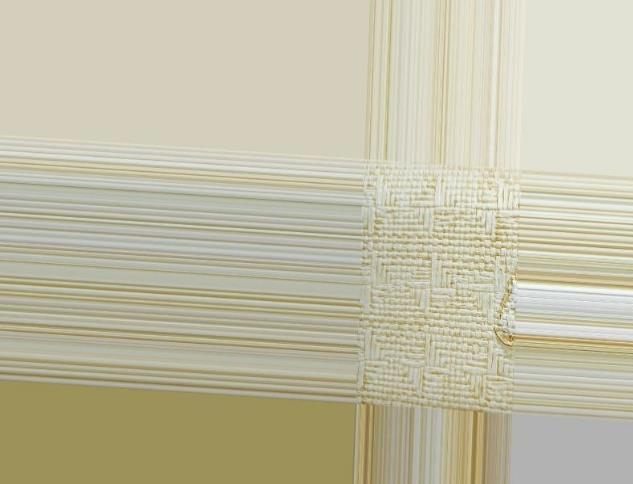


圖 10 膠皮廠商提供的瑕疵影像。

2-result

圖 11 架設於捲送機上方約 0.71 公尺之相機拍攝的影像。

在對原始影像進行隨機的旋轉、水平垂直移動、形變時，會有部分的空缺區域需要進行像素的填補，而常使用的填補像素方式有常數值填補、鄰近像素填補、重複、鏡射，各種方式結果如[圖 12](#_3whwml4)，(a)常數值填補周圍像素可由某一個固定值補充，可以看見(b)若使用鄰近像素填補，會發生瑕疵邊界被異常延伸的情況，相較於以上兩種方式，以重複(c)或是鏡射(d)會是更好的選擇，瑕疵的輪廓不會被不合理的延長，影像中所含的物件數量也有明顯的增加，在這邊本研究選用的填補方式為鏡射，因為它能使影像在填補時的物件比重複填補有更多元的變化，也可使模型在訓練上有更好的表現。



| (a) | (b) |
| --- | --- |
| (c) | (d) |

圖 12 像素填補方式 (a)常數值填補 (b)鄰近像素填補 (c)重複 (d)鏡射。

捲送機位置的照明會因為時間而有所變化，所以在資料增量的最後階段，使用隨機的方式對影像的亮度來增強或是減弱，如[圖 13](#_2bn6wsx)，產生出不同亮度的瑕疵影像，使本研究的模型在訓練出來針對亮度的影響能有更好的適應性。



圖 13 不同明亮變化的瑕疵影像。

## 深度學習物件偵測

隨著深度學習近幾年來的發展，物件偵測的方法已經非常成熟，物件偵測又可分為兩類 One-stage 和 Two-stage，早期的方法是以 Two-stage 為大宗，像是 R-CNN [[10]](#_2fk6b3p)、fast R-CNN[[11]](#_upglbi)和 faster R-CNN[[12]](#_3ep43zb)，其物件位置的偵測和分類是分開進行，造成運行速度上有一定的限制，為了改善上述運行速度的問題，One-stage 的作法則是將物件位置的偵測和分類同時進行，像是 RetinaNet[[13]](#_1tuee74)、SSD[[14]](#_4du1wux)和 YOLOv4。本研究是使用Alexey Bochkovskiy[[15]](#_2szc72q)預先訓練好的模型權重yolov4.conv.137，

當作接下來要訓練瑕疵偵測模型的初始權重，透過使用先訓練過的模型權重，可以使訓練時提取特徵與修正誤差更有效率，並漸少模型訓練所花費的時間，由於 PVC瑕疵有大小上的差異，為了防止較小的瑕疵無法被偵測出來，本研究在 YOLOv4進行預測的部分多加入一個尺度(scale)的偵測分別為 152×152、76×76、38×38、19

×19，其中 152×152 用來偵測較小的瑕疵，19×19 用來偵測較大的瑕疵，[圖 14](#_3as4poj) 為原

YOLOv4 深度學習網路架構圖，[圖 15](#_1pxezwc) 為所提方法 Improved-YOLOv4 的深度學習網路架構。

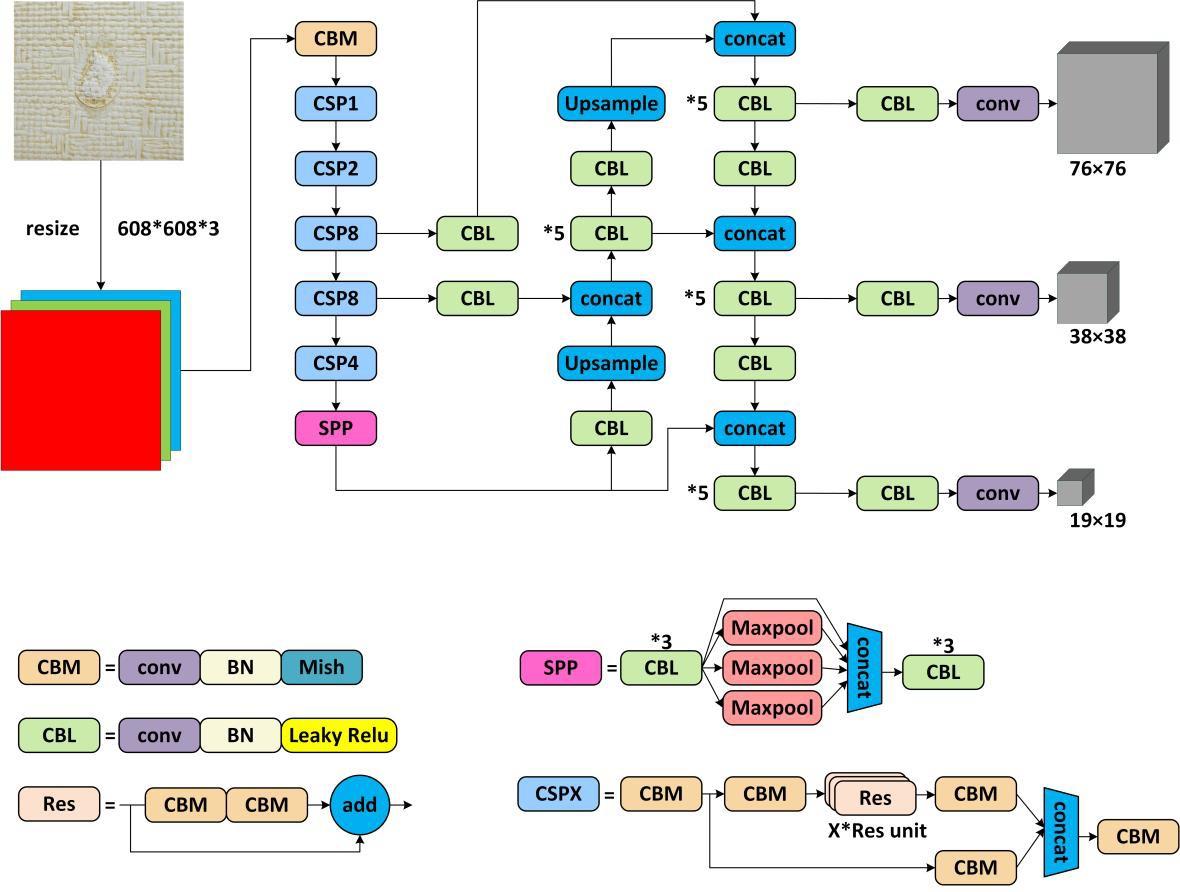


圖 14 YOLOv4 網路架構圖。

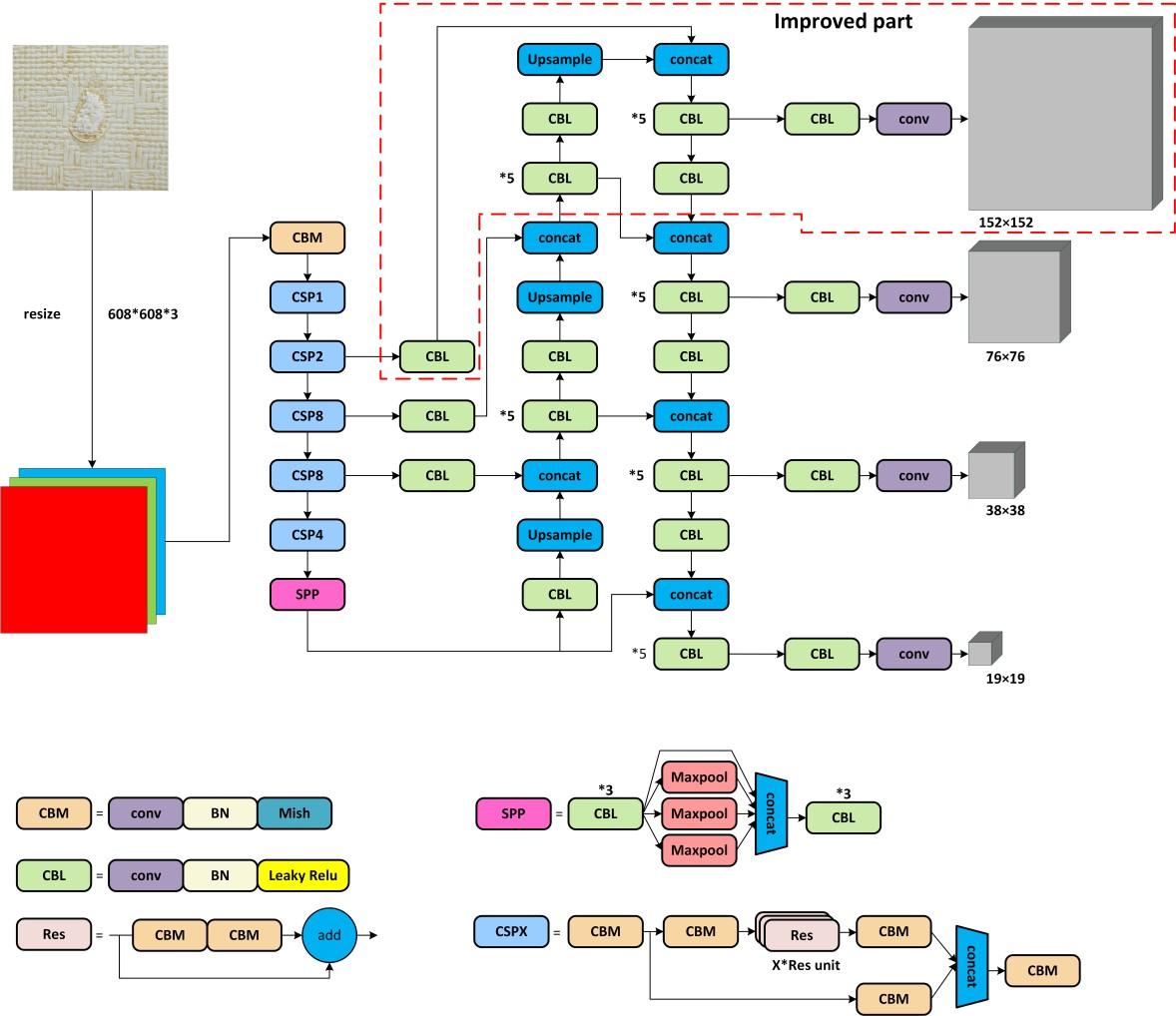


圖 15 ImproveYOLOv4 網路架構圖。

### 卷積層(Convolutional layer)

卷積層的卷積運算對於特徵提取中是一個非常重要的步驟，它是使用遮罩 (Filter 或稱 Kernel map)對影像中的所有像素位置進行卷積計算。運算分成兩個步驟，分別是滑動與內積，如[圖 16](#_2p2csry) 所示，遮罩會對覆蓋的區域進行矩陣的內積運算，而運算過後的結果即為特徵圖(Feature map)。在深度學習中這些遮罩的初始值會被隨機指派，透過訓練時的損失函數來進行反向傳播，並更新卷積層中遮罩的數值，讓遮罩能更有效的萃取出影像中的特徵。

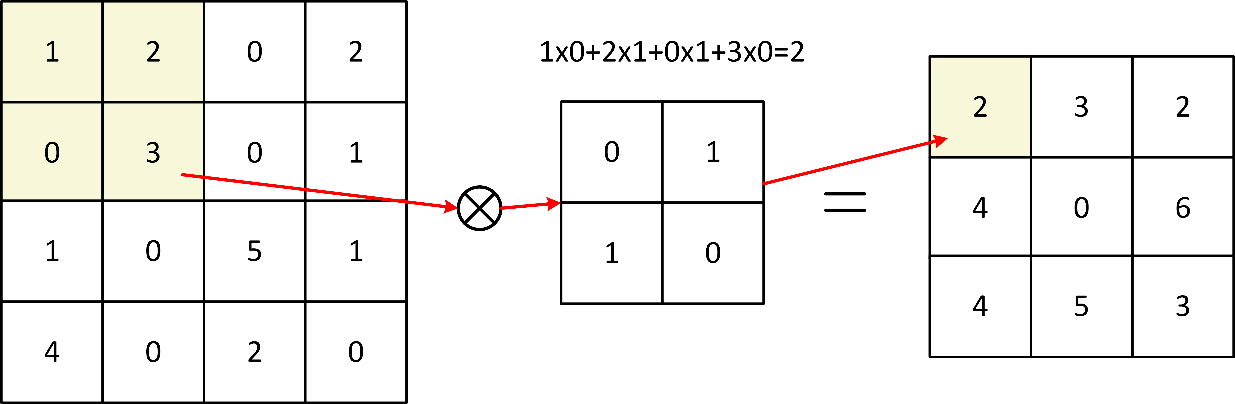


圖 16 卷積運算示意圖。

### 激勵函數(Activation function)

在深度學習模型中，如果不使用激勵函數，每層神經網路皆是以上層輸出的線性組合作為下一層的輸入，使得輸出與輸入只存在著線性關係，所以若沒有使用非線性激勵函數，訓練出來的深度學習模型便沒有辦法從訓練資料中學習到較好的結果。激勵函數常被應用在模型中的神經元與神經元之間，透過反向傳播 (Backpropagation) 來更新權重的參數， 在深度學習中 Rectified Linear Unit (ReLU)[[16]](#_184mhaj)最為廣泛被用來當作激勵函數，由於它在運算上較其它激勵函數快，使模型在訓練時可以以較少的時間達到收斂的效果，但是 ReLU 卻存在著梯度消失的問題，因為當輸入值小於 0，會導致參數無法被有效更新，如[圖 17](#_3o7alnk) 所示，這種狀況稱為Dead ReLU，因此在本研究的方法中，使用Leaky ReLU[[17]](#_3s49zyc)和Mish[[18]](#_279ka65)來當作模型的激勵函數，解決負值無法被更新的問題，其函式分佈如下[圖 18](#_23ckvvd) 所示。

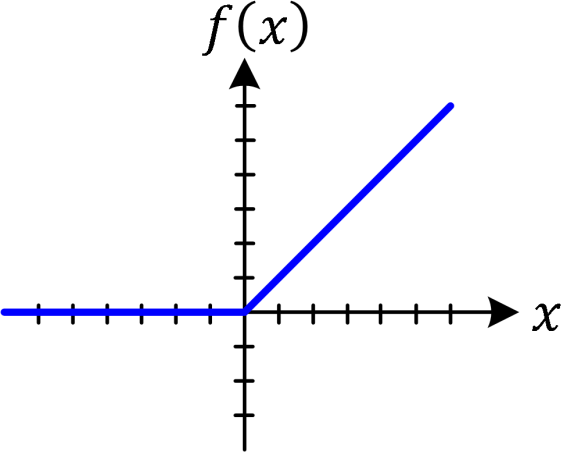
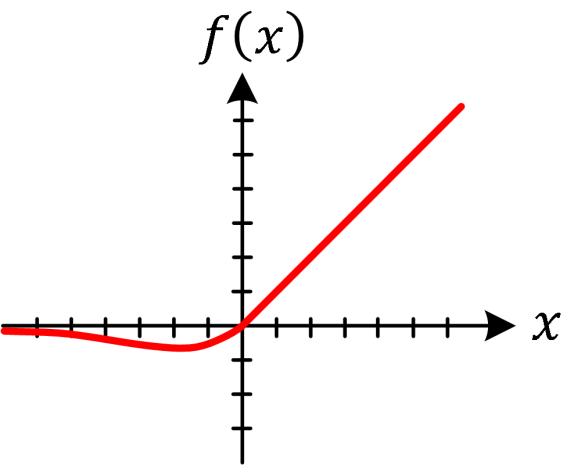
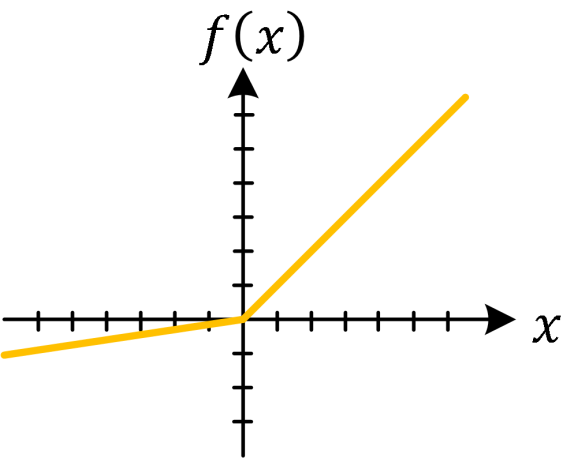


圖 17 ReLU。



| (a) | (b) |
| --- | --- |

圖 18 (a)Leaky Relu (b)mish。

### 批次正規化(Batch Normalization)

在深度學習模型中，如果要使模型的準確度提高，最常見的做法就是加深網路的深度，但是越深的模型越難以訓練，隨著層數增加，在訓練時，每一層的參數都會受到前面所有層數的參數影響，造成每層的數據分布不一致，導致神經網路要不段去學習新的參數分布，而這些變化量會隨著網路深度越深而越大，使的模型在訓練上變得困難，並使訓練時收斂的速度變慢，這個現象稱為內部協變量偏移 (Internal Covariate Shift)，早期的作法，只能透過調整較佳的學習率和初始的權重

來解決這個問題。

本研究採用 Ioffe 等人[[19]](#_meukdy)所提出的批次正規化，它是將每層網路的輸入進行正規化到平均值為 0 與標準差為 1 的常態分佈，透過在激勵函數之前，正規化能夠有效的激活傳遞資訊不會出現飽和的情況，讓模型有效學習特徵分布，這個方法可以減少內部協變量偏移的問題，也能解決網路梯度消失的問題，讓訓練時模型收斂的速度提升。

### 跨階段局部網路(Cross Stage Partial Network)

YOLOv4 優化了 YOLOv3[[20]](#_36ei31r)原本使用的 backbone 架構，在保證執行速度的同時，大幅提升了模型的準確度，YOLOv4 採用 Wang 等人[[21]](#_1ljsd9k)所提出的 Cross Stage Partial Network(CSPNet)的概念，將原本的Darknet53 替換為CSPDarknet53，藉由將輸入的特徵圖分為兩部分，其中一半的特徵圖不進行卷積運算，直接作為下一層運算的輸入，而剩下的另一半特徵圖則進行原先預定的卷積運算，最後再將兩部分的特徵圖串接合併，獲取更豐富的梯度融合資訊，如[圖 19](#_1hmsyys) 所示，它的優點是能將模型輕量化來降低運算量，加快模型訓練速度，並讓整體的準確度有所提升。

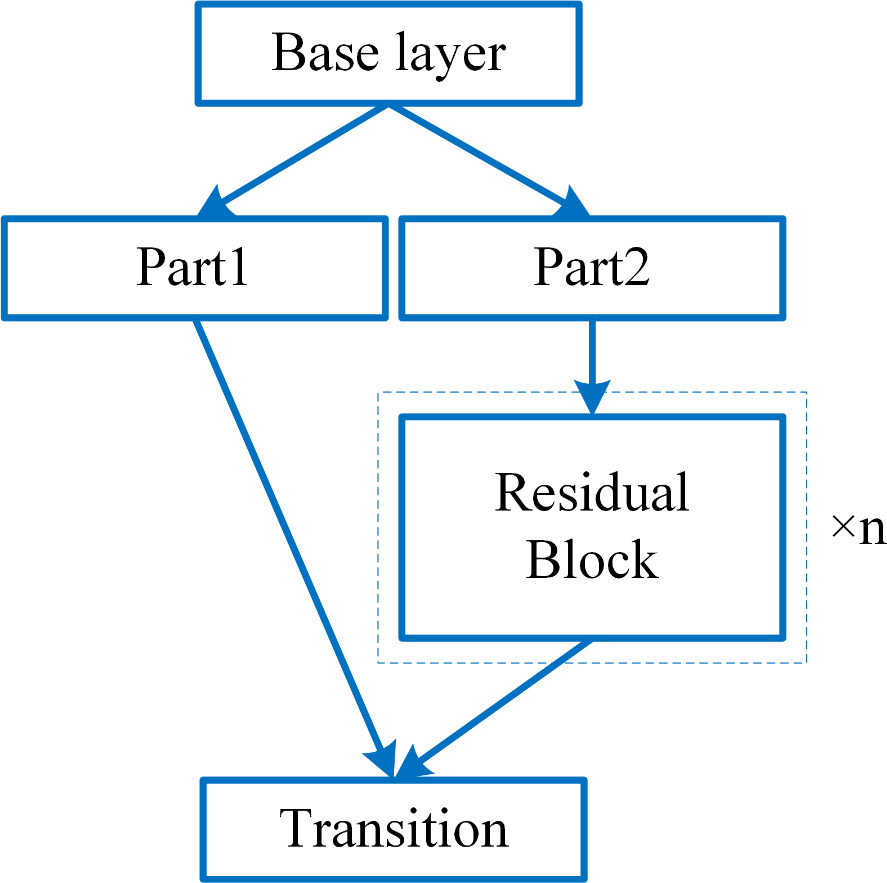
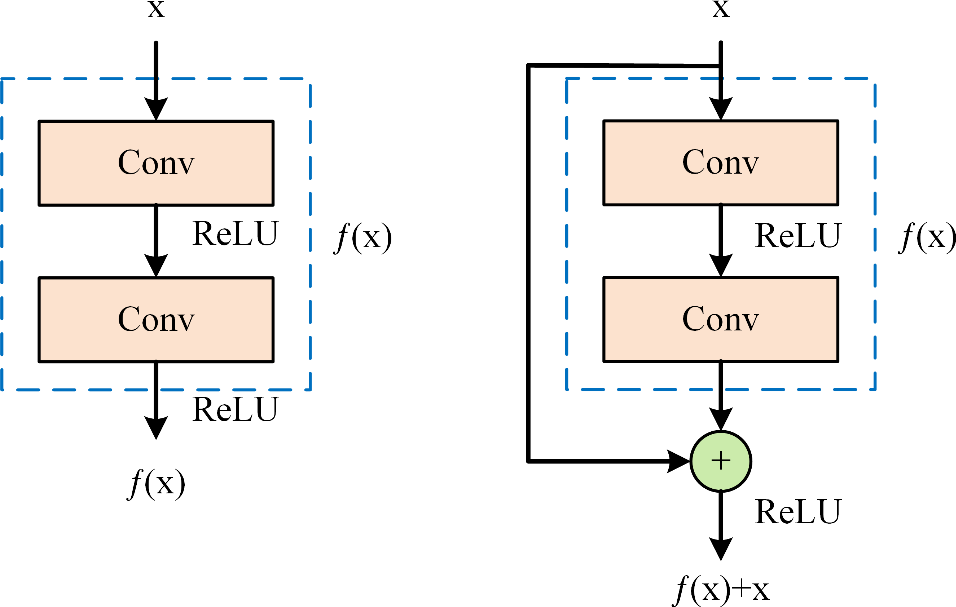


圖 19 CSPResNet 架構圖。

### 殘差網絡架構(Residual block)

在深度學習中，各個網路模型不斷朝著越來越多層的方向發展，但只是簡單的加深網路的層數，不一定會達到較好的效果，反而可能使預測的結果更差，因為會牽涉到模型在深層的情況下容易造成梯度消失的問題，雖然有時候能透過正規化來解決這個問題，使模型可以繼續收斂，但是當深層模型能夠有效收斂的時候，卻產生了另一個問題，退化(Degradation)，當網路深度增加的時候，準確度會到達飽和並且快速地下降，因為在較深層網路模型參數的誤差造成的影響會越來越大，而 Kaiming He 等人[[22]](#_45jfvxd)提出了殘差學習網路的架構來解決這個問題，透過捷徑連接(Shortcut connections)的方式增加傳播梯度，如下[圖 20](#_2grqrue) 所示，將 x 與卷積過後的結果相接，可以讓深層網路模型的參數得到有效的更新。



| (a) | (b) |
| --- | --- |

圖 20 (a)原始網路架構 (b)殘差網絡架構。

### 多尺度預測(Multi-scale prediction)

為了解決多尺度的偵測問題，Tsung-Yi Lin 等人[[24]](#_2koq656)所提出的特徵金字塔網路 (Feature Pyramid Network, FPN)，下[圖 21](#_3fwokq0) 所示，透過充分利用卷積神經網路在各個階段進行卷積時產生出的特徵圖，結合了上至下(Top-down)和捷徑連接(Shortcut connection)的概念，在上至下階段使用上採樣(Upsample)後，接著使用捷徑連接與淺層卷積產生的特徵圖連接(Concatenate)，來獲得較好的預測結果。

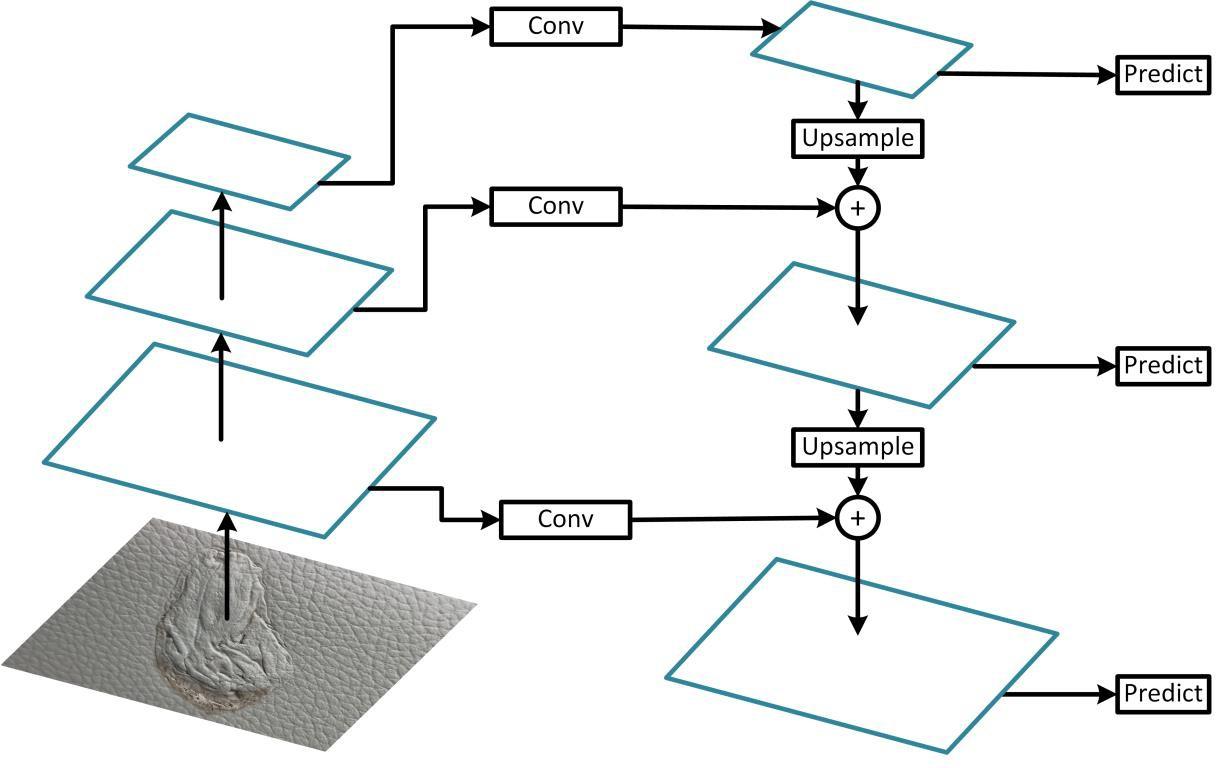


圖 21 FPN 架構圖。

而 Shu Liu 等人[[25]](#_zu0gcz)以特徵金字塔網路的概念，提出 Path Aggregation Network (PANet)，藉由增加了下至上階段(Bottom-up)來進行下採樣(Downsample)，下[圖](#_1v1yuxt) [22](#_1v1yuxt) 所示，並透過與前一個上至下階段的特徵圖做捷徑連接，讓較深層的網路也可以具有淺層的特徵資訊， 以提高各種尺度的預測準確率，在本研究中採用了 PANet 的方法，並使用了 4 個尺度來對瑕疵進行偵測。

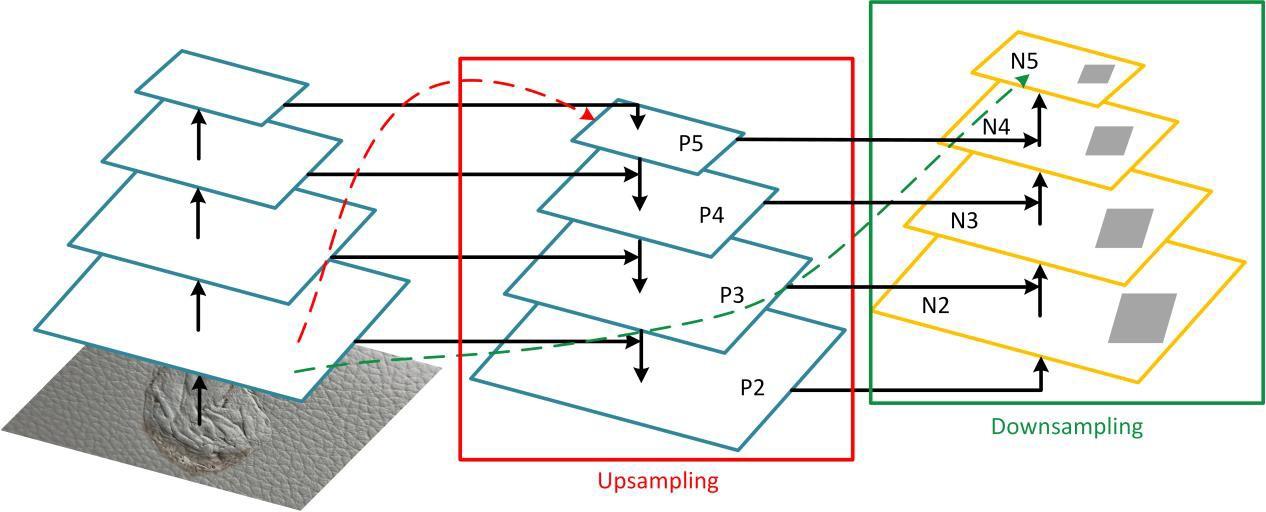
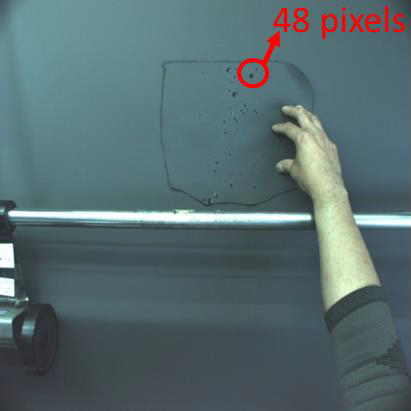


圖 22 PANet 架構。

在 CSPDarknet53 進行卷積運算時，卷積越深層的特徵圖會擁有比較豐富的語意資訊，而越淺層的特徵圖則會擁有較為豐富的物件空間資訊，而這些特徵會隨著卷積運算到更深層的網路，特徵圖中的空間資訊會逐漸消失。

由於本研究中瑕疵的大小尺寸不同，其中有些瑕疵非常細小，影像在輸入至 YOLOv4 前須要先調整大小至608 × 608，那麼原影像上的瑕疵大小也會因此被壓縮，導致瑕疵在整張影像上所佔的畫素變少，如下[圖 23](#_4f1mdlm) 所示，如果以原本 YOLOv4的 3 個尺度來進行偵測，就會無法偵測到較小的瑕疵，因此本研究針對原本的 YOLOv4 多增加了 1 個尺度來進行小尺度瑕疵的偵測，分別為152 × 152、76 × 76、 38× 38、19× 19，其中152 × 152用來偵測較小的瑕疵，所改善的多尺度預測架構圖如[圖 24](#_2u6wntf) 所示。

| (a) | (b) |
| --- | --- |

圖 23 (a)原始影像(1280 ×1024) (b)調整大小後的影像(608×608)。

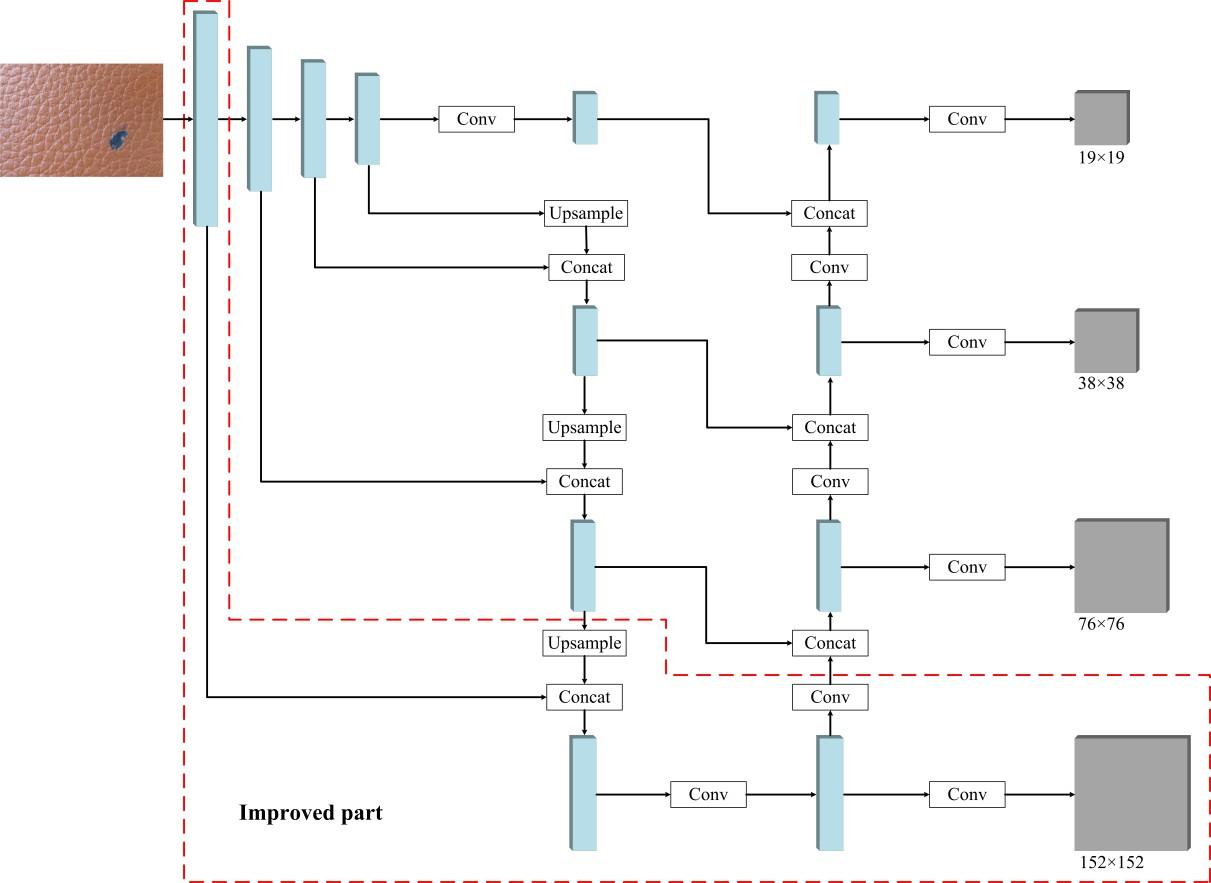


圖 24 所改善之多尺度預測架構圖。

## 分類器(Classifier)

PVC 瑕疵在經過前一階段的多尺度預測後，結果會接著由分類器來進行辨識，本研究總共將瑕疵分為 6 類，分別是變形、損傷、髒污、色差、褶皺和壓痕。分類器的種類非常多元，而本研究選擇了一個較輕巧的深度學習模型來當作分類器進行辨識，因為如果選用一個非常深層的網路模型，雖然可能會有更好的分類能力，但是由於參數過多會較難訓練，運行的速度也會相對下降，而 GoogLeNet 相較於其他著名的分類器 VGG16[[26]](#_3jtnz0s)與 AlexNet[[27]](#_1yyy98l)參數量小 20 倍和 9 倍，卻擁有很好的分類效果，由於需要分辨的種類只有 6 類，因此本研究採用 GoogLeNet 作為分類器，來達成辨識的任務。

### 模型訓練流程

由於訓練的資料有限，本研究透過資料增量產生出的影像來訓練分類器，使在訓練模型時能有更好的泛化能力，透過第一階段訓練好的深度學習物件偵測模型，取樣出局部的瑕疵主體影像，隨後將這些瑕疵本體當作資料集，將影像調整為 128 × 128的大小作為 GoogLeNet 的輸入，來進行對瑕疵主體的訓練和辨識。

### 分類器網路架構

一般的分類器為了達到更好的準確度，最直觀的作法就是加寬或加深網路的深度，但是當資料有限時，如易造成參數過擬合(overfitting)，而且網路越複雜，相對的運算量也較大，因此 Christian Szeged 等人[[9]](#_40ew0vw)提出了 Inception 的架構，如下[圖](#_nmf14n) [25](#_nmf14n) 所示，透過不同大小的卷積運算以及最大池化層(Max pooling)萃取出不同的特徵，最後將這些特徵圖連接起來，藉此增加了網路的寬度，這種方式能夠擷取出更多影像中的特徵與細節，可以看到圖中1 × 1的卷積主要目的是在降維，藉此限制

模型的參數，並降低運算量。

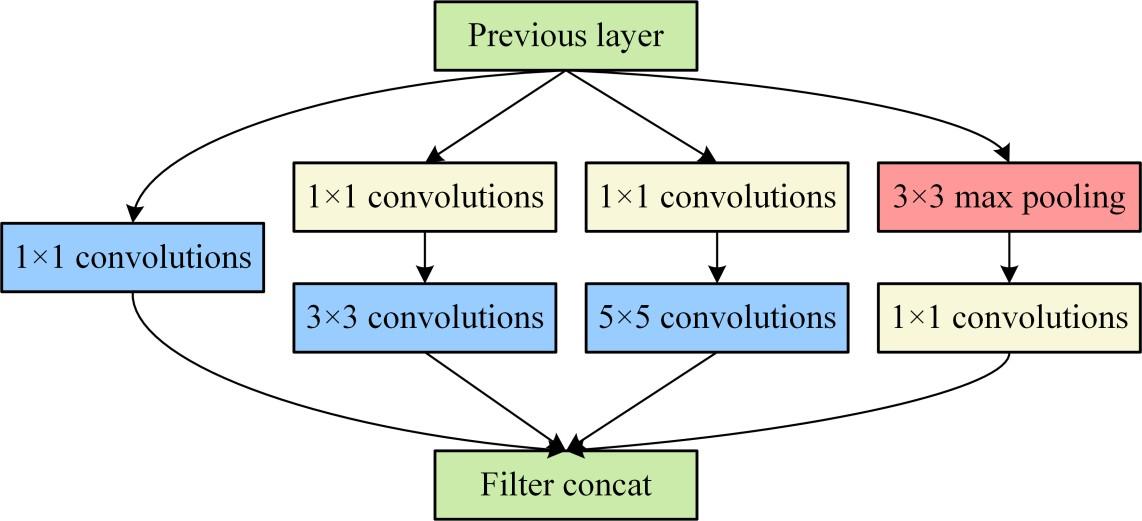


圖 25 Inception 網路架構。

GoogLeNet 的組成總共使用了 9 個 Inception 的架構，模型架構圖如下[圖 26](#_37m2jsg)所示，另外，GoogLeNet 採用了 Network In Network[[28]](#_4iylrwe)的概念，在全連接層之前加入了全域性平均池化(Global average poling)，降低參數的數量，此外，在網路中加入了 2 個輔助分類器(Auxiliary classifiers)，其作用為避免梯度消失，在訓練的過程中，透過模型的 2 個輔助分類器計算額外的損失函數，增加了反向傳播的梯度訊號，加快訓練時收斂的速度。在前一階段偵測出瑕疵所在位置後，將局部瑕疵影像輸入 GoogLeNet，最後輸出 6 種瑕疵的辨識結果。

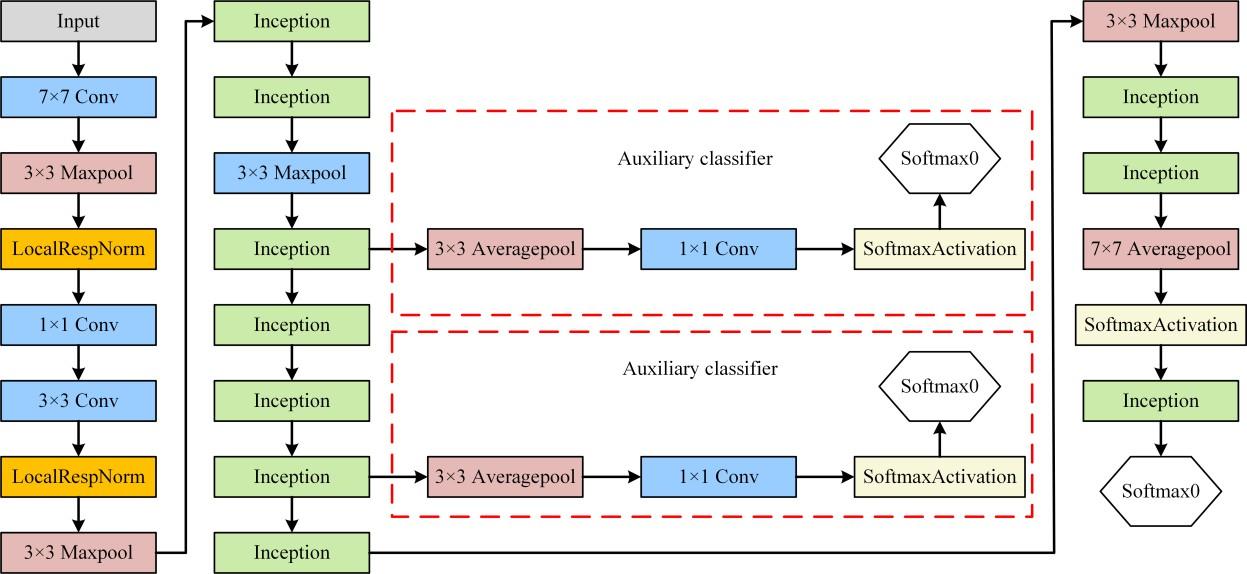


圖 26 GoogLeNet 網路架構圖。

# 實驗結果與討論

在此章節中，將說明工廠機台相關的設備環境建置與模型的訓練與評估，主要分成五個部分，4.1 節說明安裝在捲送機上方的相機與相關設備以及運行深度學習模型的伺服器規格，4.2 節說明深度學習訓練使用的資料集，4.3 節呈現實驗結果之影像，4.4 說明使用評估指標，4.5 所提方法的結果比較。

## 工廠環境與實驗裝置

由於檢測環境是在膠皮工廠的生廠線上，該區域有著許多其他生產機台，加上工廠環境屬於開方式區域，溫度及濕度並不穩定且容易產生粉塵，並不適合放置精密電腦設備，因此本研究採用可透過乙太網路傳輸的工業相機，如此一來就可透過線路傳送資料回至適合放置精密電腦的室內環境，並簡化現場檢測設備至僅存工業用相機，捲送機設置環境樣貌如[圖 27](#_2lwamvv) 所示。



圖 27 捲送機設置環境樣貌。

捲送機在運行時的速度可達每分鐘 33 公尺，而本研究需要在捲送機運轉的同時進行膠皮的檢測，因此需要一台可以在以上條件下能夠清楚取像的相機，另外考量到因為膠皮有各式各樣的顏色，為了讓深度學習模型能夠提取出色彩上的特徵，最終本研究採用 Allied Vision 的 Mako G-131C 彩色工業用相機，如[圖 28](#_111kx3o) 所示，解析度為1280 × 1024，取像速度可達每秒 62 幀(FPS)，不包含鏡頭機身的大小為60.5 × 29 × 29毫米，重量為 80 公克，它是採用GigE Vision 的介面標準，可將影像透過乙太網路傳輸至伺服器，也能夠支援乙太網路供電 (Power over Ethernet, PoE)，並可以簡化現場設備數量的配置。



圖 28 Allied Vision Mako G-131C 工業相機[[29]](#_2y3w247)。

捲送機在捲送的膠皮寬度大約為 1.63 公尺，而工業用相機架設的高度與膠皮距離0.71 公尺，本研究針對三組焦距分別為Kowa’s LM4NCL 焦距3.5 毫米鏡頭、 Fuji’s DF6HA-1S 焦距 6 毫米鏡頭以及Kowa’s LM12NCL 焦距 12 毫米鏡頭進行實測，如下[圖 29](#_3l18frh) 所示，透過測試的結果可以發現 3.5 毫米可以拍攝的範圍比較廣，但是影像卻有嚴重的變形，此外也無法拍攝出膠皮中的細節，如[圖 30](#_206ipza) 所示，6 毫米的鏡頭，拍攝出的膠皮影像範圍相較於 3.5 毫米的鏡頭小，但是卻沒有嚴重變

形的問題，也能拍出膠皮的細節，如[圖 31](#_4k668n3) 所示，以焦距 12 毫米的鏡頭拍攝出來

的膠皮表面細節更為明顯，但是其拍攝所涵蓋的範圍卻更小，如[圖 32](#_2zbgiuw) 所示。因

此，本研究最終採用 6 毫米的鏡頭作為工業用相機的搭配，雖然拍攝細節無法與

12 毫米鏡頭一樣清楚，但是依然可拍攝出膠皮上的瑕疵，搭配 6 毫米的鏡頭，使用兩台相機就可以涵蓋全部膠皮所需的拍攝範圍，搭配各鏡頭可拍攝範圍如下[表](#_1egqt2p) [1](#_1egqt2p) 所示。



| (a) | (b) | (c) |
| --- | --- | --- |

圖 29 (a)Kowa’s LM4NCL[[30]](#_1d96cc0) (b)Fuji’s DF6HA-1S[[31]](#_3x8tuzt) (c)Kowa’s

LM12NCL[[32]](#_2ce457m)。

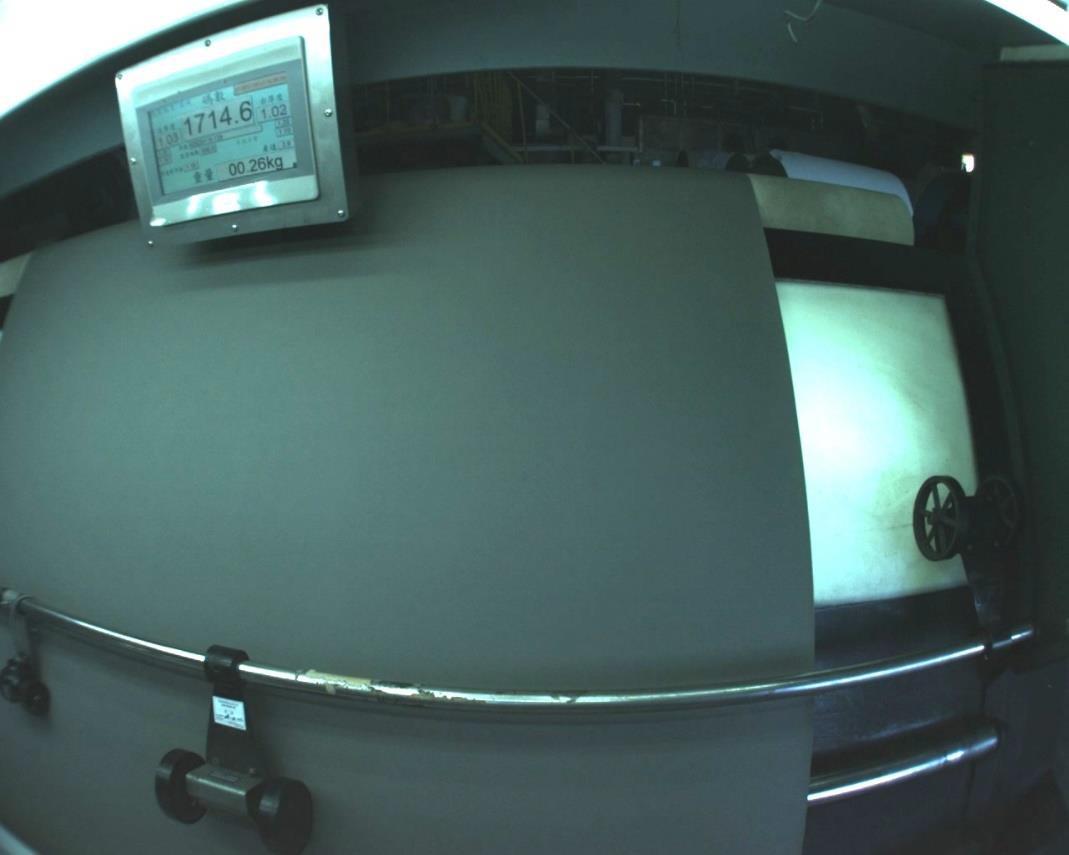


圖 30 焦距 3.5 毫米鏡頭所拍攝影像。



圖 31 焦距 6 毫米鏡頭所拍攝影像。



圖 32 焦距 12 毫米鏡頭所拍攝影像。

表 1 各鏡頭可拍攝範圍。

|  | 可視範圍(𝒎𝟐) | 涵蓋拍攝範圍所需相機台數 |
| --- | --- | --- |
| 3.5mm 鏡頭 | 1.47 × 1.18 | 2 |
| 6mm 鏡頭 | 0.86 × 0.69 | 2 |
| 12mm 鏡頭 | 0.42 × 0.34 | 4 |

下[圖 33](#_3ygebqi) 為本研究在現場實際評估過後，安裝相機於捲送機上的位置，將相機安置在捲送機上方，藉由捲送機上方的光源進行觀測物的補光，透過網路傳輸影像至室內的伺服器來進行辨識影像中是否含有瑕疵，若含有瑕疵則會透過 modbus TCP 發送警訊給可程式化邏輯控制器(PLC)來控制是否需要停止捲送機的運作，本研究偵測系統架構圖如[圖 34](#_2dlolyb) 所示。

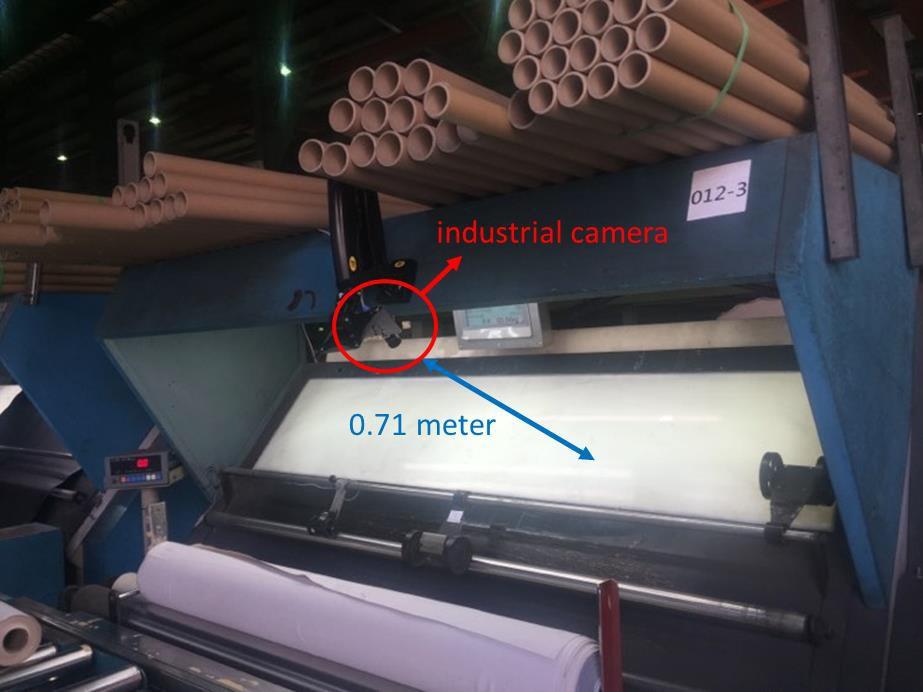


圖 33 相機建置環境示意圖。

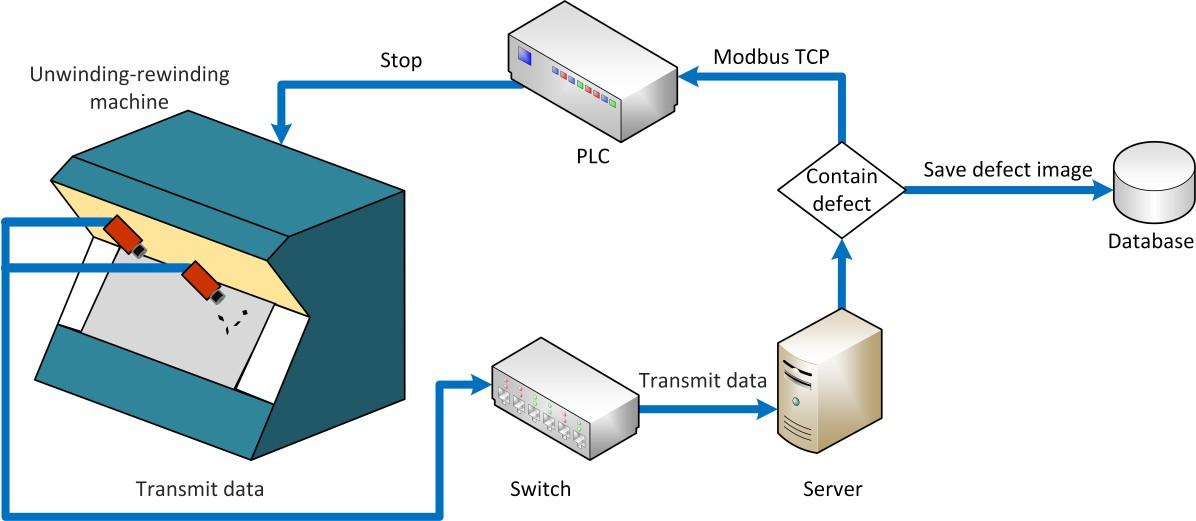


圖 34 偵測系統架構圖。

本研究用來訓練深度學習模型並進行實驗的伺服器規格如下，使用的作業系統為 Ubuntu 18.04、CPU 處理器為 Intel® Core™ i7-8700K CPU @ 3.70GHz × 12、記憶體為 32GB 以及 GPU 顯示卡 GeForce GTX 1080 Ti × 2。

## 資料集

膠皮工廠所提供的瑕疵影像，總共分成 6 種瑕疵類別，分別為變形、損傷、

髒污、色差、褶皺和壓痕，如[圖 35](#_3cqmetx) 所示，在排除模糊與重複的影像過後，可以

使用的瑕疵影像總共為130 張，分別為變形瑕疵影像21 張、損傷瑕疵影像35 張、

髒污瑕疵影像 38 張、色差瑕疵影像 19 張、褶皺瑕疵影像 5 張以及壓痕瑕疵影像

12 張，本研究將可使用的瑕疵影像百分之八十作為訓練資料，共 104 張，其餘的

百分之二十作為測試影像，共 26 張，來驗證所提方法的偵測結果，為了克服資料

量不足的問題，本研究透過資料增量的方法，使每張訓練影像能產生出 20 張可用

來訓練的影像，經過資料增量後的訓練瑕疵影像總共有 2080 張，分別為變形瑕疵

影像 340 張、損傷瑕疵影像 560 張、髒污瑕疵影像 600 張、色差瑕疵影像 300 張、

褶皺瑕疵影像 80 張以及壓痕瑕疵影像 200 張，瑕疵增量後的瑕疵影像如下[圖 36](#_1rvwp1q)

所示，而重複出現的瑕疵，是因為影像縮小後以鏡射填補空白區域所造成。

| (a) | (b) |
| --- | --- |
| (c) | (d) |
| (e) | (f) |

圖 35 各種瑕疵影像 (a)變形 (b)損傷 (c)髒污 (d)色差 (e)褶皺 (f)壓痕。



圖 36 瑕疵影像資料增量結果示意圖。

而 GoogLeNet 的部分只針對瑕疵的主體進行辨識，因此，本研究將資料增量後的訓練資料進行瑕疵主體的裁切，並把這些裁切過後的影像作為 GoogLeNet 訓練的樣本，如下[圖 37](#_4bvk7pj) 所示。

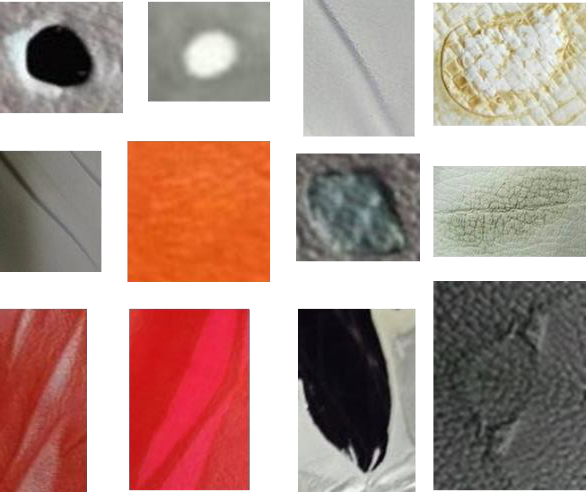


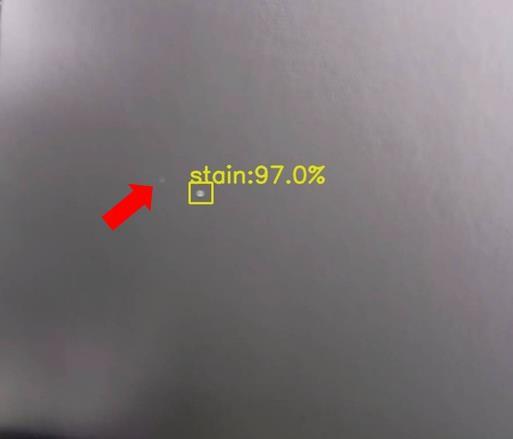
圖 37 瑕疵本體影像。

## 實驗結果

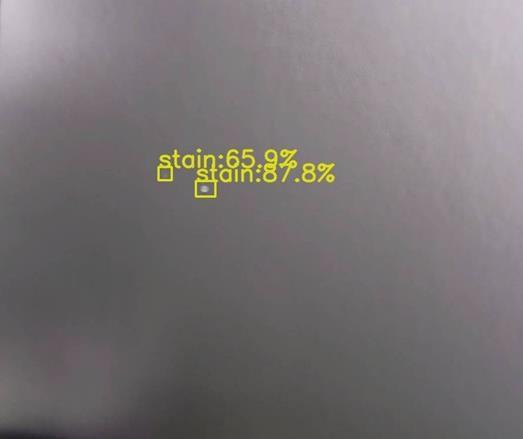
本研究模擬實際檢測環境，對廠商後續提供的樣本進行拍攝，額外取得 17 張

瑕疵影像，加上最初提供資料的百分之二十作為測試資料 26 張，總共的測試影像

為 43 張影像，使用資料增量過後的訓練資料來訓練模型，並以測試資料作為模型評估的依據，實驗結果如[圖 38](#_1664s55) 所示，可以看到優化過後的 YOLOv4 在抓取小瑕疵有更好的表現。



| (a) | (b) |
| --- | --- |
| (c) | (d) |

圖 38 (a)、(b) YOLOv4 偵測結果 (c)、(d) Improved YOLOv4 偵測結果。

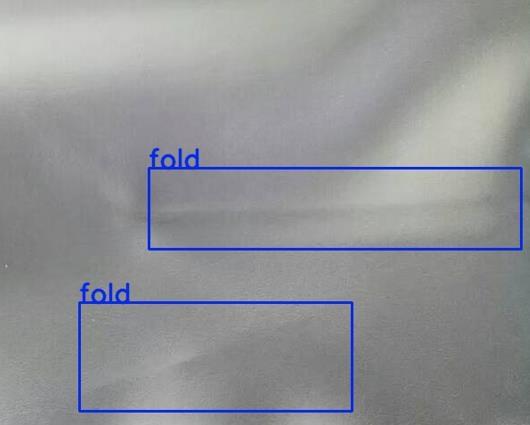
若只使用 YOLOv4 進行偵測，在分類時很容易出現誤判的情況，因此本研究在 YOLOv4 偵測到瑕疵位置後，加入了 GoogLeNet 來進行瑕疵類別的辨識，解決誤判的問題，損傷瑕疵偵測結果如[圖 39](#_3q5sasy) 所示。



| (a) | (b) |
| --- | --- |
| (c) | (d) |

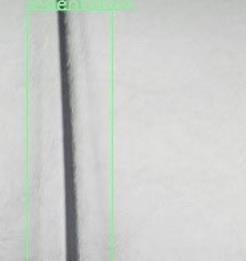
圖 39 (a)、(b) YOLOv4 誤判結果 (c)、(d) 所提方法辨識結果。

本研究所提方法在測試資料集中，分別針對 6 種瑕疵進行偵測，偵測結果如下[圖 40](#_25b2l0r)，以及後來模擬實際檢測環境的影像偵測結果，如[圖 41](#_kgcv8k)，都有良好的偵測效果，ground truth 與偵測結果各類別物件統計比較如下[圖 42](#_34g0dwd) 所示。



| (a) | (b) |
| --- | --- |
| (c) | (d) |
| (e) | (f) |

圖 40 (a)變形 (b)損傷 (c)髒污 (d)色差 (e)摺皺 (f)壓痕。

|  | C:\Users\Leonz\AppData\Local\Microsoft\Windows\INetCache\Content.Word\29-2.jpg |
| --- | --- |
| (a) | (b) |
|  |  |
| (c) | (d) |

圖 41 (a)、(b)損傷 (c)變形 (d)色差。

|  |  |
| --- | --- |
| (a) | (b) |

圖 42 (a)ground truth (b)偵測結果。

## 評估方法

本研究採用混淆矩陣(Confusion matrix)作為評估實驗結果的依據，如下[圖 43](#_43ky6rz)所示，混淆矩陣由 TP、FP、FN、TN 所組成，其中 TP 為實際是正樣本預測結果也是正樣本，FP 為實際上為負樣本但預測結果為正樣本，FN 為實際為正樣本但預測結果為負樣本，TN 為實際上是負樣本且預測結果也為負樣本。

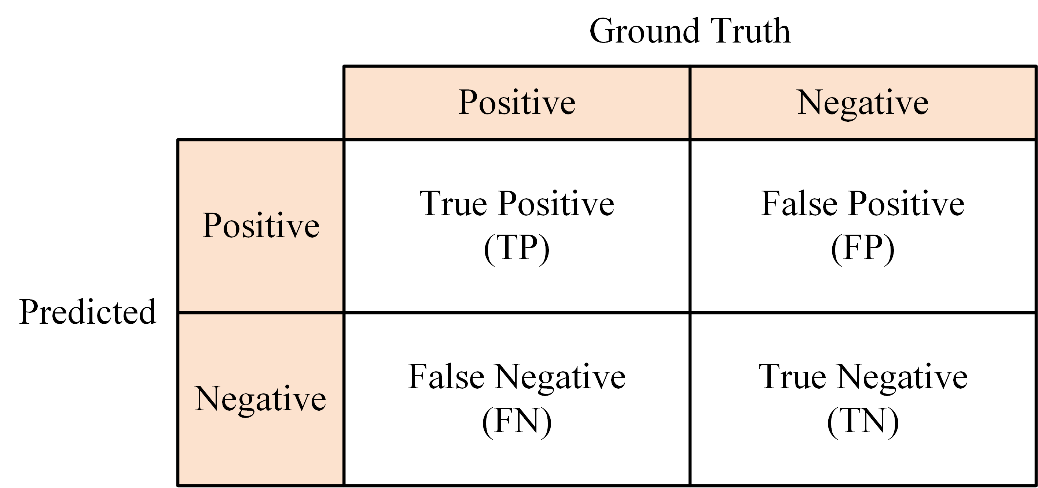


圖 43 混淆矩陣。

本實驗採用以下幾種常見的指標作為模型評估的標準，精確率(Precision)表示預測正樣本中且實際為正樣本的比例，如下式[( 3 )](#_2iq8gzs)所示，召回率(Recall)表示所有實際為正樣本中且正確預測為正樣本的比例，如下式[( 4 )](#_xvir7l)所示，以及 F1-Measure表示反映整體的指標，透過精確率和召回率來計算，如下式[( 5 )](#_3hv69ve)所示。

| 𝑇𝑃  𝑃𝑟𝑒𝑐𝑖𝑠𝑖𝑜𝑛 =  𝑇𝑃 + 𝐹𝑃 | ( 3 ) |
| --- | --- |

| 𝑇𝑃  𝑅𝑒𝑐𝑎𝑙𝑙 =  𝑇𝑃 + 𝐹𝑁 | ( 4 ) |
| --- | --- |

| 𝑃𝑟𝑒𝑐𝑖𝑠𝑖𝑜𝑛 × 𝑅𝑒𝑐𝑎𝑙𝑙  𝐹1 − 𝑀𝑒𝑎𝑠𝑢𝑟𝑒 = 2 ×  𝑃𝑟𝑒𝑐𝑖𝑠𝑖𝑜𝑛 + 𝑅𝑒𝑐𝑎𝑙𝑙 | ( 5 ) |
| --- | --- |

mAP(Mean Average Precision)是最常用來評估物件偵測的指標，而此方法的判別方式是採用 IoU(Intersection over Union)作為評估的標準。以下會說明 IoU 的判別標準和 mAP 的計算方法。

IoU 是物件偵測在判斷預測結果與ground truth 的一種方法，公式如下式[( 6 )](#_1x0gk37)，一般物件偵測在檢測IoU 常使用 0.5 作為判定的門檻值，當預測的bounding box 和 ground truth 重疊大於 0.5 以上代表預測結果為正樣本，各種 IoU 門檻值示意圖如[圖 44](#_4h042r0)。

IoU(A, B) =

𝐴 ∩ 𝐵



𝐴 ∪ 𝐵

( 6 )

| (a) | (b) | (c) |
| --- | --- | --- |

圖 44 IoU 示意圖 (a)IoU = 0.5 (b)IoU = 0.7 (c)IoU = 0.9。

AP 的計算公式如下式[( 7 )](#_2w5ecyt)所示，假設某一個類別預測出 n 個物件，首先會將偵測到的物件以信心分數高至低排序成 1~n(rank)，從信心分數最高的物件開始計算，p為𝑟𝑘在 rank k 時正樣本累積的數量。若共有 N 個類別，mAP 則是計算所有

類別的 AP(Average Precision)平均值，如下式[( 8 )](#_1baon6m)所示。

| 1 𝑛 𝑝  AP = ∑  𝑛 𝑘=1 𝑟𝑘 | ( 7 ) |
| --- | --- |

| ∑𝑁 1 𝐴𝑃𝑛  mAP = 𝑛=  𝑁 | ( 8 ) |
| --- | --- |

## 比較結果

本研究使用廠商提供瑕疵影像中的百分之二十以及後續模擬檢測環境所拍攝的影像作為測試資料，進行所提方法的評估驗證，測試資料分別為變形瑕疵影像 6 張、損傷瑕疵影像 17 張、髒污瑕疵影像 9 張、色差瑕疵影像 7 張、褶皺瑕疵影

像 2 張以及壓痕瑕疵影像 2 張。

本研究所提的方法在沒有使用資料增量的情況下，模型針對模擬檢測環境的影像偵測結果並不理想，其原因為訓練資料的瑕疵大小與實際要偵測的瑕疵大小差異很大，再加上訓練資料與測試資料的照明也有所不同，導致在偵測時，有許多瑕疵並沒有被正確的檢測出來。面對訓練與測試資料的差異，藉由使用資料增量後的訓練資料來訓練模型，模型在偵測表現上各個指標皆有明顯的提升，如[表 2](#_2afmg28) 所示。

表 2 資料增量對各個指標的影響比較。

| **方法** | **Precision** | **Recall** | **F1-Measure** | **mAP** |
| --- | --- | --- | --- | --- |
| **沒有使用資料增量** | 80.00% | 48.12% | 60.07% | 63.01% |
| **使用資料增量** | **92.42%** | **91.73%** | **92.07%** | **81.21%** |

[表 3](#_pkwqa1) 為 YOLOv4、優化過的 YOLOv4 與所提方法的瑕疵偵測比較結果，可以看出優化過的 YOLOv4 召回率比 YOLOv4 有明顯的提升，是因為在小瑕疵的漏抓上有所改善。由於只使用 YOLOv4 來進行偵測會容易出現分類錯誤的情形，因此所提方法加入 GoogLeNet 來對 YOLOv4 偵測到的瑕疵進行辨識，在各個指標上皆有所提升。

表 3 各深度學習模型於瑕疵偵測辨識的評估比較。

| **方法** | **Precision** | **Recall** | **F1-Measure** | **mAP** |
| --- | --- | --- | --- | --- |
| **YOLOv4** | 88.62% | 81.95% | 85.15% | 71.53% |
| **Improved YOLOv4** | 88.64% | 87.97% | 88.30% | 77.85% |
| **Proposed Method** | **92.42%** | **91.73%** | **92.07%** | **81.21%** |

# 結論與未來發展

本研究提出了一個機器視覺搭配串級深度學習網路即時瑕疵偵測的方法來檢測 PVC 皮革，將彩色工業用相機架設於膠皮捲送機上，並透過網路傳輸至室內的伺服器來檢測影像中是否含有瑕疵，將偵測的結果傳送給可程式化邏輯控制器進行後續的處理，加強控管製品的品質。由於只使用單一個深度學習模型容易會有分類錯誤的情況，為了解決這個問題，本研究採用串級的深度學習網路，其中，第一個深度學習模型是用來偵測瑕疵的位置，而第二個深度學習模型則是對偵測到的瑕疵加以辨識，藉由此方式提高辨識的準確率。

由於廠商提供的資料非常有限，而且拍攝的資料形態與實際工業用相機抓取到的影像差異非常大，因此本研究只用了資料增量，來模擬生成類似工業用相機視野拍攝出來的瑕疵影像來對模型進行訓練。由於得到的原始資料非常稀少，再加上瑕疵的樣貌繁多，且資料中的瑕疵種類分布也不平均，很有可能會導致一些不同樣貌的瑕疵無法被偵測出來，所以在未來的研究中，可以嘗試使用更多工業用相機所拍攝出來的真實瑕疵資料加入模型的訓練，讓偵測的表現可以有所提升並對不同樣貌的瑕疵擁有更高的適應性。

# 參考文獻

1. H. Liu, W. Zhou, Q. Kuang, L. Cao and B. Gao, "Defect Detection of IC Wafer Based on Spectral Subtraction," IEEE Transactions on Semiconductor Manufacturing, vol. 23, no. 1, pp. 141-147, Feb. 2010, doi: 10.1109/TSM.2009.2039185.
2. B. Tian, S. Xu and Z. Li, "Glassware crack defects detection based on wavelet transform," Chinese Automation Congress (CAC), pp. 4954-4958, 2017, doi: 10.1109/CAC.2017.8243657.
3. A. Kumar and G. Pang, "Defect detection in textured materials using Gabor filters," IEEE Industry Applications Conference, vol.2, pp. 1041-1047, 2000, doi: 10.1109/IAS.2000.881960.
4. M. Fathu Nisha, P.Vasuki, and S. Mohamed Monsoor Roomi, “Survey on various defect detection and classification methods in fabric images,” J.Environ.Nanotechnol., Vol.6 No.2,pp.20-29, 2017, doi:10.13074/jent.2017.06.172255
5. J. E. See, "Visual inspection reliability for precision manufactured parts," Human Factors, vol. 57, no. 8, pp. 1427-1442, Dec, 2015, doi: 10.1177/0018720815602389.
6. C. S. Cho, B. M. Chung and M. J. Park, "Development of real-time vision-based fabric inspection system," IEEE Transactions on Industrial Electronics, vol. 52, no. 4, pp. 1073-1079, Aug, 2005, doi: 10.1109/TIE.2005.851648.
7. Q. Zhang, X. Chang and S. B. Bian, "Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN," IEEE Access, vol. 8, pp. 6997-7004, 2020, doi: 10.1109/ACCESS.2020.2964055.
8. A. Bochkovskiy, C. Wang and H. M. Liao, "YOLOv4: Optimal Speed and Accuracy

of Object Detection," arXiv preprint arXiv:2004.10934, 2020.

1. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015, pp. 1-9, doi: 10.1109/CVPR.2015.7298594.
2. R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580-587, doi: 10.1109/CVPR.2014.81.
3. R. Girshick, "Fast R-CNN," IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.
4. S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017, doi: 10.1109/TPAMI.2016.2577031.
5. T. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal Loss for Dense Object Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 2, pp. 318-327, 1 Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.
6. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. E. Reed. "SSD: single shot multibox detector," arXiv preprint arXiv:1512.02325, 2015.
7. A. Bochkovskiy, "yolov4.weights", [Online], Avaliable : https://github.com/AlexeyAB/darknet/releases/download/darknet\_yolo\_v3\_optimal

/yolov4.weights.

1. A. F. Agarap, "Deep learning using rectified linear units (relu)," arXiv preprint arXiv:1803.08375, 2018.
2. B. Xu, N. Wang, T. Chen and M. Li, "Empirical evaluation of rectified activations

in convolutional network," arXiv preprint arXiv:1505.00853, 2015.

1. D. Misra, "Mish: A self regularized non-monotonic neural activation function," arXiv preprint arXiv:1908.08681, 2019.
2. S. Ioffe, C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”, arXiv:1502.03167, 2015.
3. J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018
4. C. Wang, H. Mark Liao, Y. Wu, P. Chen, J. Hsieh and I. Yeh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020, pp. 1571- 1580, doi: 10.1109/CVPRW50498.2020.00203.
5. K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.
6. F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1800-1807, doi: 10.1109/CVPR.2017.195.
7. T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature Pyramid Networks for Object Detection," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936-944, doi: 10.1109/CVPR.2017.106.
8. S. Liu, L. Qi, H. Qin, J. Shi and J. Jia, "Path Aggregation Network for Instance Segmentation," IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 8759-8768, doi: 10.1109/CVPR.2018.00913.
9. K. Simonyan and A. Zisserman. "Very deep convolutional networks for large-scale

image recognition," arXiv preprint arXiv:1409.1556, 2014.

1. A. Krizhevsky, I. Sutskever, and G. E. Hinton. "Imagenet classification with deep convolutional neural networks," Advances in Neural Information Processing Systems, pp. 1097–1105, 2012.
2. M. Lin, Q. Chen and S. Yan, "Network in network," arXiv preprint arXiv:1312.4400, 2013.
3. SPECTRAtech S.A. Allied-Vision Mako G-131C PoE, [Online], Avaliable : https:/[/www.spe](http://www.spectratech.gr/en/product/41051/Allied-Vision_Mako_G-131C_PoE)c[tratech.gr/en/product/41051/Allied-Vision\_Mako\_G-131C\_PoE](http://www.spectratech.gr/en/product/41051/Allied-Vision_Mako_G-131C_PoE)
4. Kowa LM4NCL Lens, [Online], Avaliable : https://lenses.kowa-usa.com/118-ncl- series/488-lm4ncl.html
5. FujiFilm F.A. DF6HA-1S Lens, [Online], Avaliable : https:/[/www.fujifilm.com/us/](http://www.fujifilm.com/us/en/business/optical-devices/machine-vision-lens/hf-)e[n/business/optical-devices/machine-vision-lens/hf-](http://www.fujifilm.com/us/en/business/optical-devices/machine-vision-lens/hf-) ha-1s-series#model01
6. Kowa LM12NCL Lens, [Online], Avaliable : https://lenses.kowa-usa.com/118-ncl- series/491-lm12ncl.html