國立臺灣大學工學院機械工程學系

碩士論文

Department of Mechanical Engineering

College of Engineering

National Taiwan University

Master Thesis

領域自適應應用於自動化光學檢測

Applications of Domain Adaptation in Automatic Optical Inspection

李浩平

Hao-Ping Lee

指導教授：李貫銘 博士

Advisor: Kuan-Ming Li Ph.D.

中華民國 111 年 8 月

August 2022

doi:10.6342/NTU202203562

口試委員會審定書

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

（論文中文題目）(Chinese title of Master's thesis)

領域自適應應用於自動化光學檢測

（論文英文題目）(English title of Master's thesis)

Applications of Domain Adaptation in Automatic Optical Inspection

本論文係__李浩平___(姓名)__R09522709__（學號）在國立臺灣大學機械工程所 (系/所/學位學程)完成之碩士學位論文，於民國_111_年_08_月_17_日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Department / Institute of __Mechanical Engineering__ on __17__(date)__08___ (month)__2022__ (year) have examined a Master's thesis entitled above presented by __Hao-Ping LEE___ (name)_____R09522709_____ (student ID) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

李貫銘 教授	李貫銘 (簽名)	(指導教授)
楊宏智 教授	楊宏智 (簽名)	
許志青 董事長	許志青 (簽名)	
邱弘興 副院長	邱弘興 (簽名)	

系主任 林沛群 (簽名)

doi:10.6342/NTU202203562

誌謝

時光飛逝，兩年的碩士生活已到了尾聲，期間經歷了許多挑戰，也面臨了許多困境，在無數人的幫助下，我的知識更加豐富，閱歷更加廣袤，心態也更為沉穩，感謝出現在我生命的貴人們，成就了我多彩多姿的碩士生活，也成就了這篇論文的誕生。

感謝楊宏智教授與李貫銘教授，讓我加入 138 實驗室的大家庭，您們共創了實驗室良好的學習環境，各式各樣的研究主題讓我在與其他研究生交流的過程中能拓展不同領域的知識，您們也在兩年間分享了許多專業與人生路上的經驗，專業的見解總能啟發我對更深知識的探尋，生活的智慧更將使我一輩子受用無窮。

感謝智泰科技的許志青董事長，給予我在智泰科技實習的機會，在智泰的兩年裡，我學到了許多 AOI 相關的知識與產業的思維，感謝軟體研發部的 Gordon、Hank 和 Paul，總能分享最新的深度學習資訊與應用方向，也能給予我程式撰寫上的幫助，感謝光學部的 Kevin，協助我各種研究設備的搭建，沒有智泰科技提供的實驗設備和研發能量，就沒有這篇論文的誕生。

感謝父母的悉心栽培，使我在學習的道路上衣食無憂。

感謝實驗室的總管鍾姐，處理大家行政上的大小事務。感謝實驗室的學弟們，在最煎熬的碩二期間，維持實驗室的衛生與歡笑。特別感謝唐大為學長，工作、研究都受到你的幫助，若我的碩士生活沒有你的指引，恐怕會時常手足無措。最後，感謝相互砥礪的俊延、泓諭、鵬育、柏瀚、達仁、冠良、柏秀，不只在學習的道路中給予我幫助，也在學習之餘給予我正能量，能在 138 和你們一同成長是我的幸運。

畢業是一時，但學習是一世，感謝人生路上曾有你們相伴。

doi:10.6342/NTU202203562

摘要

隨著卷積神經網路在圖像檢測領域的快速發展，近年來學界與業界致力於將卷積神經網路技術導入到自動化光學檢測的系統內，進行工業瑕疵檢測，邁向產線智慧化。然而，對於新的工件，即便已有過往之相似樣品的檢測模型，也會因樣品本質上的差異或取像光學架構的差異，而無法直接使用，檢測效果極差，故往往只能選擇對新資料重新標記訓練，而訓練檢測模型通常需要大量的瑕疵樣本標籤資料，其獲取方式往往仰賴手動標記，極耗費時間與人力。

有鑑於此，本研究使用領域自適應的技術，在未對待測之新資料進行標籤的情況下建立瑕疵分類模型。其原理為利用事先已標記且與待測目標資料相似的資源資料、待檢測的無標記目標資料進行模型訓練，讓神經網路最終可以成功檢測目標資料。本研究以實際產線的不同木種之木皮瑕疵影像、不同色系之布匹瑕疵影像與公開之金屬表面瑕疵資料集為例，比較領域自適應神經網路在不同工業應用的檢測表現，並分析網路擷取之特徵進行模型優化，根據結果統整出領域自適應技術應用於工業瑕疵檢測的合適流程。

經過本研究的實驗測試，將 ResNet50 作為特徵擷取器訓練領域自適應模型 DANN (Domain-Adversarial Neural Network)，並輔以熵調整 (Entropy Conditioning)，能有效分類無標籤之瑕疵影像。相較於直接使用舊有相似資料之模型進行辨識，對於木皮類之瑕疵影像，分類準確率能從 52.96%提升至 84.93%；對於布匹類之瑕疵影像，分類準確率能從 22.58%提升至 73.75%；對於金屬表面之瑕疵影像，分類準確率能從 31.13%提升至 95.58%。另外，若對特徵擷取的特徵層進行優化選擇，分類準確率能再有所提升，木皮類提升至 90.86%，布匹類提升至 75.68%，金屬表面瑕疵類提升至 96.22%。通過此流程能快速訓練出對無標籤新資料的辨識模型，有效節省人力與時間成本。

關鍵字：智慧製造、人工智慧、深度學習、領域自適應、瑕疵檢測

doi:10.6342/NTU202203562

Abstract

With the development of deep learning, convolutional neural network has achieved outstanding performance in the field of image detection. In recent years, the academia and industry have been committed to introducing convolutional neural network technology into the production line of automatic optical inspection. However, due to the differences of data characteristics and image acquisition methods, it is ineffective to recognize defects on new target data with a former model trained by similar data. Thus, engineers usually manually label the new target data and train a new defect recognition model, which brings a lot of labor costs.

In view of the above, in this research, the defect recognition model is built by using domain adaptation, which is able to train a neural network on a labeled similar source dataset and secure a good accuracy on the unlabeled new target dataset. Wood and textile defect images gathered from actual production lines and an open dataset of metal surface defect images, NEU-CLS, are used as the verification data. Different kinds of domain adaptation models are compared. Also, feature extraction layers are analyzed to optimize the model. Finally, a general process to train a defect recognition model using domain adaptation is organized.

According to the results, a classification model trained by the DANN (Domain-Adversarial Neural Network) domain adaptation method with a ResNet50 backbone and entropy conditioning algorithms is effective to recognize unlabeled defect images. In addition, the accuracy is further increased by choosing proper feature extraction layers. For the wood defect dataset, the accuracy increases to 90.86%. For the textile defect dataset, the accuracy increases to 75.68%. For the metal surface defect dataset, the accuracy increases to 96.22%. By following this process, an effective defect recognition model can be built without labeling new data. In other words, time and labor costs on

doi:10.6342/NTU202203562

labeling new data can be significantly reduced.

Keywords: Smart Manufacturing, Artificial Intelligence, Deep Learning, Domain Adaptation, Defect Inspection

doi:10.6342/NTU202203562

目錄

doi:10.6342/NTU202203562

doi:10.6342/NTU202203562

圖目錄

doi:10.6342/NTU202203562

doi:10.6342/NTU202203562

doi:10.6342/NTU202203562

表目錄

doi:10.6342/NTU202203562

第一章　緒論

1.1　研究背景

隨著工業技術與日俱進，業界對產品的生產品質需求也有所提升，若有過量瑕疵品流入下游市場，影響的將是公司的信譽問題。過去的品質管理多仰賴人工檢測，然而人工檢測本身品質也受工人的身心因素而不穩定，加上日益上升的工資與勞工意識，使得人工檢測的成本過大。為了解決此一問題，自動化光學檢測 (Automated Optical Inspection, AOI) 應運而生，傳統的 AOI 技術利用閾值法、圖像擬合、瑕疵分割等影像處理演算法，在工業瑕疵檢測上取得了巨大的成效 [1][2]，不僅能應用於大量製造的自動化產線，工人也成為了僅需監督檢測系統正常運作或最終檢測的角色，使得人力成本能控制在一定範圍。如圖 1-1 所示，根據美國著名分析機構 Markets and Markets 的分析 [3]，世界 AOI 市場規模將持續上升，預估在 2025 年可以達到 15.83 億美元。然而傳統 AOI 技術基於演算法的特性而仍有許多侷限，對於影像背景單純、瑕疵種類單一的產品檢測才較易維持良好的檢出率，反之則容易受到影響而無從判斷，進而導致 AOI 技術在特定產業的導入有所阻力，成為極待解決的問題。

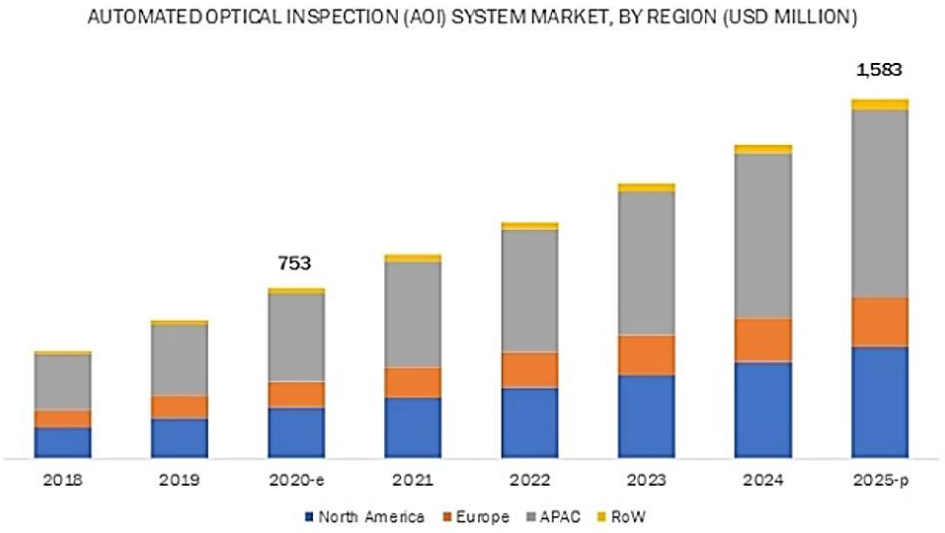


圖 1-1 AOI 市場規模發展 [3]

doi:10.6342/NTU202203562

近年伴隨圖形處理器 (Graphics Processing Unit, GPU) 等電腦硬體的效能提升，人工智慧 (Artificial Intelligence, AI) 中深度學習 (Deep Learning, DL) 的運算速度大幅提升，在實務應用上成為可能，其中於影像辨識領域也有不少進步，成為了AOI 領域問題可能的突破口。

深度學習的基礎為神經網路 (Neural Network, NN)，其模仿人體大腦的神經元系統，以層層堆疊的形式所構成，在影像辨識上學者更提出卷積神經網路 (Convolutional Neural Networks, CNN) [4]，透過卷積層、池化層以及全連接層的搭配，計算影像特徵，相較於傳統影像處理演算法，其對一定程度的特徵誤差或背景隨機性有較大的適應能力，應用在瑕疵檢測上即是能不受同種瑕疵間的些微變化影響或過濾某些背景變異 [5]，結合已成熟的傳統 AOI 技術，能相互取長補短，形成更完善的瑕疵檢測系統，為此國內外 AOI 廠商皆有提出深度學習瑕疵檢測解決方案。然而每次訓練新的卷積神經網路都需要先以人工對瑕疵樣本進行標記，如何節省此時間與人力成本成為重要議題。

木材自古即為人類生存的重要資源，除了建築生火，現今也有提煉其纖維化為服飾或是食品的新應用出現，其中經加工裁切後的木皮擁有木頭獨特的紋理，可以製成地板等既美觀又實用的產品。由於木材是自然資源，受氣候、生態影響甚大，對於木皮製造廠商來說，確保木皮的品質便成為了重要課題，例如一些裂痕或蟲的蛀洞，不僅不美觀，對於結構的堅固程度也有影響，木皮廠會需要將該部分裁切掉。因傳統的影像辨識方法無法在不規則的木紋之上進行特徵判斷，現今大多木皮廠仍仰賴人工檢測瑕疵，但也已有廠商導入深度學習瑕疵檢測系統，成功對特定木種之木皮進行瑕疵檢測。

織物的歷史可追朔至石器時代晚期，當時便有植物纖維編織而成的衣物，後續又有中國的絲綢、波斯的地毯等有名織物出現在歷史的洪河之中，人類依據材料的特性與編織手段的不同能製造出千變萬化的紡織品，時至今日，紡織品依然充斥在人類的生活之中，作為清潔工具、衣物、裝飾等用途。也因為紡織品的多樣性與快

doi:10.6342/NTU202203562

速的更替性，布匹廠商大多在產線上仍維持人工檢測瑕疵，若隨布料的推陳出新去更替 AOI 模型，其成本將更為巨大。近年由於消費者對布料品質的要求逐漸提高，布匹廠商希望朝著智慧化、自動化的產線發展，開始嘗試導入深度學習瑕疵檢測系統，因其對背景特徵隨機性的耐受力較 AOI 模型佳的特性，成功取得了一定的成效。

1.2 研究動機與目的

現行的木皮與布匹廠商在生產線中對於瑕疵檢測仍多仰賴人工檢測，且檢測師傅多根據經驗判斷瑕疵所屬類別，沒有明確判斷標準與穩定性，且難以將經驗傳承。為了打造自動化、智慧化的檢測系統以節省人力成本，許多廠商嘗試以深度學習建立瑕疵檢測模型，並已取得一定成效。然而當新產線有需求時，因為新產線與舊產線的樣品差異過大或光學環境差異，容易導致深度學習檢測模型失效。例如舊產線已訓練好之白橡木瑕疵檢測模型，在舊產線上能良好的辨識白橡木上之破裂與木珠瑕疵；然而要對新產線的胡桃木進行瑕疵檢測時，若直接使用舊產線之白橡木檢測模型，容易對胡桃木之破裂與木珠瑕疵產生誤判，因此廠商多選擇對新產線之胡桃木樣本重新訓練專門檢測胡桃木瑕疵的深度學習檢測模型。為此，工程師將重新對胡桃木瑕疵樣本進行標記並訓練模型。然而標記過程繁瑣且耗時，若每次拓產產線都要重新進行此流程，對於木皮或布匹廠商也是一可觀之人力成本消耗。

隨著學界近年深度學習在影像辨識領域的高速發展，除了基本的辨識任務，多種分支任務被相繼提出討論，其中對於不同資料集間模型的相互使用，大多被歸類為遷移學習領域 (Transfer Learning)。在遷移學習中待測物尚無標籤的情況，則稱為領域自適應 (Domain adaptation, DA)，透過將一已有標籤的資料集與一未有標籤的資料集一同映射至某一特徵空間，且使兩者於空間中之距離相近，再對已有標籤的資料集進行分類辨識訓練，所得到的辨識模型便能也具備辨識未有標籤的資料集的功能。

doi:10.6342/NTU202203562

針對目前在木紋及布匹製程 AOI 技術瓶頸，本研究擬利用領域自適應技術免去對新資料標記的需求，解決工業界目前標記成本過高的問題。本研究的主要目的為驗證領域自適應技術於工業自動化光學檢測的可行性，以產線上取得之木皮與布匹瑕疵影像資料，以及公開金屬表面瑕疵資料集為例，對於白橡木和胡桃木、深色布(黑布)和淺色布(白布)、不同視野範圍之金屬表面瑕疵影像之間，在新產線產品沒有標籤資料的情況下，訓練出瑕疵分類模型，並比較不同領域自適應模型的瑕疵辨識效果，提出適合 AOI 領域的領域自適應技術使用方法。

1.3 研究方法

本研究之流程圖如圖 1-2 所示，本研究將先收集木皮、布匹、公開金屬表面影像並整理成資料集，再以各種領域自適應方法訓練瑕疵分類模型，也以傳統 CNN 模型訓練以利相互比較，並分析各模型判斷所依據之影像特徵，以及不同適應特徵層對模型準確率的影響，最終提出適合瑕疵檢測使用的領域自適應模型訓練流程。細部內容將於第三章詳述。

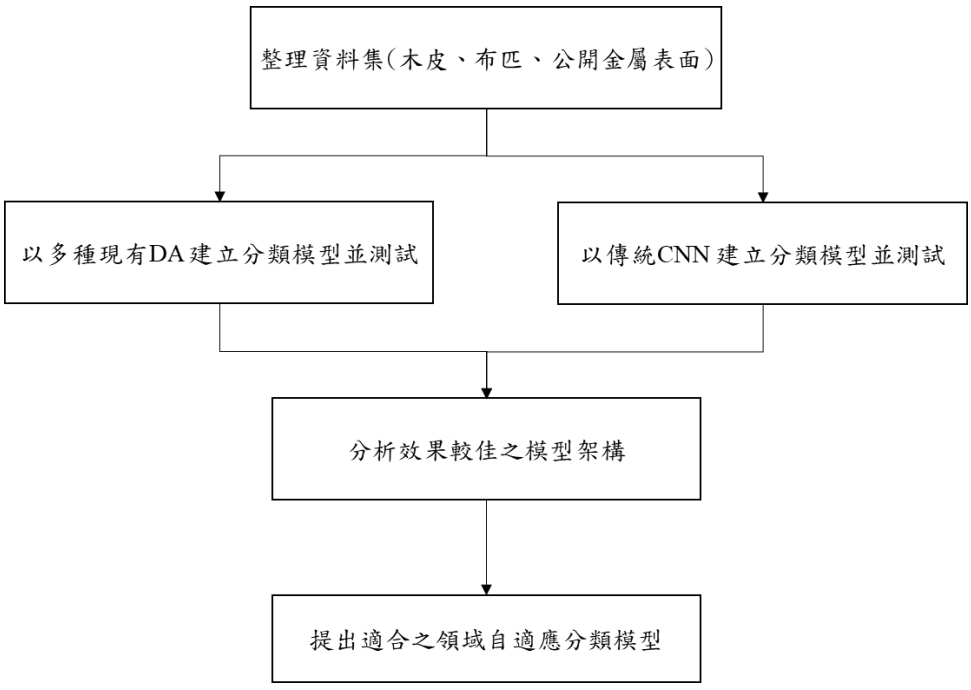


圖 1-2 研究流程圖

doi:10.6342/NTU202203562

1.4 論文架構

本論文共分五個章節，其各章內容概要如下：

第一章　緒論

說明本研究的背景、動機、目的，並簡述本研究的執行方法與整體架構。

第二章　文獻回顧

回顧學界提出的包含領域自適應在內的各式深度學習用於影像辨識的神經網路模型。

第三章　實驗流程與方法

詳述本研究的資料集取得方式與模型建立方法，並介紹研究中會使用的資料可視化方法與辨識準確率計算方式。

第四章　實驗結果與討論

整理各模型的瑕疵辨識效果，分析相對應的重要特徵，並探討改變適應特徵層對瑕疵辨識準確率的影響。

第五章　結論與未來展望

歸納整理本研究的結果，並提出未來可精進拓展的研究方向。

doi:10.6342/NTU202203562

第二章　文獻回顧

2.1　深度學習於影像辨識

近年隨著 GPU 算力的不斷提升，深度學習技術得以飛速發展，除了大規模數據的統計分析，也有學者提出卷積神經網路 (CNN) 針對於影像辨識，其利用卷積層、池化層、全連接層的複雜搭配，學習圖像特徵，在得以辨識目標物的同時又能承受一定程度的背景變異或雜訊，繼而成為了處理工業瑕疵檢測任務會考慮使用的方法之一。在本節中，將介紹深度學習技術在影像辨識領域的發展與應用。

深度學習仿照人體大腦的神經元系統，組成人工神經網路 (圖 2-1)，透過模型不斷計算與更新神經元權重，直至模型的輸出結果與目標結果相符，達成辨別的目的。其中卷積神經網路被廣泛用於電腦視覺領域之中，卷積神經網路的最大特點為卷積層，當多維資料傳遞至卷積層時，經過卷積核 (Kernel) 做卷積計算 (圖 2-2)，便能整合相鄰資料的相似度數值並映射到新的維度，適合對多維資料進行特徵分析，由於彩色圖像能視作 RGB 三層矩陣的多維資料，故卷積神經網路便適合對圖像進行分析辨識。

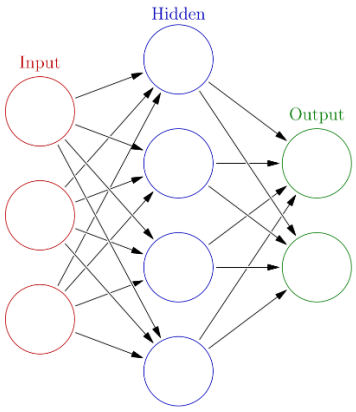


圖 2-1 人工神經網路 [6]

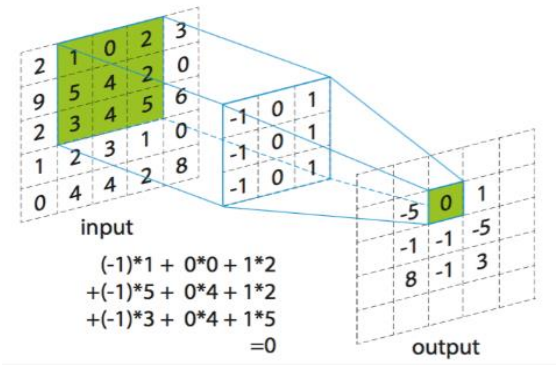


圖 2-2 卷積計算 [7]

在 1989 年到 1998 年間，Yann LeCun 發表許多卷積神經網路的相關研究，其

doi:10.6342/NTU202203562

也因此被稱為卷積網路之父，尤其 LeCun 於 1998 年發表的 LeNet 網路 [4]，其架構與現今的 CNN 已非常相似，使用了多個卷積層與池化層建構網路 (圖 2-3)，也提到了多個重要的關鍵概念，包含使用 Sigmoid 函數作為激勵函數 (Activation Function)、使用 MSE (Mean Squared Error) 作為損失函數，更利用反向傳播 (Backpropagation) 計算模型損失函數，將誤差值往回傳，讓權重在更新時能參考此誤差值做出調整，是現今大部分深度學習網路訓練時重要的概念之一。並將 LeNet 用於辨識手寫數字影像，辨識效果在當時領先其他技術。

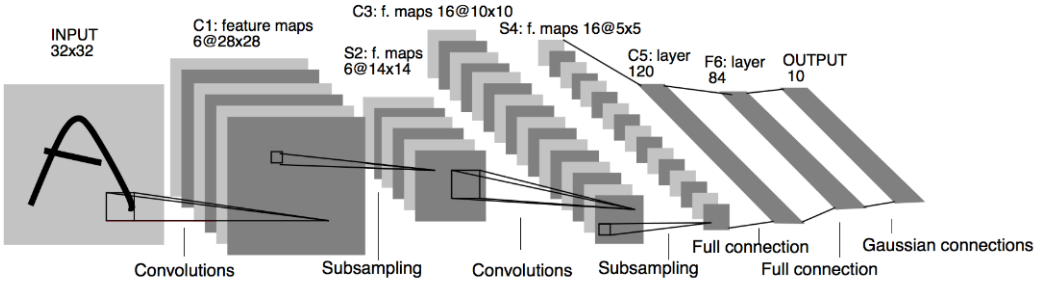


圖 2-3 LeNet 網路架構 [4]

雖然 LeNet 引起世人對卷積神經網路的注意，但相較於當時流行的 SVM (Support Vector Machine)，並沒有顯著的優勢，也仍面臨許多難題，尤其是當時硬體計算能力不足，導致大量的訓練時間花費，因此並未受特別重視。

2006 年，反向傳播法的發明人之一 Hinton 發表基於 RBM (Restricted Boltzmann machine) 的模型 [8]，其先透過預訓練 (Pre-training) 以無監督學習 (Unsupervised Learning) 調整網路權值的初始值，使網路先學習目標的大略特徵；再以監督式學習 (Supervised Learning) 對網路進行微調 (Fine-tuning)，使網路學習更細微的目標特徵，進而成功用於辨識資料。此方法除了能節省網路的訓練時間，還能有效改善使用反向傳播時梯度消失 (Gradient Vanishing) 的問題。梯度消失指的是因為梯度隨網路層數遞增而指數性遞減，最終趨於零，致使無法更新神經元權值，模型過早收斂。改善後便能增加網路層數以執行更複雜的任務，讓多層神經網路能夠真正被實踐。

doi:10.6342/NTU202203562

2012 年 Krizhevsky 提出經典模型 AlexNet [9]，其一大貢獻是採用 ReLU 函數作為激勵函數，ReLU 函數為一非線性函數 (圖 2-4)，其能更進一步解決梯度消失問題，並能有比使用 Sigmoid 函數做反向傳播時更快的收斂速度，提升網路訓練效率與複雜度。此外，AlexNet 在網路中使用了 Dropout 的正則化 (Regularization) 方法，透過在訓練的每次迭代 (Epoch) 中隨機丟棄一定比例的神經元，可有效減少網路發生過擬合 (Over Fitting) 現象；更以最大池化 (Max Pooling) 取代池化層，能減少目標特徵遺失的可能性。AlexNet 除了技術上的突破，硬體上也使用 GPU 執行平行運算對大量影像資料進行訓練，其訓練效率也較使用 CPU 高不少，首次登場便在影像分類領域的 ImageNet ILSVRC 大賽中取得冠軍，使深度學習在影像辨識領域的潛力受到研究者重視，正式開啟了深度學習技術飛速發展的時代。

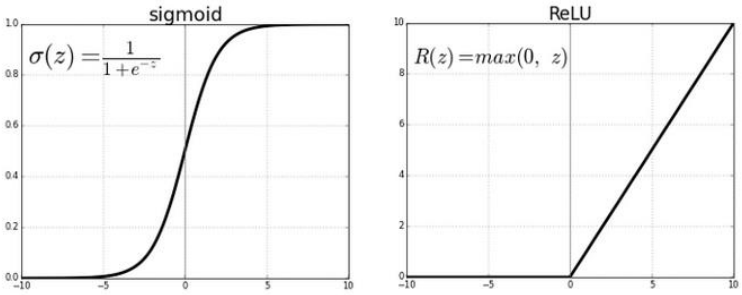


圖 2-4 Sigmoid 與 ReLU 比較 [10]

現今深度學習於影像辨識的應用主要分為三大類，以其輸出形式分為影像分類 (Image Classification)、物件偵測 (Object Detection)、影像分割 (Image Segmentation)，如圖 2-5 所示。以下將對三者做簡單介紹：

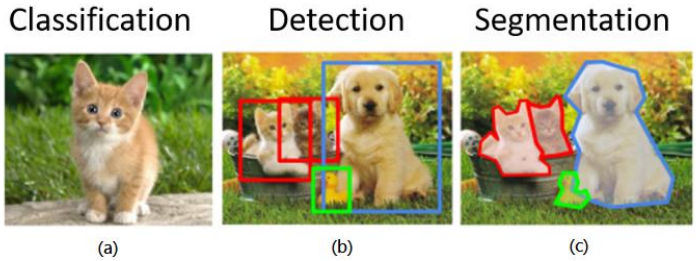


圖 2-5 影像分類、物件分割、影像分割示意圖 [11]

doi:10.6342/NTU202203562

1. 影像分類 (Image Classification)

影像分類通常用於判斷一輸入影像的所屬類別，如圖 2-5 (a)，當輸入該張影像時，一訓練良好的影像分類模型應要能輸出「貓」的類別。影像分類網路只能對一張圖片輸出一個判斷，當影像中有多種目標時，其只會輸出可能性最大的類別。經典模型如前述的 LeNet、AlexNet，而影像分類網路的發展主要和影像辨識大賽 ImageNet ILSVRC 息息相關，該競賽使用圖片數量龐大的 ImageNet 資料集做為辨識目標，自 2010 年開始至 2017 年逐年舉辦，許多冠亞軍架構都成為了後世影像分類領域的重要基礎。

2014 年 Simonyan 等人發表 VGGNet [12]，贏得了大賽分類組的亞軍，其在 AlexNet 的基礎上加深網路，拓展到了 16 層 (VGG16) 甚至 19 層 (VGG19)，驗證了增加網路的層數能有助於提升辨識性能，並改用堆疊較多較小的卷積核來取代大卷積核，藉此減少神經元參數數量。以 VGG16 為例，其架構如圖 2-6 所示，將 AlexNet 使用的 7 x 7 大小卷積核視為 3 個 3 x 3 大小卷積核的疊加，5 x 5 大小卷積核視為 2 個 3 x 3 大小卷積核的疊加，使每個卷積核全部為 3 x 3 大小，在能成功提取更細微特徵的同時減少運算量，得到更好的辨識效果。

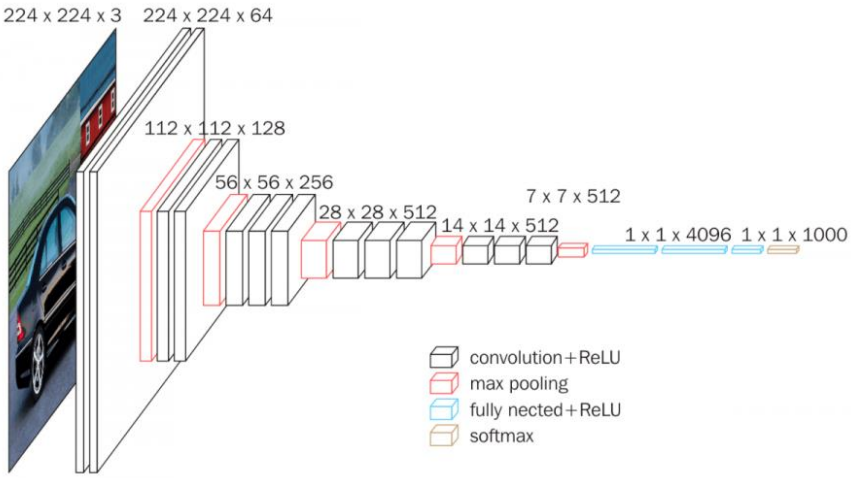


圖 2-6 VGG16 網路架構

doi:10.6342/NTU202203562

同年度 2014 年的大賽分類組冠軍則為 Szegedy 等人發表的 GoogLeNet [13]，又稱為 InceptionV1，其最大特點是證明了要增加網路效能不一定是要順序式的加深網路，也可以用非順序式的方法優化網路，其創造 Inception Module 結構 (圖 2-7)，透過並聯多個不同大小的卷積分支，形成更寬大的網路，成功在控制計算量的同時獲得良好的效能。2015 年 Ioffe 等人又進一步提出 InceptionV2 [14]，除了參考 VGGNet 以小卷積核取代大卷積核，還提出了 Batch Normalization (圖 2-8)，計算特徵對應的均值和標準差來做資料正規化，不僅降低了 Internal Covariate Shift 的問題，同時也加快收斂速度。後續團隊又提出 InceptionV3、InceptionV4，借用 ResNet [15]的結構得到更快更準確的辨識效果。

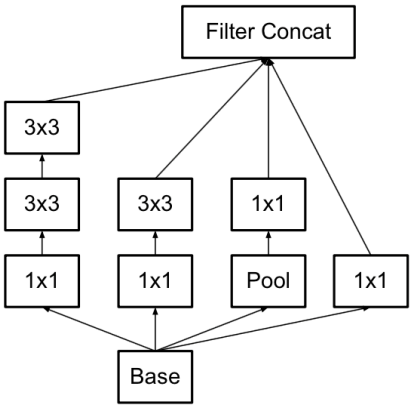


圖 2-7 Inception Module

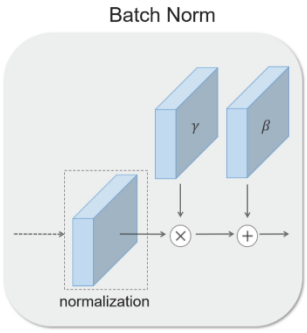


圖 2-8 Batch Normalization

而後 2015 年的大賽冠軍由 He 等人發表的 ResNet [15] 拿下。當網路深度過深時，便會發生深度退化 (Degradation) 的現象，一方面是模型中的非線性函數使得每層都會丟失一些原始特徵信息，另一方面在反向傳播的過程中，梯度逐漸更新至接近零，造成梯度消失問題，無法更新權重，最終使得辨識效果下降。ResNet 因此提出殘差學習 (Residual Learning)，透過 Shortcut Connection 概念設計了 Residual Block (圖 2-9) 改善了此問題，所謂殘差就是和原始特徵的差距，Residual Block 比起原來對特徵做學習而改對此差距做學習，最後才會將輸入特徵 x 加回去，如此一來即便殘差為零，也至少會讓特徵繼續傳遞不至於消失，更提出瓶頸結構

doi:10.6342/NTU202203562

(Bottleneck)，使用 1x1 的卷積神經網路降低參數量，使網路可以繼續加深並保持良好的影像辨識能力。其論文中根據層數不同又分為 ResNet50、ResNet101、ResNet152 等，而本研究會使用的 ResNet50 之結構如圖 2-10 所示：

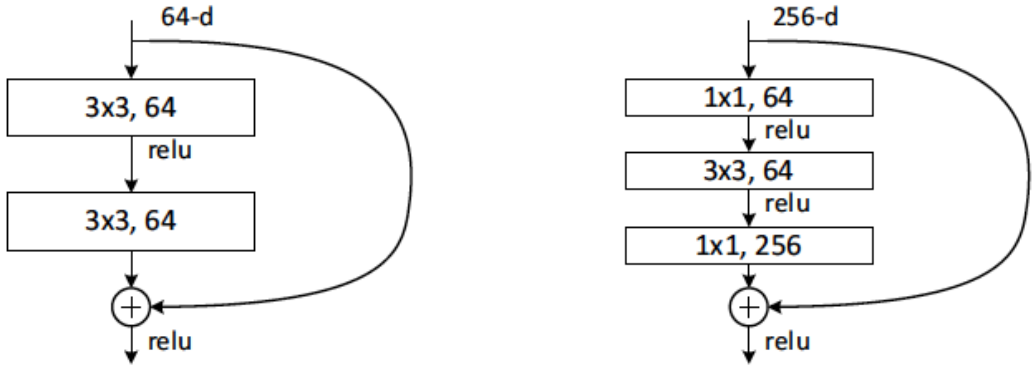


圖 2-9 Residual Block [15]

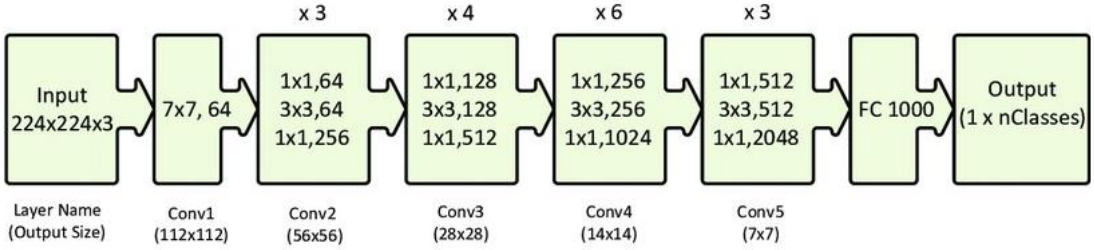


圖 2-10 Resnet50 網路架構

2. 物件偵測 (Object Detection)

物件偵測的目的為判斷輸入影像中包含的所有目標之位置與類別，如圖 2-5 (b)，當輸入該張影像時，一訓練良好的物件偵測模型應要能分別輸出所有圖中貓、狗、鴨的位置資訊與正確類別。現今的物件偵測網路大致可分為一階 (One Stage) 架構與二階 (Two Stage) 架構。

一階架構是指網路一次性在輸入影像中找出目標位置與類別，著名的網路如 2016 年 Redmon 等人提出的 YOLO[16]，其概念是直接將輸入影像分割成互不重合的小方格，分別透過卷積運算產生特徵圖，同時生成多個目標邊界框，再計算目標特徵是否存在與面積佔比來預測最終結果，其最大優點為偵測速度極快，然而缺點

doi:10.6342/NTU202203562

是對小目標因邊界框易重疊，容易預測錯誤位置或類別。後續又有對特徵提取網路和激活函數等持續做修改，如 2018 年提出之 YOLOv3，借用 ResNet 的思想引入殘差學習，在小目標準確率與運作效率上皆有顯著提升。

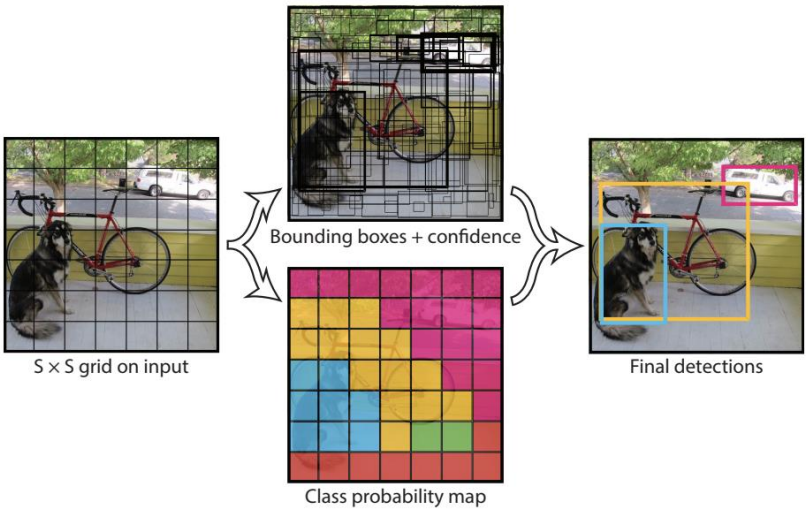


圖 2-11 YOLO 流程圖 [16]

二階架構是指網路需先在輸入影像中找出目標位置，再根據目標位置中的圖象資訊分辨出目標類別，著名網路如 2013 年 Girshick 等人提出的 R-CNN [17]，其先以 Selective Search 在輸入影像中選出數千個目標候選區域 (Region Proposal)，再使用卷積網路對所有目標候選區域提取特徵，最終通過支援向量機 (Support Vector Machine, SVM) 對特徵進行分類以找出目標。R-CNN 為首個可有效執行物件偵測的深度學習架構，然而其計算量極大，使用時需花費大量時間於計算，因此在實務應用上並不合適。而後 Girshick 於 2015 年提出 Fast R-CNN [18]，對特徵進行投影並以 Softmax 層取代 SVM 進行特徵分類，一如其名改善了耗時問題。2016 年 Ren 等人又提出 Faster R-CNN [19]，使用 Region Proposal Network (RPN) 取代 Selective Search 獲得目標候選區域，不必再計算數千個目標候選區域，更進一步節省運算時間，甚至達到可以實時物件偵測的程度 (表 2-1)。

doi:10.6342/NTU202203562

表 2-1 R-CNN、Fast R-CNN、Faster R-CNN 速度比較 [17, 18, 19]

	R-CNN	Fast R-CNN	Faster R-CNN
Test Time per Image	50 Seconds	2 Seconds	0.2 Seconds
Speed Up	1x	25x	250x

3. 影像分割 (Image Segmentation)

影像分割的目的為辨別輸入影像中各像素的所屬目標類別，如圖 2-5 (c)，當輸入該張影像時，一訓練良好的影像分割模型應要能分別輸出所有圖中貓、狗、鴨的像素與正確類別。而影像分割又分為語意分割 (Semantic Segmentation) 與實物分割 (Instance Segmentation)，語意分割只能對影像中所有像素進行分類，對於同類別之不同像素點，其無法理解各點之間的差別，所以無法分辨同類別的不同目標；而實物分割則加入了物件偵測的概念，除了辨別像素所屬類別外還能區分哪些像素屬於同一目標，並且即便同類別的不同目標也將是兩獨立物件。兩者辨識效果比較如圖 2-12 所示。

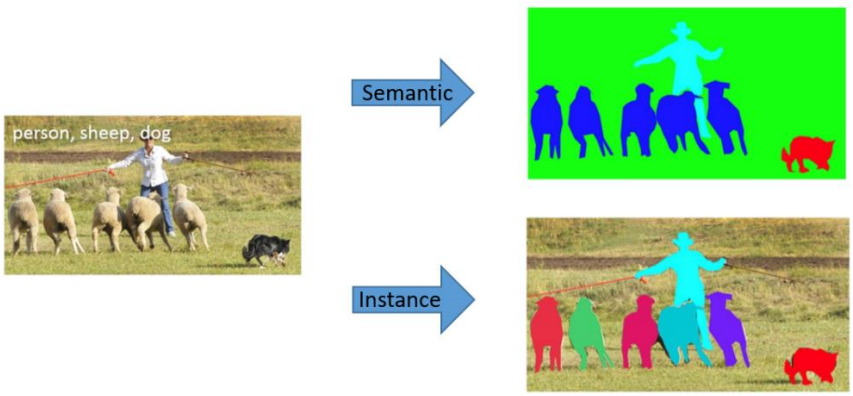


圖 2-12 語意分割與實物分割差異 [20]

最早的深度學習語意分割網路為 Long 等人於 2014 提出之 FCN [21]，其將 VGGNet 改為全卷積網路 (Fully Convolutional Networks)，因原最後為全連結層，只會輸出一維資訊而丟失空間訊息，無法找出各像素點間的關係，故改為卷積層並

doi:10.6342/NTU202203562

加上 Softmax 函數，網路的輸出便可改為二維的目標特徵平面，而因此平面經過多次降維，故需再以反卷積 (Deconvolution) 將特徵平面上採樣 (Upsampling) 來放大影像。然而，即便 FCN 融合了不同層的降維結果逐步做反卷積，仍舊會丟失影像中的不少空間資訊。於是 2015 年 Ronneberger 等人提出了 U-Net [22] 改善此問題，其在自編碼器的結構中加入對應的殘差連接 (Skip Connection)，其概念是將輸出寫成輸入和輸入經一非線性變換後的線性疊加，即將淺層的卷積特徵引入上採樣過程，進而保留更多的原始位置資訊 (圖 2-13)。

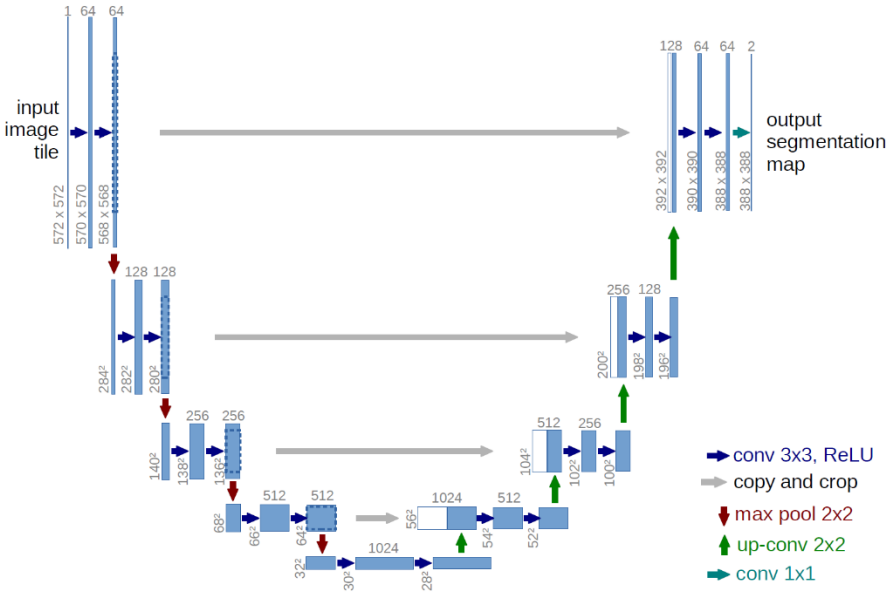


圖 2-13 U-Net 網路結構 [22]

實物分割網路的著名代表則為 2017 年 He 等人發表的 Mask R-CNN [23]，其以物件偵測網路 Faster R-CNN 的架構為基礎進行修改，以 RoiAlign 取代 RoI Pooling，不將邊界框位置取整數，降低了物件定位的誤差；且在獲得預測位置後串接 FCN 網路，對預測位置部分的影像進行分割計算，因此能對影像中的目標像素做出分類，同時又能區分出不同物件的位置，即便有多個像素群屬於同個類別也能辨識成不同像素群。

doi:10.6342/NTU202203562

2.2 領域自適應

在本節中，將對遷移學習 (Transfer Learning) 中的領域自適應與本研究用於瑕疵檢測的領域自適應模型做進一步的介紹。

2.2.1 遷移學習

在機器學習的實務上，常常會遇到要處理相似的數據分析卻不能直接使用同一模型的問題，這是因為不同資料之間存在領域偏移 (Domain Shift)，起因為製程參數或量測方法的變化造成的資料分布不同 [24]，若將兩組不同資料的參數敘述為因果因子 (Casual Factor) 與干擾因子 (Confounding Factor)，第一組數據下模型學習到的特徵可能屬於綜合兩因子後的分類或迴歸，如此在應用至另一組數據時，另一組數據的干擾因子便會致使這個分類或迴歸的結果不同，最終導致模型失效。

於是遷移學習的研究領域便應運而生，其最主要的的概念是將一現有機器學習模型，應用到一個尚未有解決辦法的相似資料分析之上，並能處理該相似數據分析而得到良好效果或是節省重頭訓練新模型的時間，例如利用原有的象棋策略模型來訓練出西洋棋的策略模型；利用原有的小提琴樂曲分析模型來訓練出鋼琴的樂曲分析模型；利用原有的腳踏車辨識模型來訓練出辨識摩托車的模型，如圖 2-14。

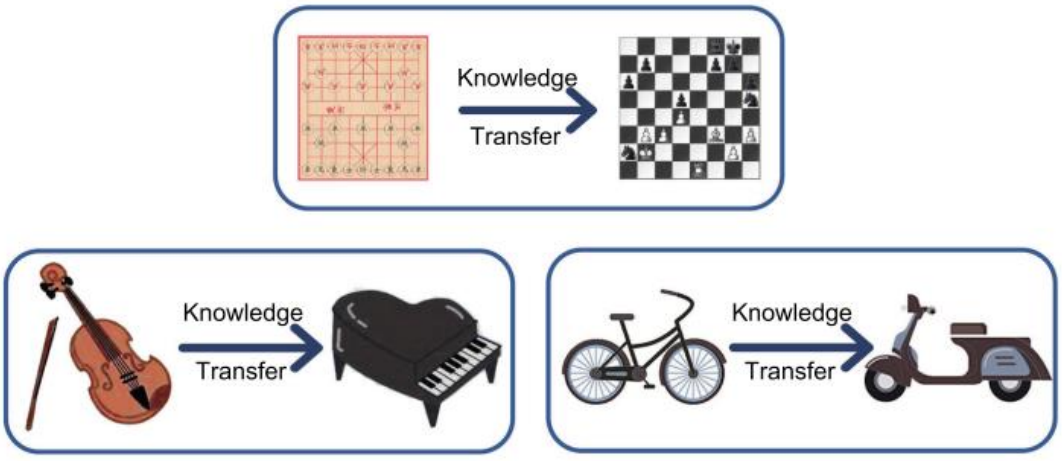


圖 2-14 遷移學習範例 [25]

doi:10.6342/NTU202203562

在遷移學習的領域中，原始的資料被稱為源域 (Source Domain)，而欲遷移過去的相似資料稱為目標域 (Target Domain)。而遷移學習的方法可以依照在訓練模型時源域與目標域之資料是否有標籤而分類，如圖 2-15，當源域與目標域的資料皆有標籤時，如果目標域有標籤的資料夠多，直接重新訓練即可；如果目標域有標籤的資料不多，可以將已訓練好之源域模型權重作為目標域模型權重的初始值 (Initial guess)，對有標籤的目標域資料訓練幾個迭代，或是將有標籤的目標域資料與源域資料合在一起訓練模型，都能取得不錯的辨識效果。

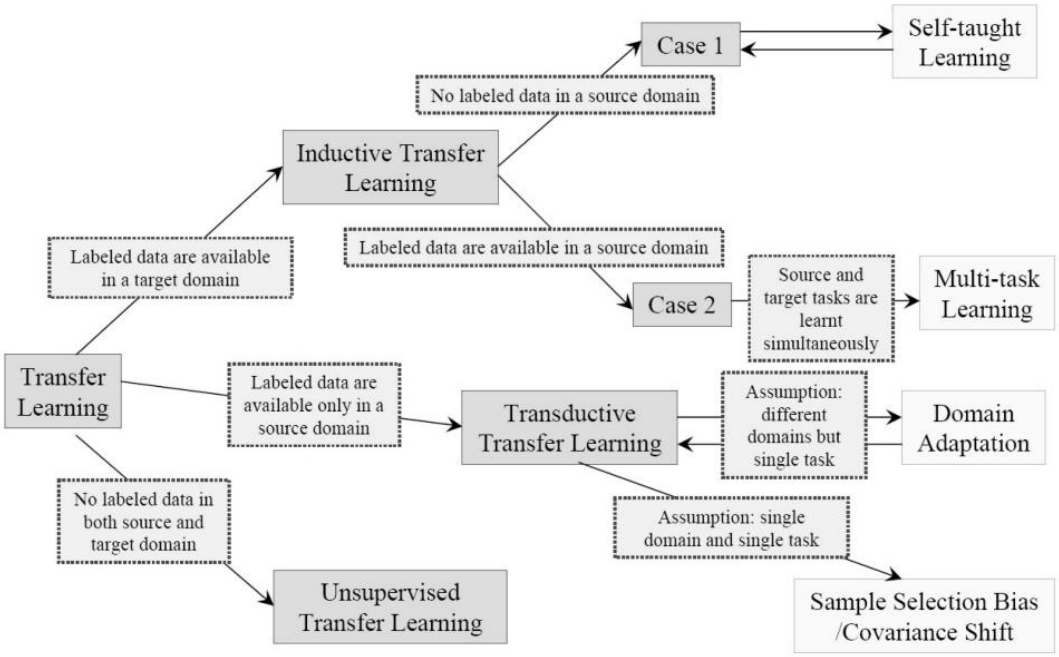


圖 2-15 遷移學習的解決辦法 [26]

然而本研究的目的是節省工業瑕疵檢測時，減少相似資料集之人工標籤成本。此種情況類似於目標域之待測資料並無標籤，這時適用的遷移學習方法之一便是領域自適應，其大略流程是先將源域與目標域的特徵提取出來，再找到一個方法使目標域的特徵分布能在某一特徵平面上和源域的特徵分布對齊 (Align)，如圖 2-16 所示，而且要是均勻的對齊，如圖 2-17 (b)，如此便能減少領域偏移的影響，使用源域資料訓練出的分類器對目標域資料進行辨識。特徵擷取器 (Feature Extractor) 可以採用經典分類模型的前幾層，由於現今的領域自適應研究多半採用 ResNet50

doi:10.6342/NTU202203562

作為特徵擷取器時的影像辨識效能作為比較標準，且以分類應用為主，加上在自動化光學檢測設備公司智泰科技的經驗中，ResNet50 常能有良好的瑕疵分類表現，如其於 2021 年的研究中[27]，ResNet50 比起 VGG19 和 InceptionV3 能對手機蓋板瑕疵有更高的分類準確率，因此本研究也將使用 ResNet50 作為特徵擷取器進行分類辨識，並將擷取的網路層命名為適應特徵層，另外，根據對齊的方法不同可將領域自適應模型概分為差異型網路 (Discrepancy-based methods) 與對抗型網路 (Adversarial methods)，在後續的小節中將分別對本研究中會使用的各個模型做基本介紹。

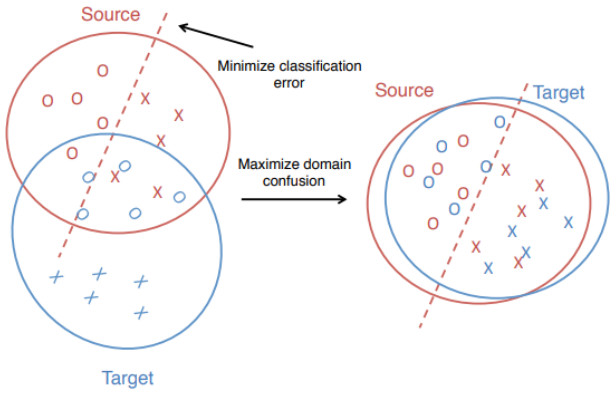


圖 2-16 源域與目標域資料對齊 [28]

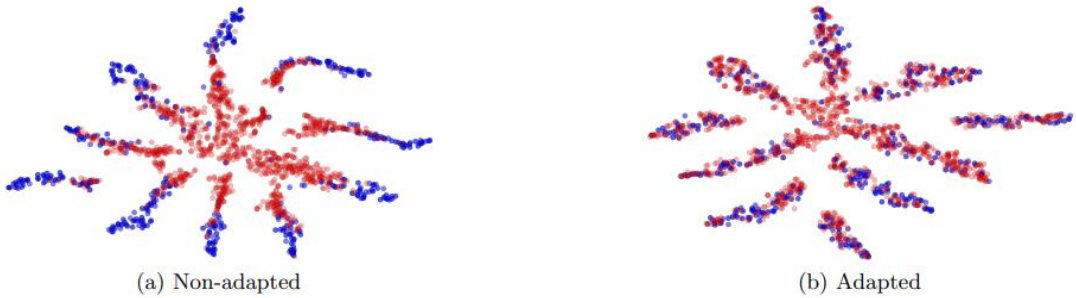


圖 2-17 (a) 不均勻對齊 (b) 均勻對齊 [29]

學界在比較領域自適應演算法時，早期常使用 office-31 資料集，其有 A (Amazon)、D (Digital single-lens reflex camera)、W (Webcam) 共 3 域資料各 31 類，

doi:10.6342/NTU202203562

源域與目標域能有 6 種排列組合，通常會以對 6 種分析組合之平均準確率作為比較基準；後又有較為複雜的 office-home 資料集，其有 Ar (Art)、Cl (Clipart)、Pr (Product)、Rw (Real-world) 共 4 域資料各 65 類，源域與目標域能有 12 種排列組合，並以 12 種分析組合之平均準確率作為比較基準，呈現時會以「→」符號表示分析組合，符號前為源域，符號後為目標域，例如「A→W」表示源域為 A 域資料，目標域為 W 域資料。

2.2.2 差異型網路

1. DAN (Deep Adaptation Network)

DAN [30] 是基於 DDC (Deep Domain Confusion) [28] 改良的。2014 年 Tzeng 等人提出了 DDC [28]，其網路架構如圖 2-18 所示，其分別使用一個 AlexNet 分類網路對源域資料和目標域資料進行訓練，除了共享兩網路權值，也計算兩網路的第 7 層輸出特徵間的 MMD (Maximum Mean Discrepancy) [31]，MMD 是計算兩域資料映射到 Hilbert space 後的差值總和，當這個值最小時能表示兩域資料差異最小，並以此設計損失函數來改變網路權重，以讓兩域資料機率分布情況相近，成功達到目標域分類的目的，在公開資料集 office-31 中的 A→W、D→W、W→D 此 3 種分析組合取得 81.2%的平均辨識準確率。當時論文並沒有說明 6 種分析組合的平均準確率，根據後來 DAN [30] 論文的描述，DDC 對 6 種分析組合的平均準確率為 70.6%。

doi:10.6342/NTU202203562

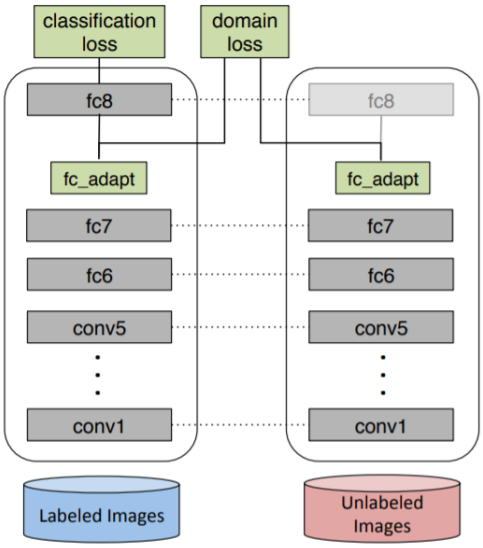


圖 2-18 DDC 網路架構 [28]

2015 年 Long 等人則在 DDC 的基礎上提出 DAN [30]，其對更多層間進行特徵差異計算，並且以 MK-MMD (Multi-Kernel Maximum Mean Discrepancy) 取代 MMD，即將映射至 Hilbert space 時需要的再生核 (Reproducing kernel)，由原本的單一核改為多個不同核的加權，成功使分類準確率再上層樓，在 office-31 的 6 種分析組合上取得了 72.9%的平均辨識準確率。

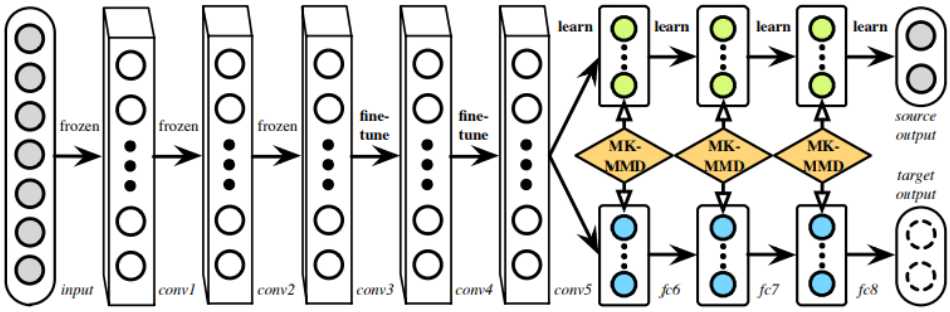


圖 2-19 DAN 網路架構 [30]

2. DEEPCORAL (Correlation Alignment for Deep Domain Adaptation)

2015 年 Sun 等人提出了 CORAL (Correlation Alignment) [32]，其先提取特徵，再利用線性變換使兩域特徵對齊，最後利用 SVM 訓練分類器，做出對無標籤目標

doi:10.6342/NTU202203562

域資料的分類。隔年團隊才又發表 DEEPCORAL [33]，將主架構換成 AlexNet，如其它差異型網路一樣共享源域與目標域資料訓練時的權值，並將變換函數改為非線性函數而設計出 CORAL loss，如公式 2-1，計算兩域特徵的協方差距離並利用反向傳播使網路更新權重，來達到使兩域特徵機率分布情況相似的目的，成功將 CORAL 與深度學習卷積網路架構結合，達到極高的分類準確率，在 office-31 的 6 種分析組合上取得了 72.1%的平均辨識準確率，且其中的 A→D、A→W 比 DAN 的辨識效果還高，整體架構如圖 2-20 所示。

$$L_{CORAL} = \frac{1}{4d^2}\|C_S - C_T\|_F^2 \tag{2-1}$$

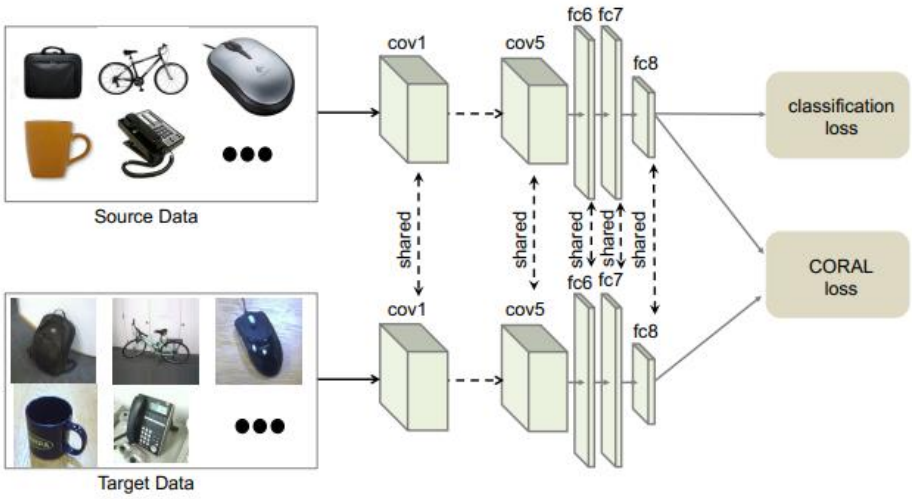


圖 2-20 DEEPCORAL 網路架構 [33]

3. DSAN (Deep Sub-domain Adaptation Network)

2020 年 Zhu 等人發表了 DSAN [34]，其分別將源域與目標域資料輸入兩特徵擷取器如 ResNet，並共享權值，並以 LMMD (Local Maximum Mean Discrepancy) (公式 2-2) 計算模型其中幾層兩域特徵的差異得到 loss，LMMD 可以看作是很多 MMD 函數的加權。另外，其也對子領域 (Subdomain) 進行分析研究，如圖 2-21，即相較於直接讓源域與目標域資料的機率分布情況相似，不如先各自根據分類器類別分出不同的子領域，再將源域的每個子領域去找適合的目標域子領域做差異

doi:10.6342/NTU202203562

計算，所以其在計算 loss 時還需要整個模型最終類別的信息，整個網路架構如圖 2-22 所示，在 office-31 的 6 種分析組合取得了 88.4%的平均辨識準確率。

$$\hat{d}_H(p,q) = \frac{1}{C}\sum_{c=1}^{C}\left\| \sum_{x_i^s \in D_s} \omega_i^{sc}\varphi(x_i^s) - \sum_{x_j^t \in D_t} \omega_j^{tc}\varphi(x_j^t) \right\|_H^2 \quad (2\text{-}2)$$

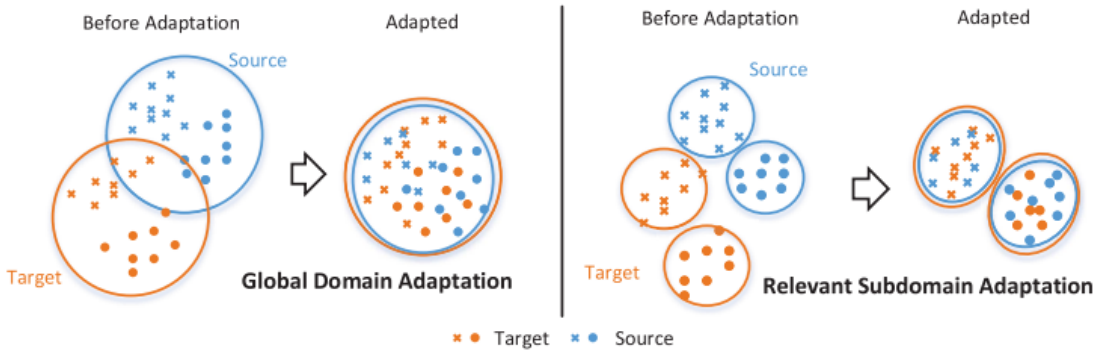


圖 2-21 子領域概念 [34]

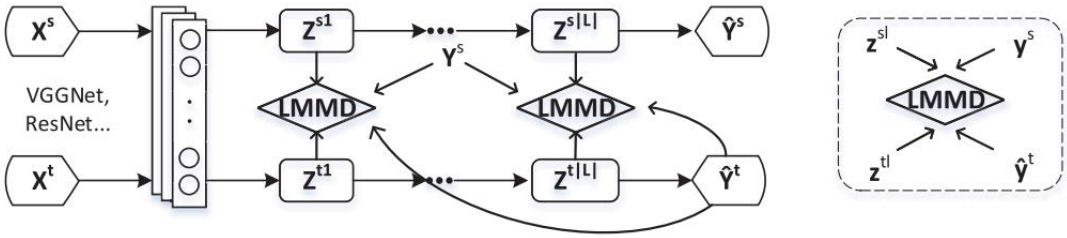


圖 2-22 DSAN 網路架構 [34]

doi:10.6342/NTU202203562

2.2.3 對抗型網路

1. DANN (Domain-Adversarial Neural Network)

2015 年，Ganin 等人發表了 DANN [29]，其借用了生成對抗網路 (Generative Adversarial Network, GAN) [35]的概念，整個網路架構如圖 2-23 所示，圖片中的上半部分可視為一完整的影像分類模型，當時文獻使用的是 AlexNet，專門對已有類別標籤的源域資料做訓練，並使用負對數似然 (Negative log-likelihood, NLL) 計算 loss L_y，而其又將網路中某一層的輸出特徵提取出來，輸入到圖 2-23 下半部分的另一分類器中，此分類器的目標是區分輸入圖像為有類別標籤之源域資料或無類別標籤之目標域資料，並計算 loss L_d，分支出來的部分被稱為鑑別器 (Discriminator)，而分支前的部分被稱為生成器 (Generator)。常理來說，loss 越小代表一分類器分類效果越好，於是 Ganin 等人便設計，讓L_y盡量小，代表網路對源域圖片有不錯的分類效果，同時，又讓L_d盡量大，即網路分不出究竟輸入的是源域資料還是目標域資料，統合起來，Ganin 等人讓整個網路的 loss 設為了$\min(L_y - L_d)$，便達成了能分類源域資料。而且在分類依據的特徵平面上，源域資料和目標域資料的特徵機率分布是幾乎重合的所以分不出來，也就解決了領域偏移的問題。此時便能使用源域資料訓練出的分類器對目標域資料進行分類，也成功對公開資料集取得了良好的辨識效果，以黑白手寫辨識集 MNIST 對彩色手寫辨識集 MNIST-M 取得了 76.66%的辨識準確率。

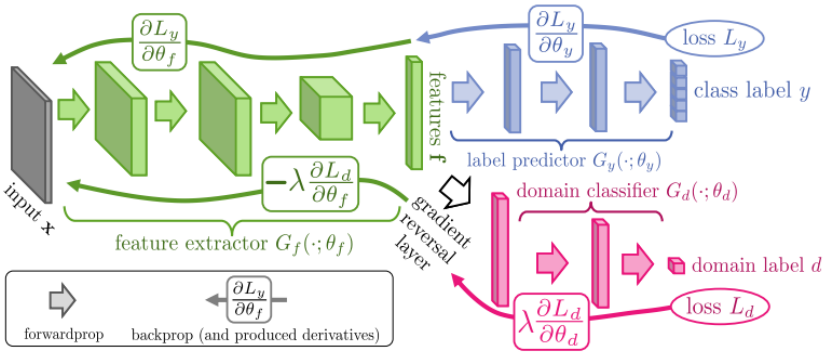
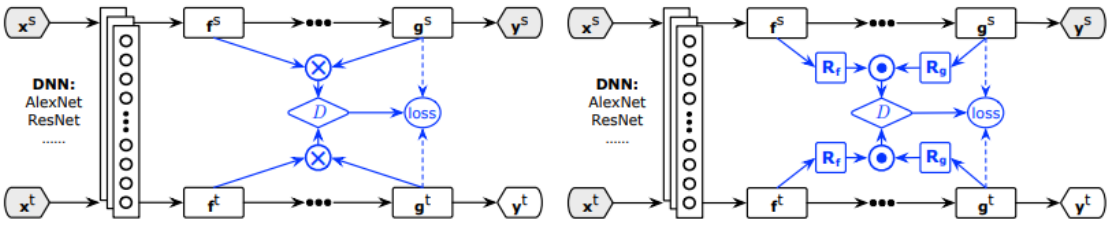


圖 2-23 DANN 網路架構 [29]

doi:10.6342/NTU202203562

2. CDAN (Conditional Domain Adversarial Network)

過往的對抗型領域自適應模型往往設定所有類別同等重要而做全特徵對齊，然而實際上有些類別確實會比較複雜而難以訓練，硬是去對齊常造成網路模型無法收斂或分類效果不佳。於是 2017 年 Long 等人提出了 CDAN [36]，其網路架構如圖 2-24 (a) 所示，其將提取出來的特徵層，和最後經過 softmax 函數後的分類預測機率做外積，才輸入 discriminator，換言之 discriminator 有接收到分類標籤相關的信息，另外，作者又進一步提出熵調整 (Entropy Conditioning)，設計了一個熵函數 R 對提取特徵和預測機率都做轉換，形成如圖 2-24 (b) 的架構，其意義是做加權計算，讓預測機率較為肯定時對模型的影響權重大一點，一定程度上的緩解了全特徵對齊的問題，在 office-31 的 6 種分析組合上取得了 87.7%的平均辨識準確率。在後世的論文中，有使用熵調整的模型被記做 CDAN+E，也可以應用到其他網路如 DANN 上，記做 DANN+E。



(a) Multilinear Conditioning (b) Randomized Multilinear Conditioning

圖 2-24 CDAN 網路架構 [36]

3. DAAN (Dynamic Adversarial Adaptation Network)

此前的研究多專注於邊緣對齊 (Marginal Adaptation)，即對齊時整個特徵平面上的分布範圍重合度，或專注於局部對齊 (Conditional Adaptation)，即各類別的重合度，對於不同課題，專注於不同面向的模型會有不同效果，如圖 2-25。於是 2019 年 Yu 等人提出 DAAN [37]，其網路架構如圖 2-26 所示，除了一個邊緣對齊用的 discriminator，其也如其他專注於局部對齊的研究，對各分類類別獨立安置一個 discriminator，並提出了動態對抗因子 (Dynamic Adversarial Factor)，透過建立邊緣

doi:10.6342/NTU202203562

對齊和局部對齊用的 discriminator 間 loss 的關係式，讓模型能動態的去改變何時該專注於何種對齊，也讓訓練者可以觀察分析，若動態對抗因子越大，表示局部對齊越重要，反之越小時邊緣對齊越重要，成功在公開資料集 office-home 的 12 種分析組合上取得 61.8%的平均辨識準確率，相對於其論文測試 DANN 時得到的 57.6%準確率略高。然而使用過多的 discriminator 會導致訓練時間大幅變長，於是 Yu 等人使用了隨機梯度下降 (SGD)，一次只用一個資料做擬合而不是整個批量，雖然缺點是收斂方向會隨機而很可能只收斂到局部最佳解，但確實大幅下降了訓練時間，且因動態對抗因子而使整體準確率比前人高，證實依照資料分布的特性考慮邊緣對齊還是局部對齊是重要的。

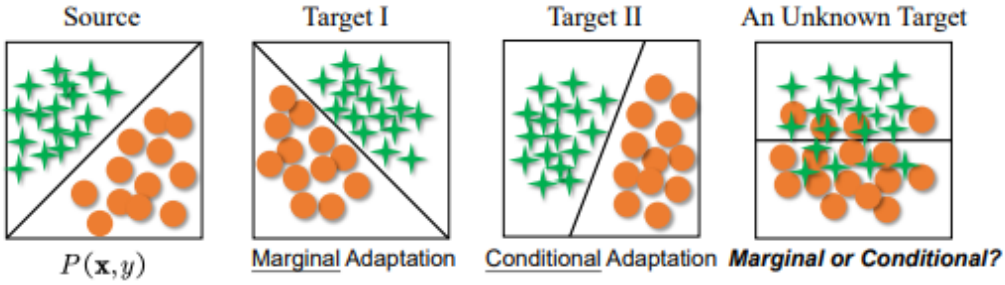


圖 2-25 邊緣對齊與局部對齊 [37]

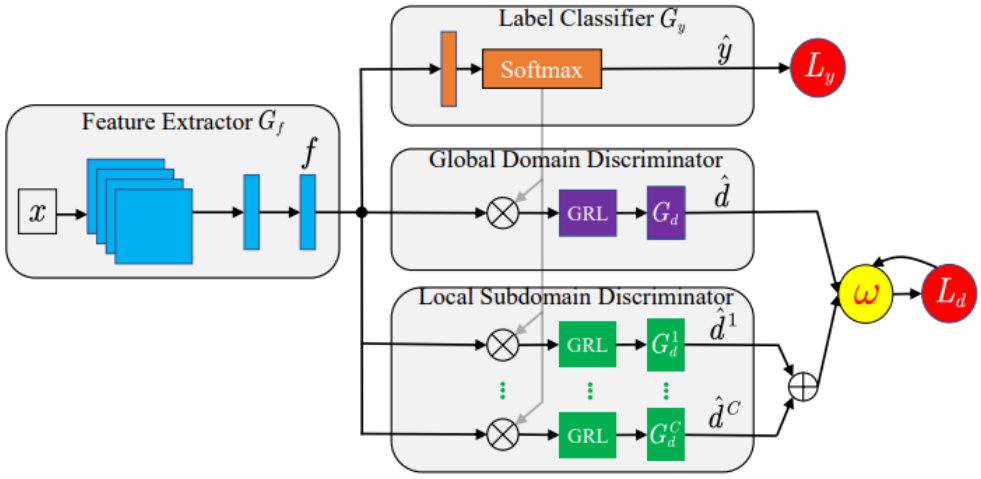


圖 2-26 DAAN 網路架構 [37]

doi:10.6342/NTU202203562

4. GVB (Domain Adaptation with Gradually Vanishing Bridge)

2020 年 Cui 等人提出 GVB [38]，其認為直接將距離遙遠的源域與目標域距離最小化太過困難，換句話說，直接遷移對齊太過困難，因此其提出橋函數 (Bridge Layer) 和中間域 (Intermediate Domain) 概念。在 generator 上，可以透過橋函數，以逐漸拉近兩域距離的想法來逐漸擴大兩域重疊區域，建立中間域，如圖 2-27，能有效降低中間域中源域和目標域各自的特殊特徵的影響，當已重疊部分越多想進行直接對齊就越簡單，並透過計算中間域的生成情況計算 generator 的 loss，另外作者也用相似觀念設計了 discriminator 的橋函數，主因是 discriminator 通常是一個沒有使用預訓練模型的分類器，隨機初始化而容易陷入局部最小值 (Local Minimum)，如此可以提升這個 discriminator 的辨別能力，整體模型架構如圖 2-28 所示，其模型 GVB-GD 成功在 office-home 的 12 種分析組合上取得 70.4%的平均辨識準確率。

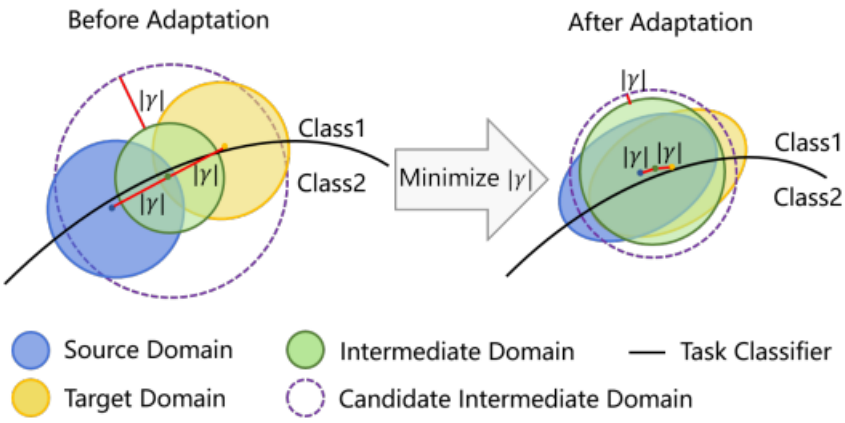


圖 2-27 中間域概念 [38]

doi:10.6342/NTU202203562

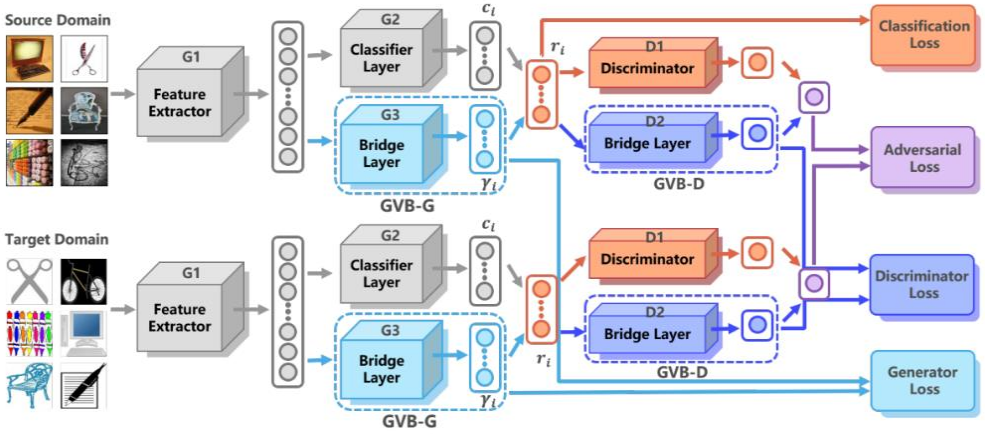


圖 2-28 GVB 網路架構 [38]

模型上 Cui 等人建立了一個基本對抗型領域自適應模型記作 Baseline，其不完全像 DANN，而是參考了許多模型而生 [36][39][40]，在本研究中將記為 GVB，而其餘模型記法將仿照 Cui 等人的規則，若對 GVB 的 generator 使用橋函數，記為 GVB-G；若對 GVB 的 discriminator 使用橋函數，記為 GVB-D；若對 GVB 的 generator 和 discriminator 皆使用橋函數，記為 GVB-GD。另外，Cui 等人也有提到，橋函數能應用到其他對抗型領域自適應模型上，故有 CDAN-G、CDAN-D、CDAN-GD 等。

doi:10.6342/NTU202203562

2.3 研究方向可行性分析

在探討了許多現有的領域自適應模型後，確認了此技術在各公開資料集上都有成功提升對於無標籤影像的辨識準確率，由於許多 CNN 相關的分支研究領域都有在工業瑕疵檢測上應用的成功案例，故本研究嘗試以領域自適應技術處理業界遇到的新產線瑕疵標記成本問題。

小結

在本章中回顧了深度學習技術應用於影像辨識的發展，且因領域自適應技術的誕生背景和業界所遇之新產線樣品無瑕疵標籤問題相似，故探討了學界現有的解決方法後，本研究將嘗試以領域自適應技術處理標記成本問題，並以木皮、布匹瑕疵為例，比較不同領域自適應演算法的辨識效果。

doi:10.6342/NTU202203562

第三章　實驗流程與方法

本章將詳述圖 1-2 研究流程圖中所示之步驟，包括木皮、布匹、公開金屬表面瑕疵影像資料集整理的方法；以及所使用的各種神經網路模型的實現方法，包括傳統 CNN 檢測模型、多種領域自適應辨識模型、為分析適應特徵層改變之影響所建立之模型；還有後續討論需要的可視化工具的原理與實踐方法，包括混淆矩陣、T-SNE 資料降維法與 Grad-CAM。

3.1　資料集建立

本節將依流程順序闡述由實體產線上木皮與布匹瑕疵樣本至建立瑕疵影像資料的方法，而本研究為於最終提出適合工業瑕疵檢測用之領域自適應技術應用流程，亦使用中國東北大學的 Kechen Song 教授的公開熱軋鋼金屬表面資料集 (NEU-CLS) [41] 進行實驗，因其為已整理好之影像檔案，故將直接於 3.1.3 小節說明。

3.1.1　光學取像

光學取像的方式主要分為面掃描式 (Area Scan) 與線掃描式 (Line Scan)。面掃描式每次取像便會得到一張二維平面影像，而線掃描式每次取像只會取得單一長度的一維線段影像，透過相機與取像目標的相對運動，整合多段影像形成完整二維平面影像。兩者的優缺點如表 3-1 所示：

表 3-1 面掃描式與線掃描式取像比較

	面掃描式	線掃描式
優點	● 取像速度快 ● 架設容易 ● 可直觀量測形狀、面積、位置等	● 取得影像分辨率高，適合用於精確量測 ● 適用動態取像 ● 成像尺寸靈活
缺點	● 取像視野範圍受限	● 架設複雜，需搭配運輸帶 ● 取像時間較長

doi:10.6342/NTU202203562

單作為取得樣本影像的手段，兩者皆可達成目的。然而線掃描式的架構搭建時間長且複雜，成本較高，故本研究選擇使用面掃描式取像結構，此種方式最大缺點是當樣本寬大時，相機和樣品要保持足夠距離才拍的完整，同時相機的解析度就要相應增加，而越高解析度的相機會越難取得，成本也越高，不過木皮與布匹皆屬狹長型樣品，只要在寬度上能完整拍攝樣品即可。本研究選用的鏡頭為 Teledyne FLIR 的 BFLY-PGE-50S5C-C 彩色面相機 (圖 3-1)，其解析度有 2K，足夠捕捉木皮與布匹的瑕疵特徵，其規格如表 3-2 所示。

圖 3-1 BFLY-PGE-50S5C-C

表 3-2 面相機規格 [42]

項目	規格
品牌	Teledyne FLIR
型號	BFLY-PGE-50S5C-C
解析度	2448 x 2048
取像色彩	Color
像素大小	3.45μm
幀率	22FPS
工作溫度	0 ~ 45 °C
尺寸	29 mm x 29 mm x 30 mm

光學取像整體架構為一載物台，相機則固定在載物台上方適當距離，光源選擇上，本研究使用同軸光以獲得均勻的照明，減少反光帶來的瑕疵影像干擾，同時於

doi:10.6342/NTU202203562

載物台下方安裝一背光板，主要用於淺色布料拍攝，以凸顯瑕疵特徵，實際架構如圖 3-2 所示。

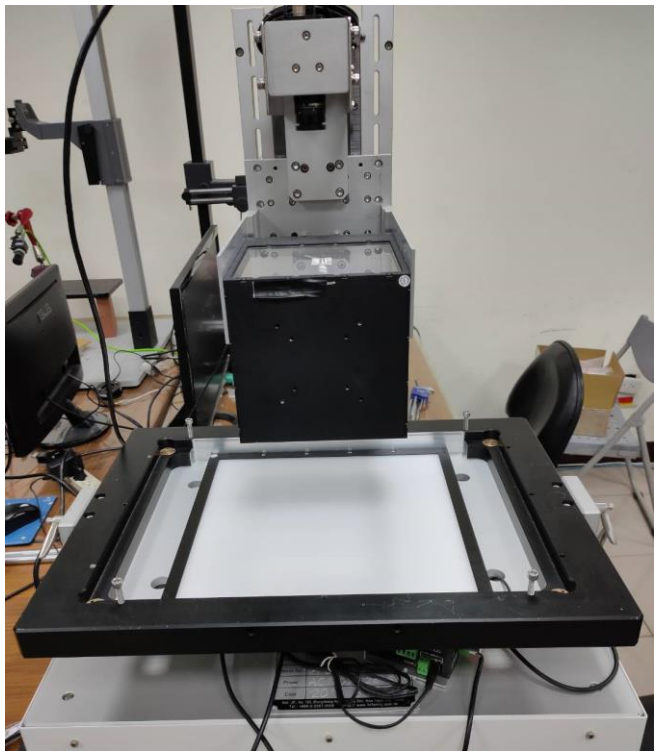


圖 3-2 光學取像架構圖

完成光學取像架構的搭建後，便進行木材、布匹瑕疵樣品影像的拍攝。在木皮種類的選擇上，本研究選擇了以白橡木和胡桃木進行領域自適應模型訓練，一方面是因瑕疵樣本缺稀，研究所能取得的此二種木皮瑕疵樣本相對較充足，另一方面是因在木皮成色與紋理上，白橡木是相對白的木種，而胡桃木則是相對深色的木種，此二種木皮的瑕疵特徵處於兩種極端，特徵差異極大，在模型遷移的難度上極高，研究雖在取得多種木皮上有困難，但對於其他瑕疵特徵差異更小的木皮種類，本研究結果的適用性將是可以被期待的；布匹的選擇上，研究也選擇了瑕疵特徵差異極大的深色布與淺色布，各自形成一領域的資料集，深色布包含純黑布、藍黑布等，研究基於方便統一紀錄為黑布，淺色布包含純白布、米白布等，研究基於方便統一紀錄為白布。經拍攝後得到的影像張數如表 3-3 所示，影像尺寸為 2448 x 2048 pixels，

doi:10.6342/NTU202203562

而本研究後續會對這些影像進行裁取，無瑕疵以及瑕疵的樣本都能藉由裁取影像中對應的部分得到，並考慮資料平衡後，選取合適的樣本數量進行分析。

表 3-3 瑕疵樣本影像數量

	白橡木	胡桃木	深色布(黑布)	淺色布(白布)
影像數量(張)	141	312	93	107

3.1.2 瑕疵影像裁取

木皮上最嚴重的瑕疵主要有兩種，一是木珠，為木頭生長時枝椏的分岔處，通常會是螺旋圓球的樣子，二是破裂，為運送途中或加工時受到過度擠壓而斷裂處，此二種瑕疵皆會導致木皮結構脆弱，在製成後續產品時容易影響產品可靠度；布匹的瑕疵主要有摺痕、破洞、勾痕、色汙，多由產線上的不正常突出物、油漬或人為觸碰所造成，會影響布匹的美觀。故這些瑕疵需要於品質檢驗階段挑除。

由於本研究要訓練瑕疵分類模型，資料集內每張影像只包含一種瑕疵，以利模型訓練。故需要以人工方式對影像進行裁取，找出瑕疵的部分。為此，本研究利用 Python 撰寫簡易的影像裁取工具，利用 OpenCV 函式庫對影像進行開啟、儲存等動作，利用 EasyGUI 函式庫回傳滑鼠位置資訊，整體介面如圖 3-3 所示，只要在目標區域按下相對的瑕疵代號鍵盤，框選的區域影像便會被存儲至對應的資料夾內，通過將同類別瑕疵整理在同一資料夾內，也做到了標籤的目的，方便後續訓練模型時導入程式。而本研究將框選區域大小設為 400 x 400 pixels，主要是測試後確認可以完整包含單一木珠瑕疵的特徵範圍，其餘瑕疵之特徵尺寸皆在最大之木珠尺寸之下。

doi:10.6342/NTU202203562

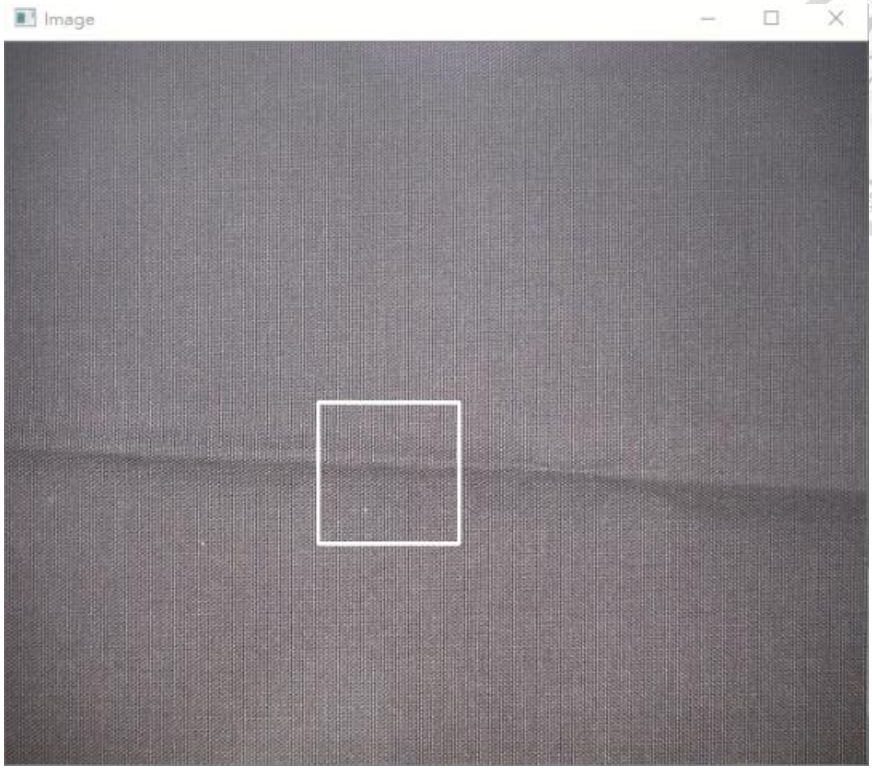
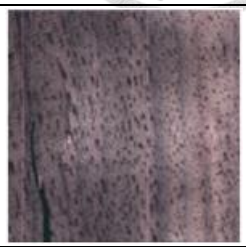
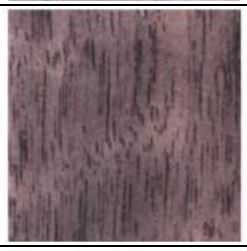
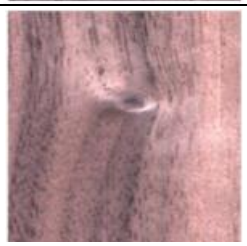


圖 3-3 影像裁取工具介面

另外，在工業瑕疵檢測時，正常樣品的數量還是佔大多數，因此本研究於木皮和布匹資料集皆有另設一類良品影像。經過影像裁切之後，各類別瑕疵的範例圖與所得影像張數如表 3-4 與表 3-5 所示。

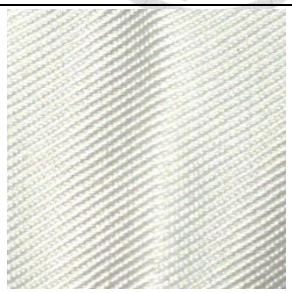
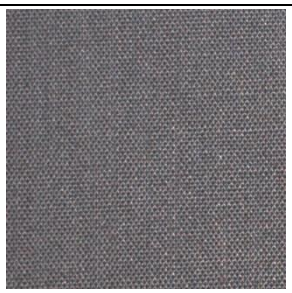
doi:10.6342/NTU202203562

表 3-4 木皮裁切後瑕疵影像數量與範例圖

分類名稱	白橡木		胡桃木	
	數量(張)	範例圖	數量(張)	範例圖
破裂	84		56	
良品	311		308	
木珠	118		122	

doi:10.6342/NTU202203562

表 3-5 布匹裁切後瑕疵影像數量與範例圖

分類名稱	黑布		白布	
	數量(張)	範例圖	數量(張)	範例圖
摺痕	169		187	
良品	299		289	
破洞	159		174	
勾痕	157		179	
色汙	147		200	

doi:10.6342/NTU202203562

3.1.3 製作訓練資料集

在訓練神經網路模型時，當訓練時某一類別的影像多於其他類別影像太多，容易使模型的預測結果往該類別靠攏，換言之，模型只要預測影像數量最多的類別，便能得到不錯的準確率，這樣的模型完全失去了特徵辨識的意義。本研究為了避免發生此情況，對前一小節得到之木皮與布匹瑕疵影像各自做了資料平衡，通過 Python 程式的 random 函式隨機捨棄較多影像之類別內的影像，雖然調整後各類別的影像數量仍不盡相同，但差距已縮小，且根據 Google Developers 的機器學習線上課程整理[43]，只要訓練出的分類模型準確率良好，便表示資料不平衡的影響在此分類任務中極其微小可以忽略，這部分的驗證本研究將於第 4.1 節說明。同時，由於本研究後續會訓練傳統 CNN 模型做為比較，需要一訓練集與一測試集以計算模型辨識準確率，故將各類別影像按 8：2 的比例隨機分為訓練集與測試集，然而因此各資料集剩餘影像數量非常稀少，為避免資料集影像稀少導致模型難以收斂，也避免單張影像對準確率計算上影響過大，本研究又再使用資料增強 (Data augmentation) 的方式增加影像數量。資料增強為訓練神經網路模型時應對資料不足常用的方法，藉由對影像進行旋轉、平移、色調變換、縮放、施加雜訊等處理，產生新的影像。本研究對木皮、布匹的訓練集與測試集各自於每 90° 施以旋轉與鏡射，因旋轉和鏡射並不會造成瑕疵特徵的形變，於是 1 張影像能產生 7 張影像。並且為了程式撰寫方便，給予各類別瑕疵英文代號，最終各資料集的張數如表 3-6 與表 3-7 所示。

doi:10.6342/NTU202203562

表 3-6 木皮資料集影像張數

單位：(張)	白橡木 (oak)		胡桃木 (walnut)	
	訓練集 (oak_train)	測試集 (oak_test)	訓練集 (wal_train)	測試集 (wal_test)
破裂 (crack)	544	128	360	88
良品 (good)	712	160	448	104
木珠 (knot)	760	184	424	104

表 3-7 布匹資料集影像張數

單位：(張)	黑布 (black)		白布 (white)	
	訓練集 (black_train)	測試集 (black_test)	訓練集 (white_train)	測試集 (white_test)
摺痕 (crease)	1088	264	1200	296
良品 (good)	1024	248	1120	272
破洞 (hole)	1008	248	1152	280
勾痕 (hook)	944	232	1280	320
色汙 (stain)	1024	256	1240	304

本研究亦使用熱軋鋼金屬表面瑕疵公開資料集 (NEU-CLS) 進行實驗測試，NEU-CLS 有兩種視野範圍下的金屬瑕疵影像資料，分別是影像尺寸皆為 200 x 200 pixels 的，和影像尺寸皆為 64 x 64 pixels 的，分別給予英文代號 s200 與 s64。由於視野範圍不同也是領域偏移的一種，故本研究將此二資料集做為領域自適應的源域和目標域來使用。兩資料集各包含六種瑕疵，分別是黑點 (Rolled-in scale, RS)、黑斑 (Patches, PA)、龜裂 (Crazing, CR)、孔蝕 (Pitted surface, PS)、夾雜物 (Inclusion, IN)、刮痕 (Scratches, SC)，其形貌範例如圖 3-4 所示。

doi:10.6342/NTU202203562

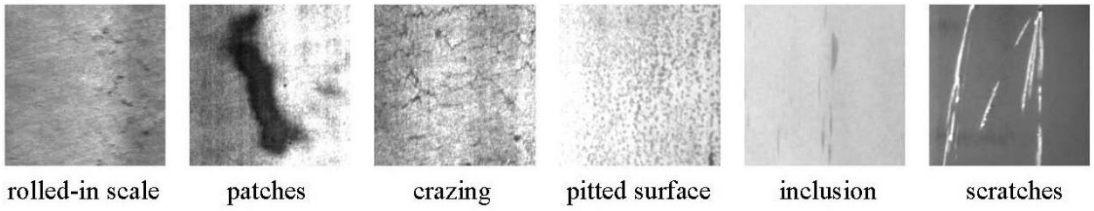


圖 3-4 NEU-CLS 瑕疵範例 [41]

本公開資料集在整理者 Kechen Song 的研究 [41] 及大量其他學界研究中皆有被使用，在資料皆有標籤的情況下訓練出準確率高的分類模型的難度不高，故本研究便不再對其進行資料平衡，詳細驗證會於第 4.1 節說明。不過為了後續實驗，本研究將 s200 與 s64 資料集亦按照處理木皮與布匹資料集時的方法，各自以 8：2 的比例將影像分為訓練集與測試集。另外，由於過短的標籤名稱有時候會觸發訓練模型時的路徑報錯，例如讀取資料時路徑間的「\」符號和標籤名稱「n」寫在一起，形成「\n」，被模型底層程式誤認為換行指令，進而導致模型訓練程式無法執行，因此，本研究在其原瑕疵類別的英文代號前又多各加上「cc」以避免路徑報錯問題，最終兩資料集的金屬瑕疵影像張數與代號如表 3-8 所示。

表 3-8 金屬表面瑕疵資料集影像張數

單位：(張)	s200		s64	
	訓練集 (s200_train)	測試集 (s200_test)	訓練集 (s64_train)	測試集 (s64_test)
黑點 (ccrs)	240	60	1272	317
黑斑 (ccpa)	240	60	919	229
龜裂 (cccr)	240	60	968	242
孔蝕 (ccps)	240	60	638	159
夾雜物 (ccin)	240	60	620	155
刮痕 (ccsc)	240	60	619	154

doi:10.6342/NTU202203562

3.2 分類模型建立

經上節取得三大類瑕疵的資料集後，本研究將以多種神經網路模型對三大資料集進行辨識效果測試與分析。本節將詳述本研究建立各神經網路的方法，包含訓練用硬體的架設與程式碼的撰寫。

3.2.1 建立訓練環境

在訓練深度學習模型前，要先建立訓練環境，包含電腦的架設與虛擬環境的搭建。

GPU 最早是為了加速 3D 彩現而生的處理器，具有大量小而精於圖形計算的核心，2012 年 AlexNet 在 ImageNet 影像辨識大賽上聲名大噪後，GPU 便成為了訓練深度學習模型的主流配備，尤其訓練卷積神經網路需要大量的矩陣運算來計算模型參數，GPU 透過多核心執行平行運算，訓練模型的效率比使用 CPU 高了許多。作為 GPU 最早的提出公司，NVIDIA 也持續不斷推出各種 GPU 來滿足不同族群的使用需求，並打造了 CUDA 平行運算架構和 cuDNN（CUDA Deep Neural Network library）加速庫來支援深度學習網路運算。GPU 和主記憶體 (Random Access Memory, RAM) 的選擇直接影響了訓練模型時能同時處理的影像數量與影像大小，本研究使用的 GPU 為 GeForce GTX 1080 Ti (圖 3-5)，其規格和整體電腦規格如表 3-9 與表 3-10 所示。

圖 3-5 GeForce GTX 1080 Ti [44]

doi:10.6342/NTU202203562

表 3-9 圖形處理器規格 [44]

項目	規格
品牌	NVIDIA
型號	GeForce GTX 1080 Ti
CUDA® 核心	3584 個
加速時脈	1582 MHz
記憶體時脈	11 Gbps
記憶體大小	11 GB
記憶體介面寬度	352-bit
記憶體頻寬	484 GB/s

表 3-10 電腦設備規格表

中央處理器	Intel® Core™ i7-8700
圖形處理器	NVIDIA GeForce GTX 1080 Ti
主記憶體	DDR4 3200 16GB x2
硬碟	ADATA SX8200 Pro 512G
作業系統	Windows 10

程式語言的使用上，本研究選擇以 Python 撰寫。Python 以其直觀的語法成為訓練深度學習網路的主流語言，且隨著深度學習研究者不斷研發並開源相關函式庫，更多的套件能被整合並呼叫，大幅降低使用的門檻，更加鞏固 Python 在深度學習領域的主流地位，如此往復形成正向循環。另外，本研究以廣泛使用的 Python 數據科學平台 Anaconda 搭建虛擬環境，如此便能將各個專案分開處理，並導入 Pytorch 作為深度學習框架。Pytorch 是 Facebook 開源的深度學習框架，能將矩陣轉為張量 (Tensor)，在 GPU 上運算，因其語法簡潔易懂，而在需要快速搭建不同神經網路進行實驗的學界受到廣泛使用。詳細的版本配置如表 3-11 所示。

doi:10.6342/NTU202203562

表 3-11 軟體環境配置表

項目	版本
Anaconda	4.6.14
CUDA	10.1.105
cuDNN	7.6.5
Python	3.7.7
Pytorch	1.10.1

3.2.2 訓練 ResNet50 模型

為了和領域自適應模型的瑕疵辨識效果做比較，本研究先使用目前業界常用的深度學習瑕疵分類方法，即以傳統的 CNN 神經網路訓練瑕疵分類模型，例如：以有標籤的 oak_train 資料集訓練 CNN 木皮瑕疵分類模型，並對無標籤的 wal_test 資料集進行預測，再和 wal_test 資料集的實際標籤比對並計算準確率。本研究以 ResNet50 為使用模型，如圖 3-6 所示，其第一層為一卷積層，而後接了 4 個 layer，每層 layer 內分別有 3、4、6、3 個 bottleneck 結構的 Residual Block，每個 Residual Block 有 3 層，共 48 層，最後再經過一層平均池化層 (Average pool)，總共 50 層，透過呼叫 torchvision.models 函式庫即可完成，最後再根據要分類的類別數量加上一層全連接層 (Fully Connected Layer) 即可輸出分類結果。

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

圖 3-6 Pytorch 對不同層數 ResNet 的實現 [45]

doi:10.6342/NTU202203562

另外，透過 torch.utils.data 函式庫內的 DataLoader 函式，能將各瑕疵影像讀入，並依據其影像資料夾名稱形成分類標籤，同時在輸入模型前，所有影像都會利用 torchvision.transforms 函式庫調整成 256 x 256 pixels 的尺寸，並且模型初始會使用對 ImageNet 的預訓練權重，以更容易收斂，此部分架構也會作為後續領域自適應模型的特徵擷取器使用。

本研究也嘗試以資料增強的方式提升傳統 CNN 模型的瑕疵辨識準確率，經過觀察，認為木皮、布匹的瑕疵影像特性，最大的差別是色澤差異，故對源域資料施以色調與亮度變換，增加源域資料的變異性，嘗試分類目標域資料。程式實現上，使用 Albumentations 函式庫的 RandomBrightnessContrast 和 RGBshift 函式，兩者使用時皆需設置發生機率，本研究皆設為 0.5，分別表示每張源域資料集有 0.5 的機率在影像 RGB 的灰階值出現±0.2 倍和±20 的隨機變換。

3.2.3 訓練領域自適應模型

本研究比較第二章提及的多種領域自適應模型應用於木皮與布匹瑕疵資料集的辨識效果，包含 DAN、DeepCoral、DSAN、DANN、CDAN、DAAN、GVB，使用兩個 GitHub 原始碼平台上的專案訓練瑕疵分類模型，一是 Jindong Wang 的 transferlearning 專案 [46]，另一則是 GVB 論文的開源程式碼，使用時需要輸入的資訊如表 3-12 所示，另外，一些超參數的設定將於 3.2.5 節說明。

表 3-12 模型訓練之需要資訊

項目	範例
源域資料集路徑	路徑
目標域資料集路徑	路徑
模型儲存路徑	路徑
特徵擷取器	ResNet50、VGG16 等
使用的領域自適應模型架構	DAN、DANN 等

doi:10.6342/NTU202203562

其中 DANN 兩套程式碼皆有，本研究選擇以 GVB 論文的開源程式碼實現，並且全部的對抗型領域自適應模型的基本 discriminator 設為和 GVB 論文的開源程式碼一樣 (圖 3-7)，在經過特徵擷取器 ResNet50 後還需要經過一層 bottleneck 層降成 1 x 1 張量，才會送入 discriminator。

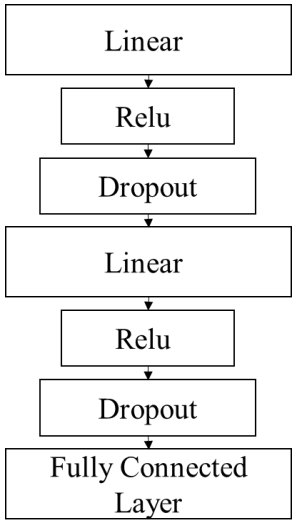


圖 3-7 GVB 論文的開源程式碼實現的基本 discriminator

本研究利用此二套程式碼在木皮、布匹、金屬表面瑕疵資料集各自的兩種資料集內，訓練領域自適應模型並比較瑕疵辨識準確率，例如：以有標籤的 oak_train 資料集和無標籤的 wal_test 資料集訓練領域自適應木皮瑕疵分類模型，並對無標籤的 wal_test 資料集產生預測標籤，再和 wal_test 資料集的標籤資料計算準確率。

3.2.4 訓練多鑑別器領域自適應模型

在觀察了上小節的實驗結果後，本研究希望進一步提升對抗型領域自適應模型的準確率，在 DAN [30] 的文獻中，其分析了 DDC 網路適合的適應特徵層組合，最終對 AlexNet 的 6、7、8 層皆進行了領域適應，成功提高辨識準確率，故本研究也嘗試分析 DANN 的適應特徵層對模型準確率的影響，為此，本研究改寫原 DANN 的網路架構，根據前述中對 DANN 的實現，原先 DANN 的 discriminator 接在 ResNet50 的最後的平均池化層後的 bottleneck 層後，亦即針對第四個大 layer (圖 3-

doi:10.6342/NTU202203562

6 的 conv5_x) 最後的卷積特徵做領域自適應，故本研究仿照而在另外三個 layer 的後方，皆各接上一層平均池化層與 bottleneck 層，並各自送入一個 discriminator，形成一多鑑別器的領域自適應模型 (圖 3-8)，並依影像在模型中經過的 layer 順序為相關參數命名。本研究便以此分析所有適應特徵層組合對模型準確率的影響。以四個 discriminator 皆使用的情況為例，原 DANN 論文中 discriminator 部份的 loss 就會是四個 discriminator 的 loss 的平均值。

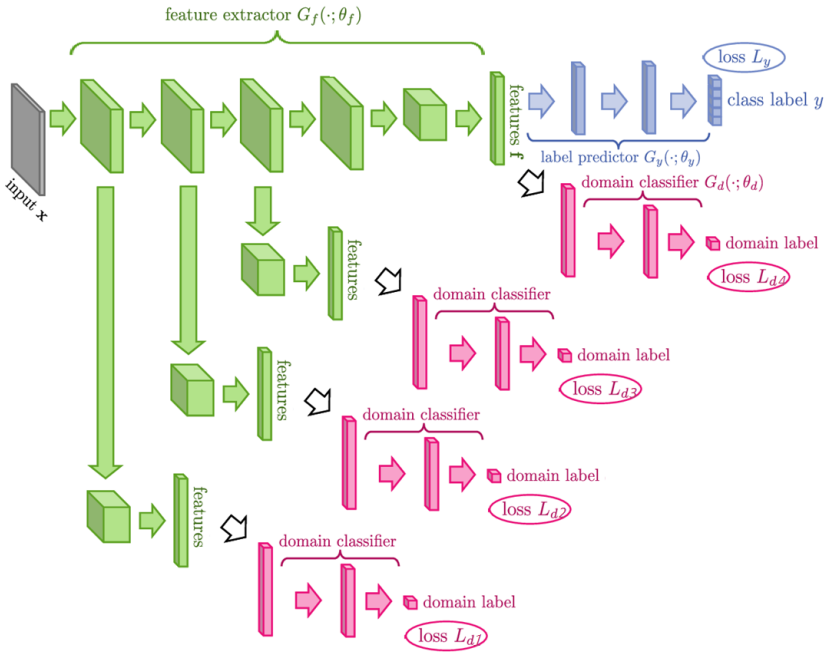


圖 3-8 多鑑別器的領域自適應模型架構

doi:10.6342/NTU202203562

3.2.5 超參數設定

人為給定的神經網路模型的參數稱為超參數，在前述所有影像分類模型中有幾個比較重要的超參數，其設定值如下所述。

1. Learning rate

每次更新權重的幅度，設太低會導致訓練時間過長，設太高會導致權重可能在局部最佳解附近震盪卻無法收斂。本研究統一設為 0.001。

2. Epoch

將訓練資料全部遍歷一遍稱為 1 epoch。本研究統一設為 100。

3. Batch size

每次訓練輸入的影像張數，此數字受限於電腦算力和模型結構，設太高會導致模型產生記憶體不足 (Out of Memory, OOM) 的錯誤。本研究統一設為 16。

4. Iterations (per epoch)

模型要跑完 1 epoch 需要的 Batch 數量，和 Batch size 及 Epoch 有關，三者只要設置其中兩個，故本研究不設定。

doi:10.6342/NTU202203562

3.3 實驗結果評判

本小節將介紹本研究在評估模型預測結果時會使用的評判工具。

3.3.1 混淆矩陣

在機器學習領域中，混淆矩陣為一常見的統計工具，其以預測結果與真實類別區分表格，藉由其呈現的數字可以迅速算出精準度 (Precision)、召回率 (Recall，又作 Sensitivity)、準確率 (Accuracy) 等統計量，各統計量算法如圖 3-9 所示。由於本研究是分類模型，在本研究中將以準確率作為模型效果的呈現，其物理意義是所有樣本中被正確分類的比例，也是大多領域自適應模型比較的標準。而在各類別效果的探討中，本研究將採用召回率作為探討重點，而非精準度，原因是目前在工業檢測領域，深度學習的檢測方式仍普遍被認為是一個初步篩檢瑕疵的輔助工具，其能節省大量人力並快速篩出大部分瑕疵，但後續始終會有第二部分人力檢測站來確定產品近乎零瑕疵的高品質。召回率的物理意義是在一類樣本中有多少樣本被正確的分出來為此類，若召回率高，即便精準度很低，廠商也能將多分入此類的樣品在後續挑出；然而若召回率低，即便精準度高，模型也沒有意義，因為還有表示還有很多的此類瑕疵未被挑出，因此對於廠商來說，深度學習瑕疵檢測模型要應用於工業瑕疵檢測其更被注重的是召回率。有時廠商也會關注過殺率 (Overkill)，其和召回率的和為 100%，故也能反映出瑕疵檢測模型的辨識能力。

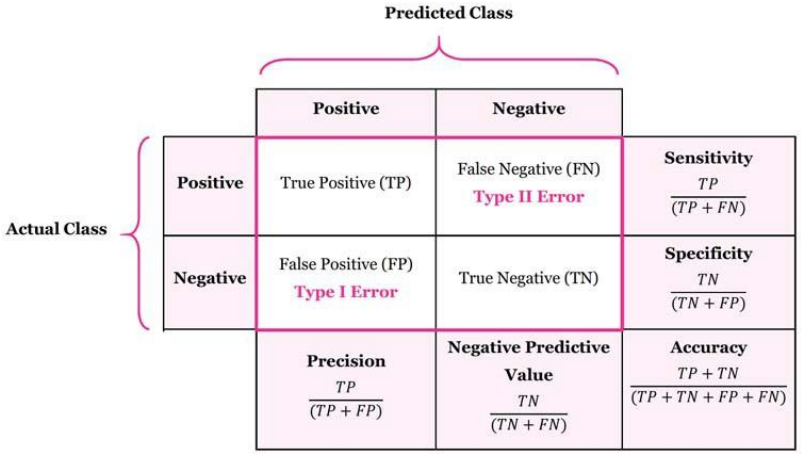


圖 3-9 混淆矩陣 [47]

doi:10.6342/NTU202203562

3.3.2 t-SNE (t-Distributed Stochastic Neighbor Embedding)

為了將領域自適應模型的適應情形可視化以方便觀察，常使用 t-SNE 進行影像降維，相較於主成分分析 (PCA) 降維，t-SNE 更能表達非線性的特徵關聯，以手寫辨識集 MNIST 為例，其呈現效果差別如圖 3-10 所示，左方為利用 t-SNE，右方為利用 PCA，可以發現利用 t-SNE 降維更能看出不同類別的資料分布。t-SNE 透過條件機率和高斯分布、t 分布來計算高維資料和低維資料的相似度，再以 KL 散度 (Kullback-Leibler divergence, KLD) 搭配梯度下降法求解，達到降維目的。在程式實踐上，本研究利用 sklearn 函式庫中的 TSNE 函式實踐，其需要設置困惑度 (Perplexity)，表達有效鄰近點數量，通常困惑度會設在 5 至 50 之間，若困惑度過低，只受少數鄰居資料影響，可能導致同分類的群被拆開；反之，若困惑度過高，不同類別間可能無法區分，本研究的困惑度設定為 10。

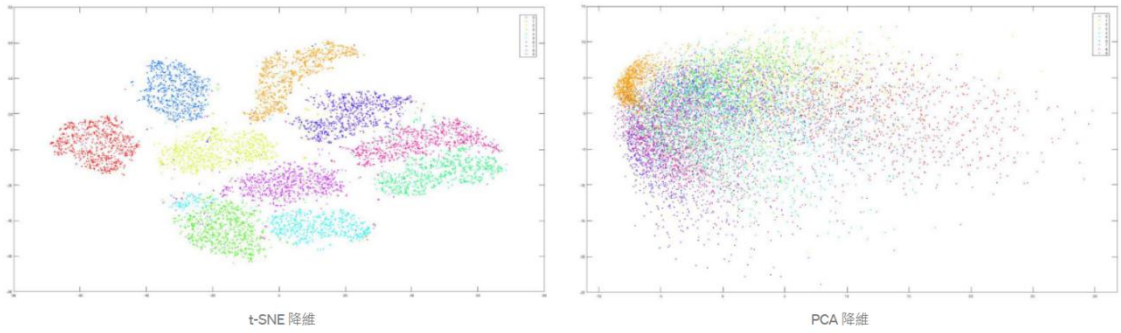


圖 3-10 MNIST 資料分布的可視化 [48]

doi:10.6342/NTU202203562

3.3.3 Grad-CAM (Gradient-weighted Class Activation Mapping)

相較於傳統的線性回歸等統計方法，深度學習演算法常因其難以解釋模型的分類判斷原因而被稱為黑盒子。因此，Zhou 等人於 2015 年提出了 CAM (Class Activation Mapping) [49]，其反向思考模型計算分類結果的數學方式，將最後的 softmax 前計算用的特徵權重 ω 和整張特徵圖的每個像素點相乘並疊加，由於 ω 越大表示對分類為該類的影響越大，反之則越小，如此做便能找到被模型關注最重的區域。然而，卷積神經網路需要在最後一層卷積層後有 GAP 層 (Global average pooling) 才會有特徵權重 ω，因此使用上有所侷限，故 Selvaraju 等人於 2016 年提出 Grad-CAM [50]，其利用反向傳播計算梯度，以求取神經元對最後一層卷積層的特徵圖的偏微分，透過計算的方式得到特徵權重 ω，如此便不需要有 GAP，也能產出關注程度的熱力圖，藉此判斷模型分類的依據，其範例效果如圖 3-11 所示。在本研究中使用 grad-cam 函示庫的 GradCAM 實現。

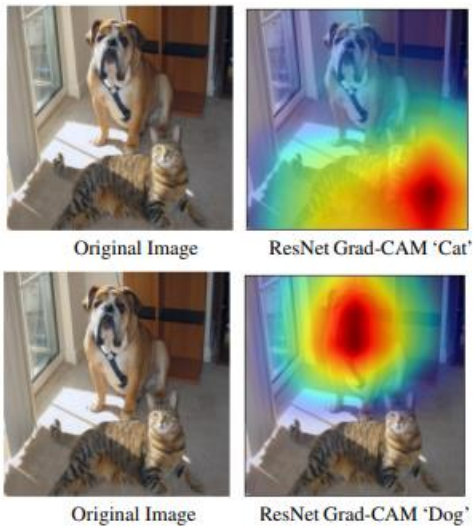


圖 3-11 以 Grad-CAM 產生之關注區域熱力圖 [50]

doi:10.6342/NTU202203562

小結

本章詳細按照實驗進行的流程，說明了本研究整理木皮、布匹、金屬表面瑕疵的資料集的步驟；也說明了本研究建立分類模型的方法，包括傳統的 ResNet50 分類模型與多個領域自適應模型，以及更改 DANN 的模型得到的多鑑別器領域自適應模型，用以分析適應特徵層對瑕疵辨識效果的影響；最後說明本實驗在第四章討論實驗結果時會用到的資料可視化工具，包括混淆矩陣、t-SNE、Grad-CAM，以及各自的程式實現方法。

doi:10.6342/NTU202203562

第四章　實驗結果與討論

本章將詳述本研究實驗得到的數據，並使用可視化方法檢視領域自適應模型判別瑕疵特徵的依據和領域適應情況，在本章最後會說明改變適應特徵層造成的辨識準確率影響。

4.1　ResNet50 分類結果

本實驗仿照業界瑕疵檢測時，直接用舊產線模型檢測新產線資料之情境，以木皮、布匹、金屬表面瑕疵各資料集內的兩領域資料互為源域與目標域，使用 ResNet50 模型進行瑕疵分類模型訓練，其檢測結果如表 4-1、表 4-2、表 4-3 所示。

表 4-1 ResNet50 分類木皮瑕疵結果

準確率(%)		模型訓練資料	
		oak_train	wal_train
測試資料	oak_train	100.00	58.70
	oak_test	94.92	49.15
	wal_train	58.44	97.40
	wal_test	56.76	94.59

表 4-2 ResNet50 分類布匹瑕疵結果

準確率(%)		模型訓練資料	
		black_train	white_train
測試資料	black_train	97.01	19.81
	black_test	94.23	21.79
	white_train	24.43	98.80
	white_test	23.37	92.39

doi:10.6342/NTU202203562

表 4-3 ResNet50 分類金屬瑕疵結果

準確率(%)		模型訓練資料	
		s200_train	s64_train
測試資料	s200_train	97.29	36.25
	s200_test	96.39	35.83
	s64_train	26.51	99.96
	s64_test	26.43	99.44

由前述三表發現，當測試資料和訓練模型時使用的資料來自同一領域時，模型皆能保持良好的辨識效果，準確率皆在 90%以上，也同時說明在第 3.1.3 小節提及的資料不平衡問題對本研究所使用的資料集的影響皆非常小可以忽略；但模型對於另一領域資料的辨識準確率便會大幅降低，例如：以 oak_train 訓練模型，對同領域的 oak_test 測試集能有 94.92%的辨識準確率，然而對不同領域的 wal_train、wal_test 資料集只能有不到 60%的辨識準確率，證實使用以往已有標籤的資料訓練的模型來檢測相似的新資料是不可行的。故對於一條新產線資料，廠商大多選擇重新標記並訓練模型，例如：同樣是測試 waln_test，如果是用 wal_train 訓練出的模型，準確率能達到 94.59%，但也造成人力成本和時間的消耗。

接著本研究也使用亮度和色調變換的資料增強方式進行實驗，增加訓練模型資料的變異性，以期望模型能對瑕疵特徵得到更好的泛化程度。其結果如表 4-4、表 4-5 所示，表中的 walnut 是結合 wal_train 和 wal_test、white 是結合 white_train 和 white_test。

doi:10.6342/NTU202203562

表 4-4 木皮之資料增強實驗結果

增強		亮度 0.5　色調 0.5
準確率(%)		模型訓練資料
		oak_train
測試資料	oak_train	100.00
	oak_test	98.02
	walnut	47.12

表 4-5 布匹之資料增強實驗結果

增強		亮度 0.5　色調 0.5
準確率(%)		模型訓練資料
		black_train
測試資料	black_train	94.34
	black_test	92.94
	white	23.25

由實驗結果可以推論，多使用影像顏色上的資料增強對木皮、布匹在目標域上的準確率並沒有提升效果，單純的亮度與色調變換並不能完全符合白橡木和胡桃木、黑布和白布間各種瑕疵的特徵差異。

4.2 領域自適應模型分類結果

上節之中證實了傳統的 CNN 方法無法檢測無標籤的目標域資料。本節將說明利用領域自適應技術的實驗結果。而後本研究會以可視化方法，嘗試討論模型分類瑕疵的合理性。

doi:10.6342/NTU202203562

4.2.1 分類準確率

本實驗以木皮、布匹、金屬表面瑕疵各資料集內的兩領域資料互為源域與目標域，使用第二章中提及的多種領域自適應模型進行瑕疵分類模型訓練，其檢測結果如表 4-6、表 4-7、表 4-8 所示。

表 4-6 領域自適應模型辨識木皮瑕疵結果

		準確率 (%)		
	model	白橡→胡桃	胡桃→白橡	avg
	ResNet50	56.76	49.15	52.96
差異型	DAN	83.39	74.58	78.98
	DEEPCORAL	83.39	72.88	78.14
	DSAN	86.49	74.58	80.53
對抗型	DANN	82.63	72.88	77.75
	DANN+E	91.89	77.97	84.93
	CDAN	82.63	82.20	82.42
	CDAN+E	83.78	83.05	83.42
	CDAN+G	83.78	83.05	83.42
	CDAN+D	86.49	89.83	88.16
	CDAN+GD	86.49	88.14	87.31
	DAAN	85.71	81.36	83.53
	GVB	78.38	71.19	74.78
	GVB+G	91.89	84.75	88.32
	GVB+D	81.08	69.49	75.29
	GVB+GD	91.22	84.75	87.98

doi:10.6342/NTU202203562

表 4-7 領域自適應模型辨識布匹瑕疵結果

		準確率 (%)		
	model	黑布→白布	白布→黑布	avg
	ResNet50	23.37	21.79	22.58
差異型	DAN	56.25	57.05	56.65
	DEEPCORAL	57.61	63.46	60.53
	DSAN	65.76	71.15	68.45
對抗型	DANN	63.07	64.10	63.59
	DANN+E	76.36	71.15	73.75
	CDAN	51.63	62.82	57.87
	CDAN+E	52.17	64.10	59.10
	CDAN+G	62.50	66.03	64.58
	CDAN+D	55.98	66.67	59.72
	CDAN+GD	51.63	63.46	62.35
	DAAN	67.39	73.07	70.23
	GVB	66.30	64.10	65.20
	GVB+G	66.30	59.62	62.96
	GVB+D	67.39	62.82	65.11
	GVB+GD	60.87	65.39	63.13

doi:10.6342/NTU202203562

表 4-8 領域自適應模型辨識金屬表面瑕疵結果

		準確率 (%)		
	model	s200→s64	s64→s200	avg
	ResNet50	26.43	35.83	31.13
差異型	DAN	79.70	79.72	79.71
	DEEPCORAL	86.14	85.83	85.99
	DSAN	92.41	96.39	94.40
對抗型	DANN	91.08	95.83	93.46
	DANN+E	93.94	97.22	95.58
	CDAN	88.61	95.28	91.95
	CDAN+E	88.93	96.11	92.52
	CDAN+G	92.91	96.94	94.93
	CDAN+D	91.16	97.5	94.33
	CDAN+GD	92.91	95.83	94.37
	DAAN	84.39	85.00	84.70
	GVB	89.73	90.83	90.28
	GVB+G	91.32	93.06	92.19
	GVB+D	90.21	94.17	92.19
	GVB+GD	93.07	95.28	94.18

三表中模型的順序是按差異型和對抗型網路各自的發表年份由上而下排列，綜觀木皮、布匹、金屬表面瑕疵資料集的實驗結果，本研究發現在瑕疵辨識率上，領域自適應模型相較於傳統 CNN 的 ResNet50 模型皆有長足的進步，顯示領域自適應模型確實能在目標域資料集沒有標籤的情況下，靠著有標籤的相似資料訓練出瑕疵分類模型，其中各行的最高準確率皆出現在對抗型模型的部分，在白橡測胡桃課題中，DANN+E 和 GVB+G 有最高的準確率；在胡桃測白橡課題中，CDAN+D 有最高的準確率；在黑布測白布課題中，DANN+E 有最高的準確率；在白布測黑布課題中，DAAN 有最高的準確率；在 s200 測 s64 課題中，DANN+E 有最高的準確率；在 s64 測 s200 課題中，CDAN+D 有最高的準確率。平均準確率來說，GVB+G 對木皮瑕疵的辨識表現最好，能有 88.32%的平均準確率；而 DANN+E 在布匹和金屬表面瑕疵檢測的表現最好，其平均準確率分別為 73.75%與 95.58%。為了統整出

doi:10.6342/NTU202203562

一套適合工業檢測的領域自適應模型訓練方法，綜觀上述三張表格的實驗結果，可以發現 DANN+E 在每張表格都至少有一行是最高值，換言之，DANN+E 能在兼顧辨識準確率的同時又有對不同瑕疵樣本的良好泛用性，所以當瑕疵檢測廠商需要建立一套標準作業程序來訓練領域自適應模型測試一種新的瑕疵品時，DANN+E 會是一個適合首先嘗試的模型選擇。

另外，在本研究做完對木皮和布匹瑕疵的辨識實驗後，發現模型的辨識率和模型被提出的年代順序沒有直接的關係。本研究認為一部份原因來自特徵擷取器的進步，例如在 DANN 被提出的年代根本還沒有 ResNet50，而 CDAN 模型提出的 Entropy Conditioning 也同樣能提升準確率，可以發現 DANN+E 和 CDAN+E 模型對瑕疵資料集的辨識準確率皆比原 DANN 和 CDAN 高；另一部份本研究認為和工業瑕疵影像本身的特性有關，工業瑕疵影像在拍攝時有固定的光學架構和拍攝角度，造成同一產線的影像瑕疵特徵會是比較單純、沒有這方面雜訊的。反之，學界論文在發表領域自適應新演算法時，最常使用的比較資料集 office 31 (圖 4-1) 或 office home，即便是同一領域資料內同一類別影像也有不同的拍攝角度，這些新的演算法可能在無形中對這些差異有明顯改善，對於工業瑕疵影像的改善幅度卻不明顯。為了驗證此一想法，本研究又針對 NEU-CLS 金屬表面瑕疵資料集來進行實驗，而後觀察表 4-8 的實驗結果，也得到類似木皮和布匹瑕疵的結果。

圖 4-1 office 31 範例圖 [51]

doi:10.6342/NTU202203562

4.2.2 可視化分析

本小節將以可視化方法進一步分析前一小節木皮、布匹、金屬表面瑕疵分類模型的實驗結果，將以各辨識課題之最高準確率模型為例。

1. 木皮瑕疵辨識結果

在白橡測胡桃課題中，DANN+E 模型有最高的辨識準確率，其混淆矩陣 (圖 4-2) 之對角線成深藍色，表示模型有極高的分類準確率，細看錯誤張數的話，在破裂與良品之間有一些判錯案例。

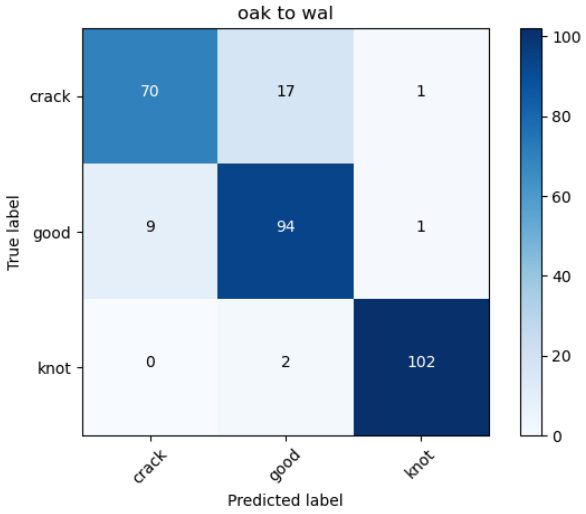


圖 4-2 白橡→胡桃之混淆矩陣

本研究觀察其判斷正確樣本之 Grad-CAM 圖 (圖 4-3)，其關注之瑕疵區域符合人類對瑕疵的判定區域。

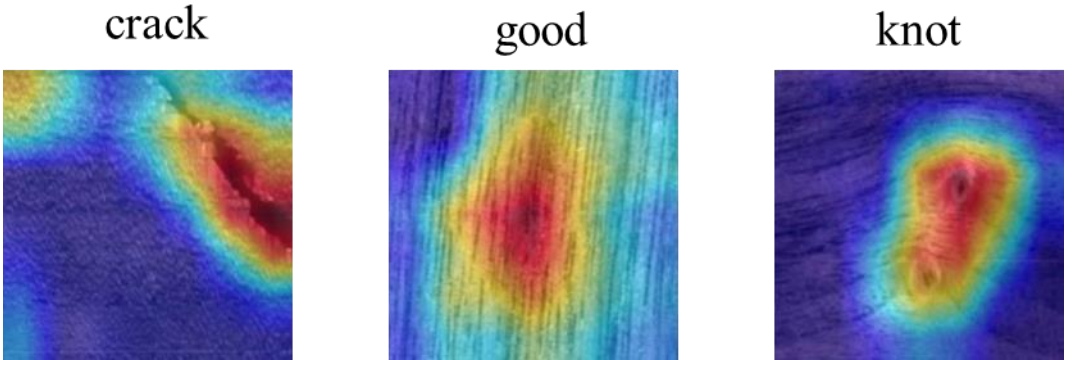


圖 4-3 白橡→胡桃之正確辨識案例 Grad-CAM 圖

doi:10.6342/NTU202203562

再觀察其判斷錯誤樣本之 Grad-CAM 圖 (圖 4-4)，對於一破裂瑕疵，其判定為良品，本研究認為是因為對於胡桃木而言，整體木皮成色較深，其紋理顏色故而和破裂之成色相近，不像白橡木上之破裂般突兀，造成了分類上的混淆。而觀其關注區域，主要偏影像右上方，該區域為良品，破裂主要在影像左下方，雖然模型的判斷邏輯是合理的，但是並未真正以瑕疵特徵進行分類，而造成最終錯誤的分類。

圖 4-4 白橡→胡桃之錯誤辨識案例 Grad-CAM 圖與原圖

而在胡桃測白橡課題中，CDAN+D 模型有最高的辨識準確率，其混淆矩陣 (圖 4-5) 之對角線成深藍色，表示模型有極高的分類準確率，細看錯誤張數的話，木珠瑕疵類別有一些判錯案例。

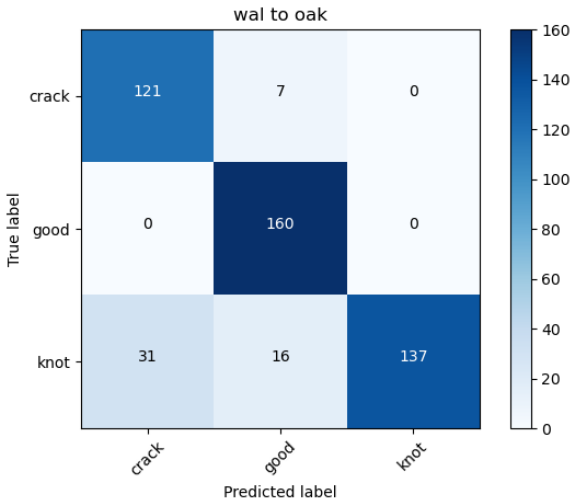


圖 4-5 胡桃→白橡之混淆矩陣

觀察其判斷正確樣本之 Grad-CAM 圖 (圖 4-6)，其關注之瑕疵區域符合人類對瑕疵的判定區域。

doi:10.6342/NTU202203562

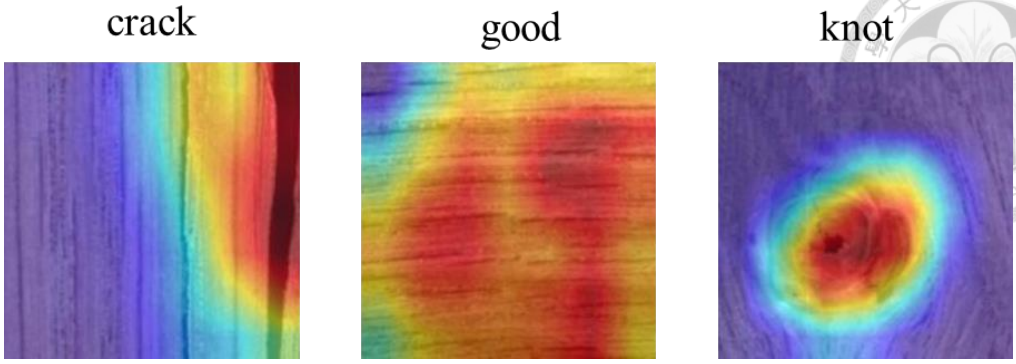


圖 4-6 胡桃→白橡之正確辨識案例 Grad-CAM 圖

再觀察其判斷錯誤樣本之 Grad-CAM 圖 (圖 4-7)，其會將少部分木珠辨識成破裂，本研究認為是因白橡木相較於胡桃木，質地本身較脆弱，在木珠之中心偶爾也會破裂，造成模型混淆；而模型也會將少部分木珠辨識成良品，由圖可以發現其判定成良品的根據區域正好不包括木珠而誤判。

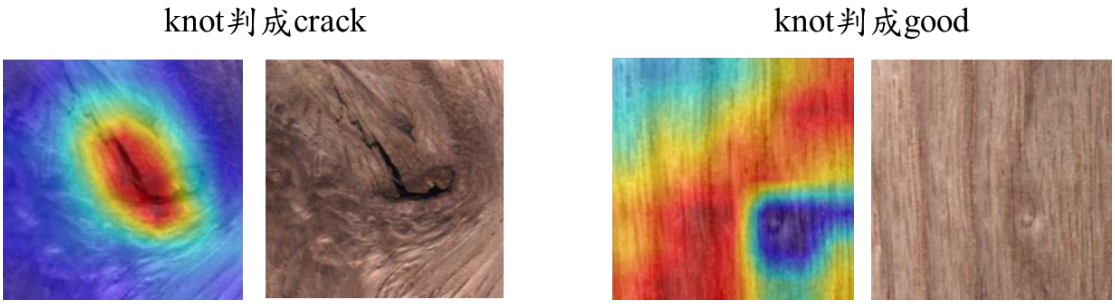


圖 4-7 胡桃→白橡之錯誤辨識案例 Grad-CAM 圖與原圖

本研究再以 t-SNE 影像降維方法觀察領域適應的情況 (圖 4-8)，以源域為胡桃木，目標域為白橡木時之 CDAN+D 模型為例，可以發現訓練前源域與目標域資料在適應特徵平面上的分佈情形不同，而訓練後無論源域還是目標域的資料都隱約被分為了三群，正好木皮的類別標籤也是三種，其中圖左上方和圖右上方兩群兩域的分佈範圍是比較接近的，領域適應效果良好，而圖下方一群雖兩域資料各自成群，但結合模型仍達到 89.83%的分類準確率，這樣的分佈對於模型來說還算是容易找到分類的邊界 (Boundary) 的。

doi:10.6342/NTU202203562

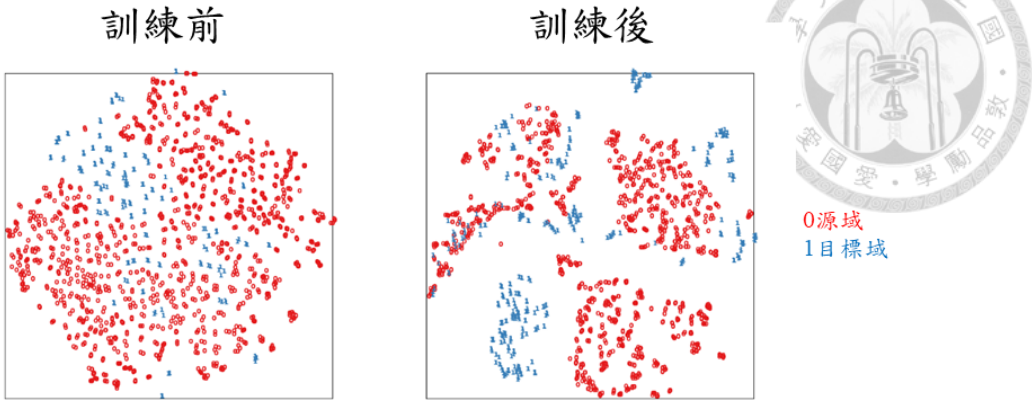


圖 4-8 胡桃→白橡之適應特徵層 t-SNE 圖

2. 布匹瑕疵辨識結果

在黑布測白布課題中，DANN+E 模型有最高的辨識準確率，其混淆矩陣 (圖 4-2) 除對角線外也有偏藍色的格子，表示模型的準確率有限，細看錯誤張數的話，有大量色汙瑕疵被判為摺痕。

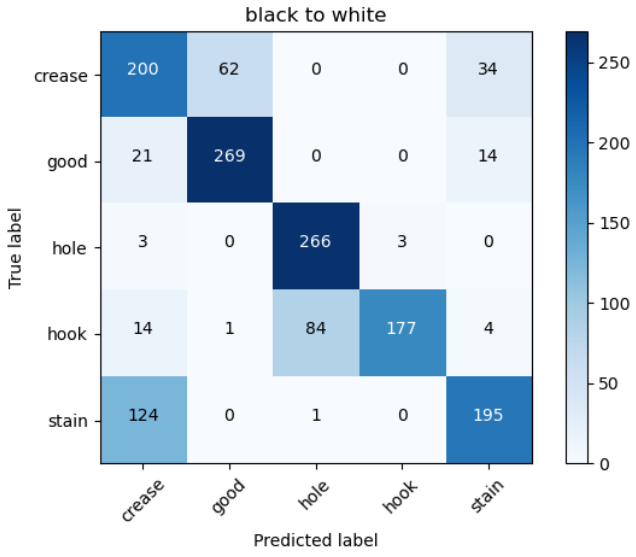


圖 4-9 黑布→白布之混淆矩陣

本研究觀察其判斷正確樣本之 Grad-CAM 圖 (圖 4-10)，其關注之瑕疵區域符合人類對瑕疵的判定區域。

doi:10.6342/NTU202203562

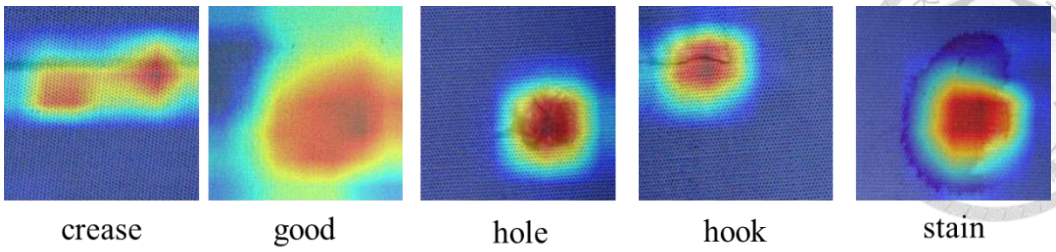


圖 4-10 黑布→白布之正確辨識案例 Grad-CAM 圖

再觀察其判斷錯誤樣本之 Grad-CAM 圖 (圖 4-11)，對於有大量色汙瑕疵被判為摺痕，本研究推測之原因將於觀察用白布測黑布之課題後探討。

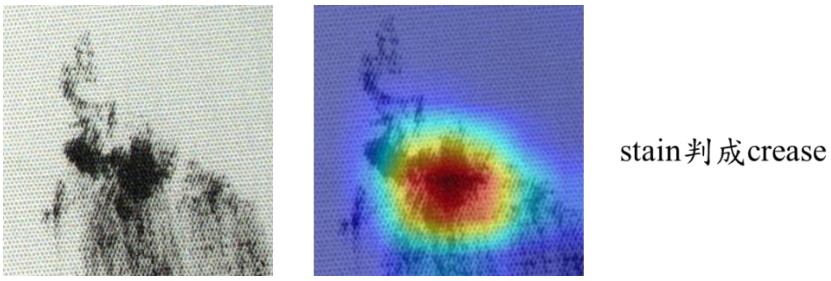


圖 4-11 黑布→白布之錯誤辨識案例 Grad-CAM 圖與原圖

而在白布測黑布課題中，DAAN 模型有最高的辨識準確率，其混淆矩陣 (圖 4-12) 之對角線並非全為深藍色，明顯可見對於色汙的判斷效果極差，只有兩張能正確辨識為色汙，其餘都被模型歸類於摺痕和良品。

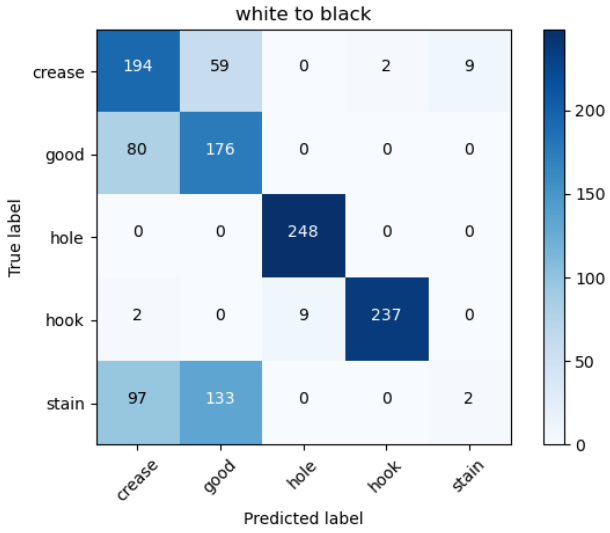


圖 4-12 白布→黑布之混淆矩陣

doi:10.6342/NTU202203562

觀察其判斷正確樣本之 Grad-CAM 圖 (圖 4-13)，其關注之瑕疵區域符合人類對瑕疵的判定區域，值得注意的是對於良品，此模型雖也會整張影像關注，但對影像邊緣的關注度更高。。

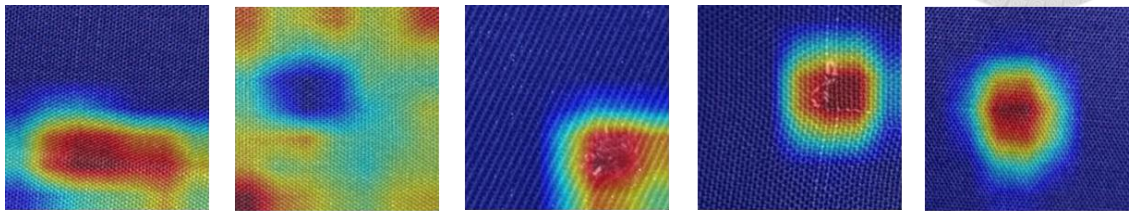


圖 4-13 白布→黑布之正確辨識案例 Grad-CAM 圖

再觀察其判斷錯誤樣本之 Grad-CAM 圖 (圖 4-14)，綜合黑布測白布課題時也是色汙分類效果不佳，研究認為模型對於色汙的判定標準應有一部份是基於頻率，影像的頻率指的是相鄰區域的灰度差，灰度差距很大的特徵稱為高頻特徵，反之則為低頻特徵，在白布資料集中，色汙屬於高頻特徵，摺痕特徵則相對低頻，然而在黑布資料集中，色汙偏向低頻特徵，故造成領域適應上的難度，最終導致白布測黑布課題時，模型把色汙瑕疵往低頻特徵的摺痕或良品類別去歸類。

圖 4-14 白布→黑布之錯誤辨識案例 Grad-CAM 圖、原圖與其他錯誤案例

本研究再以 t-SNE 影像降維方法觀察領域適應的情況 (圖 4-15)，以源域為白布，目標域為黑布時之 DAAN 模型為例，可以發現訓練前源域與目標域資料在適應特徵平面上的分佈情形不同，而訓練後目標域的資料隱約被分為了四群，對照混淆矩陣，少一群應是色汙資料混在了其他群裡，四群各自有靠近的源域資料族群，不過也單純是靠近而非重疊並均勻分布，能夠有不錯的分類準確率，更多是靠分類

doi:10.6342/NTU202203562

邊界的界定，如此出現類別特徵模稜兩可的影像時模型便容易誤判，整體而言，領域適應有效果但效果有限。

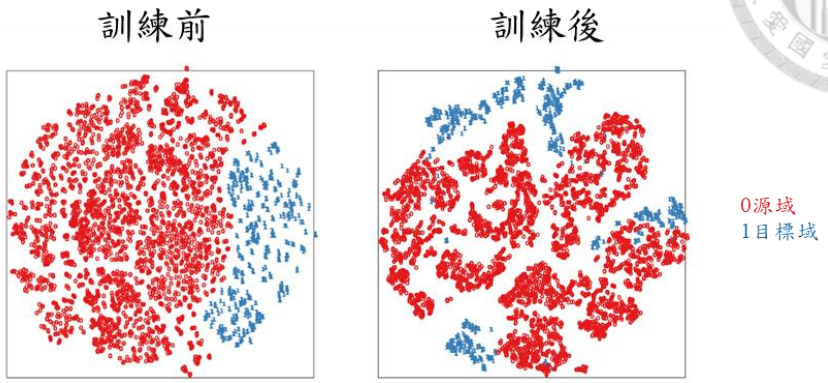


圖 4-15 白布→黑布之適應特徵層 t-SNE 圖

3. 金屬表面瑕疵辨識結果

在 s200 資料集測 s64 資料集課題中，DANN+E 模型有最高的辨識準確率，其混淆矩陣 (圖 4-16) 之對角線成深藍色，表示模型有極高的分類準確率，細看錯誤張數的話，有一些黑斑 (ccpa) 被辨識為孔蝕 (ccps)。

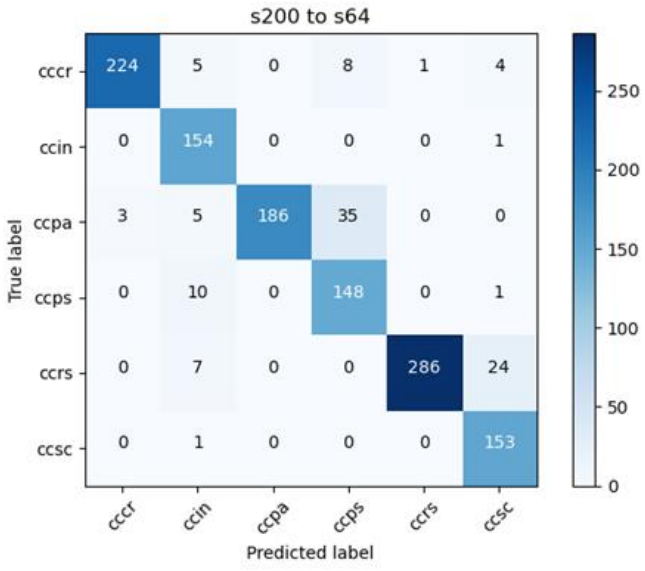


圖 4-16 s200→s64 之混淆矩陣

本研究觀察其判斷正確樣本之 Grad-CAM 圖 (圖 4-17)，其關注之瑕疵區域符合人類對瑕疵的判定區域。

doi:10.6342/NTU202203562

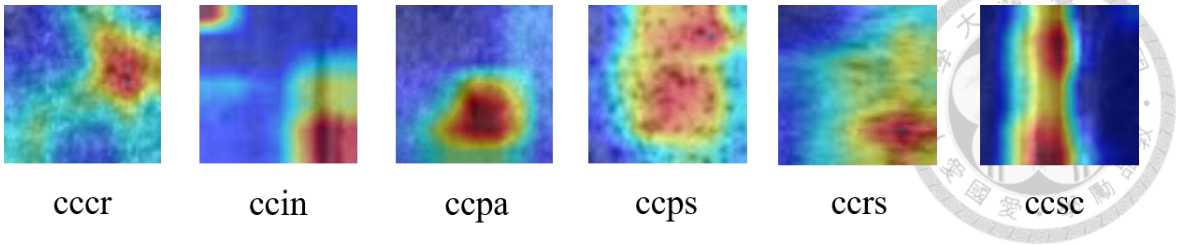


圖 4-17 s200→s64 之正確辨識案例 Grad-CAM 圖

再觀察其判斷錯誤樣本之 Grad-CAM 圖 (圖 4-18)，對於一黑斑瑕疵，其判定為孔蝕，本研究認為是因為受限於影像視野範圍，被裁切的黑斑確實會像孔蝕，其關注的區域確實有些範圍不大的深色特徵，而圖中左上角的黑斑並不在模型最終判定時的關注區域內，而造成最終錯誤的分類。

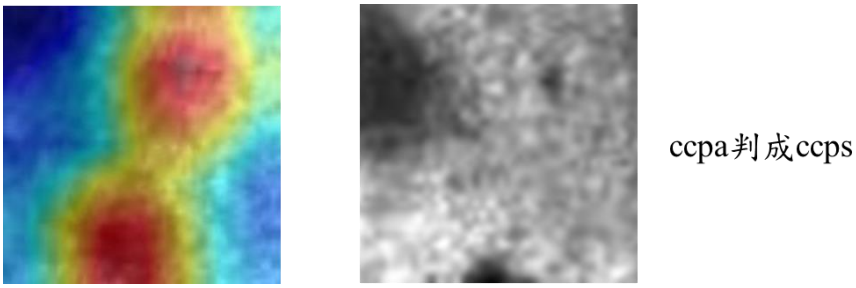


圖 4-18 s200→s64 之錯誤辨識案例 Grad-CAM 圖與原圖

而在 s64 資料集測 s200 資料集課題中，CDAN+D 模型有最高的辨識準確率，其混淆矩陣 (圖 4-19) 之對角線成深藍色，表示模型有極高的分類準確率，幾乎沒有判錯的影像。

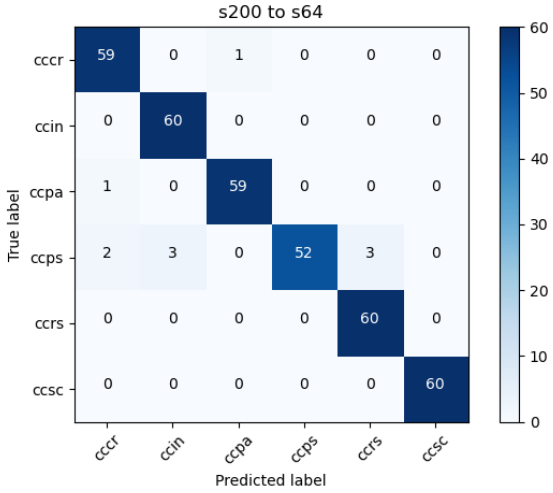


圖 4-19 s64→s200 之混淆矩陣

doi:10.6342/NTU202203562

觀察其判斷正確樣本之 Grad-CAM 圖 (圖 4-20)，其關注之瑕疵區域符合人類對瑕疵的判定區域。

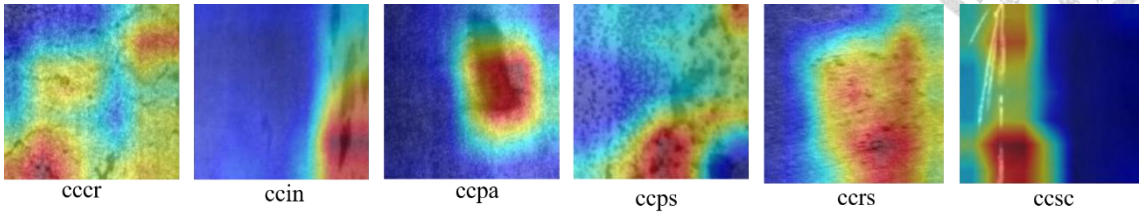


圖 4-20 s64→s200 之正確辨識案例 Grad-CAM 圖

本研究再以 t-SNE 影像降維方法觀察領域適應的情況 (圖 4-21)，以源域為 s200，目標域為 s64 時之 DANN+E 模型為例，可以發現訓練前源域與目標域資料在適應特徵平面上的分佈情形不同，而訓練後無論源域還是目標域的資料都大致被分為了六群，對應金屬表面瑕疵的類別標籤也是六種，其中圖右方和圖右下方兩群兩域的分佈範圍是比較接近的，領域適應效果良好，其他群雖兩域資料各自成群，但結合模型達 93.94%的分類準確率，這樣的分佈對於模型來說還算是容易找到分類的邊界的。

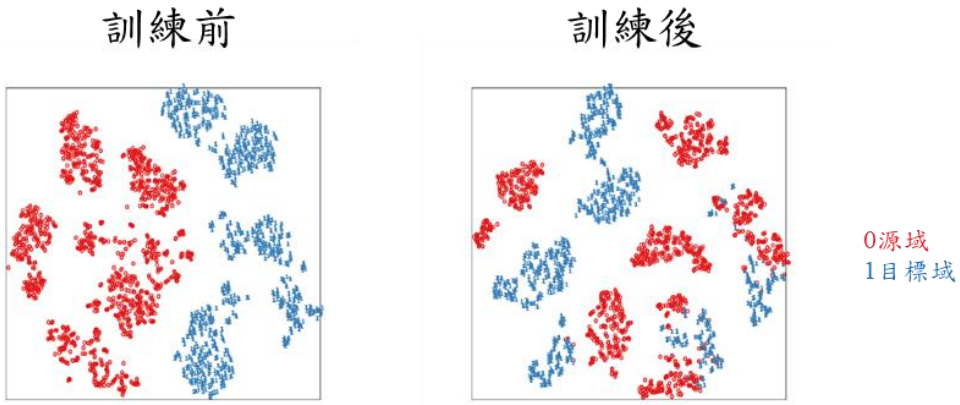


圖 4-21 s200→s64 之適應特徵層 t-SNE 圖

doi:10.6342/NTU202203562

4.3 改變鑑別器之適應特徵層實驗

在 4.2.1 節中本研究認為對抗型領域自適應是相對適合工業瑕疵影像的，並且 DANN+E 模型是效果良好的模型，在本節中實驗嘗試以多鑑別器領域自適應模型在 DANN+E 的模型基礎上，分析改變適應特徵層的影響，以進一步提升辨識準確率，其結果如表 4-9、4-10、4-11 所示。其中最靠近影像輸入端的 discriminator 稱為 1 號 discriminator，最靠近分類結果輸出端的 discriminator 稱為 4 號 discriminator，模型紀錄上，舉例：如果有使用 1、3、4 號 discriminator，模型會記成+134；在前面小節的實驗中，DANN+E 模型使用的是 4 號 discriminator，模型會記成+4。

表 4-9 多鑑別器領域自適應模型辨識木皮瑕疵結果

	準確率 (%)		
適應特徵層	白橡→胡桃	胡桃→白橡	avg
+1	75.34	68.01	71.67
+2	83.78	81.36	82.57
+3	72.97	74.58	73.77
+4	91.89	77.97	84.93
+12	78.72	77.33	78.02
+13	82.09	80.51	81.30
+14	84.13	84.75	84.44
+23	86.49	88.14	87.31
+24	91.89	86.44	89.17
+34	86.49	89.83	88.16
+123	78.38	77.75	78.06
+124	85.81	82.42	84.11
+134	80.41	81.99	81.2
+234	83.78	84.75	84.27
+1234	82.77	80.93	81.85
最高組合	91.89	89.83	90.86

doi:10.6342/NTU202203562

表 4-10 多鑑別器領域自適應模型辨識布匹瑕疵結果

	準確率 (%)		
適應特徵層	黑布→白布	白布→黑布	avg
+1	54.58	54.09	54.33
+2	45.11	71.15	58.13
+3	54.89	63.46	59.18
+4	76.36	71.15	73.76
+12	61.54	68.27	64.91
+13	64.14	63.78	63.96
+14	67.00	70.27	68.64
+23	54.89	64.74	59.82
+24	62.5	75.00	68.75
+34	63.59	74.36	68.97
+123	62.98	61.37	62.17
+124	67.42	65.94	66.68
+134	65.78	67.71	66.74
+234	73.37	65.39	69.38
+1234	66.46	66.26	66.36
最高組合	76.36	75.00	75.68

doi:10.6342/NTU202203562

表 4-11 多鑑別器領域自適應模型辨識金屬表面瑕疵結果

	準確率 (%)		
適應特徵層	s200→s64	s64→s200	avg
+1	85.99	86.67	86.33
+2	92.44	89.72	91.08
+3	86.07	90.00	88.03
+4	93.94	97.22	95.58
+12	84.39	87.78	86.09
+13	86.54	87.22	86.88
+14	90.29	89.22	89.75
+23	85.67	93.33	89.50
+24	95.22	93.89	94.56
+34	95.14	91.67	93.40
+123	88.22	88.06	88.14
+124	93.71	92.22	92.97
+134	89.73	91.11	90.42
+234	88.06	93.06	90.56
+1234	93.07	92.50	92.79
最高組合	95.22	97.22	96.22

不論是在 DAN [30] 或 FCN [21] 或更早的論文都有提到，卷積神經網路的淺層特徵多關注於整個資料集的共同特徵，而深層特徵則多關注於類別的特色特徵。通過實驗結果可以發現，訓練胡桃木測白橡木、訓練白布測黑布、訓練 200 尺寸金屬表面瑕疵測 64 尺寸金屬表面瑕疵時，其最高準確率模型分別為+34、+24、+24，相較於原+4 模型，辨識準確率是有提升的，但對於另外三個源域與目標域的實驗組合，+4 模型還是瑕疵辨識率最高的，所以無法歸納出使用特定哪幾層作領域適應能得到最好的辨識效果，不過，皆是在有使用 4 號 discriminator 的情況，換言之，對木皮、布匹、金屬表面瑕疵資料集使用 DANN+E 分類模型訓練，特徵適應層是否包含特徵擷取器 ResNet50 的第四個大 layer 的輸出特徵是重要的。

總的來說，本研究認為同以往論文一樣對 ResNet50 的第四個大 layer 的輸出特徵做領域適應是合理的，但當檢測工業瑕疵時，想要進一步提升準確率，嘗試在

doi:10.6342/NTU202203562

前面的某一層 layer 也進行領域適應是有可能提升辨識準確率的，也只要嘗試含有 4 號 discriminator 的模型就好。

小結

本章詳細討論了運用領域自適應技術辨識木皮、布匹、金屬表面瑕疵的效果，證實相較於用源域資料訓練出的傳統 CNN 模型直接檢測目標域資料，運用領域自適應技術訓練出的模型辨識效果更好，其中研究認為 DANN+E 是個通用且能在瑕疵資料集上取得良好辨識效果的模型，如欲進一步提升準確率，可再經由改變適應特徵層的組合搭配訓練準確率更高的模型，提升後對木皮瑕疵資料的平均辨識準確率可達 90.86%，對布匹瑕疵資料的平均辨識準確率可達 75.68%，對金屬表面瑕疵資料的平均辨識準確率可達 96.22%。

doi:10.6342/NTU202203562

第五章 結論與未來展望

5.1 結論

本研究以不同領域自適應模型對木皮、布匹、金屬表面瑕疵之影像進行瑕疵辨識，分析模型判斷瑕疵的特徵依據與適應特徵層對模型準確率的影響，以下就本論文的貢獻提出總結：

1. 確認領域自適應技術應用於木皮、布匹、金屬表面瑕疵 AOI 檢測的可行性，證實其適用於檢測具有紋路的產品。領域自適應確實能一定程度克服相似瑕疵影像辨識課題卻需重新訓練的問題，跳過標記新資料的步驟，節省人力與時間成本。
2. 工業瑕疵影像與學界分析用影像於特性上有所不同，工業瑕疵影像在拍攝角度上較為單一，故最新的領域自適應模型不一定會得到較好的辨識結果。
3. 本研究提出領域自適應技術用於 AOI 瑕疵影像分類的合適流程，即以已有標籤的相似資料和尚無標籤的待測資料一同訓練 DANN 領域自適應模型，並輔以 Entropy Conditioning 運算，再嘗試調整適應特徵層的組合能得到對待測資料良好的辨識準確率。

綜合研究結果，本研究認為領域自適應技術用於工業瑕疵檢測的方式還有以下兩種：

1. 在事前評估階段，若花費時間人力標記訓練不符成本，能先利用此技術快速訓練瑕疵分類模型。
2. 如若準確率是客戶最大的考量，需要使用重新標記訓練新資料的方式得到高辨識準確率，也能透過領域自適應先產生一定準確率之預測標籤，再由人工稍微更改，節省人工標記時間，模型的權重也能用作新模型的預訓練模型，節省訓練與收斂的速度。

doi:10.6342/NTU202203562

5.2 未來展望

以下基於本研究不足之處所提出的後續研究方向：

1. 待物件偵測式領域自適應技術成熟後導入瑕疵檢測。對於工業產線來說，能知道瑕疵的位置與類別更有利於後續自動化產線的設定，例如以機械手臂將瑕疵部位裁切挑出，因此物件偵測相較於分類是更適合工業瑕疵檢測的。然而目前將領域自適應技術用於物件偵測的準確率不高 [52]，其 mAP (Mean Average Precision) 很難超過 50%。

2. 本研究中的各種領域自適應演算法皆是建立在源域資料和目標域資料的瑕疵類別完全相同的情況下使用，然而工業上可能會遇到新產線雖然也是生產相似產品，但瑕疵類別卻和舊產線略有不同的情況，換句話說，源域資料和目標域資料的分類類別略有不同，學界將此類課題歸納為 Open Set Domain Adaptation [53]，可能的解決方法如 2019 年 You 等人提出的 UDA(Universal Domain Adaptation) [54]。本研究雖無法處理此一問題，但提出一可能可行之方法作為後續的研究方向，即以布匹資料為例，若源域之黑布資料有摺痕、破洞、勾痕、良品四類，目標域之白布資料有摺痕、破洞、色汙、良品四類，可將源域資料之良品類別和其餘瑕疵類別之一分別組合訓練領域自適應模型，訓練出三個分別針對摺痕、破洞、勾痕瑕疵的二元分類器，再藉由計算三模型對目標域資料判斷類別時的肯定程度等方式整合三模型的判斷資訊，對目標域資料做出最終的瑕疵判斷類別，並將所有不存在於源域資料的類別如色汙歸至一類別名為 unknown。

3. 導入偽標籤 (Pseudo-Labelling) 概念進一步提升準確率。2013 年 Lee 提出偽標籤 [55]，其概念是用已有模型對新的無標籤資料產生不一定正確的標籤，

doi:10.6342/NTU202203562

再和已有正確標籤之資料一起訓練模型，形成半監督訓練，在後世的論文中也證實此方法有助提升神經網路模型準確率。

doi:10.6342/NTU202203562

參考文獻

[1] C. J. Lu and D. M. Tsai, "Automatic defect inspection for LCDs using singular value decomposition," in *The International Journal of Advanced Manufacturing Technology*, vol. 25, pp. 53-61, 2005.

[2] J. Zhao, Q. J. Kong, X. Zhao, J. P. Liu and Y. C. Liu, "A method for detection and classification of glass defects in low resolution images," in *2011 Sixth International Conference on Image and Graphics (ICIG)*, pp. 642-647, 2011.

[3] MARKETSANDMARKETS BLOG, "What are their major strategies to strengthen Automated Optical Inspection (AOI) System market presence?" June 2020. [Online] https://mnmblog.org/what-are-their-major-strategies-to-strengthen-automated-optical-inspection-aoi-system-market-presence.html [Accessed 07 03 2022]

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, November 1998.

[5] Y. H. Won, H. Joo, and J. S. Kim, "Classification of Defects in the Polarizer of Display Panels using the Convolution Neural Network (CNN)." in *International Journal of Computing, Communications and Instrumentation Engineering (IJCCIE)*, Vol. 4, No. 1, 2017.

[6] Wikipedia, "Artificial neural network," February 2013. [Online] https://en.wikipedia.org/wiki/Artificial_neural_network [Accessed 07 03 2022]

[7] 程式人生, "更好的理解分析深度卷積神經網路," January 2019. [Online] https://www.796t.com/content/1547645949.html [Accessed 07 03 2022]

[8] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, 313(5786), 504-507, 2006

doi:10.6342/NTU202203562

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, 2012.

[10] Sagar Sharma, "Activation Functions in Neural Networks," September 2017. [Online] https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6. [Accessed 07 03 2022]

[11] Shivani Kolungade, "Object Detection, Image Classification and Semantic Segmentation using AWS Sagemaker," August 2020. [Online] https://medium.com/@kolungade.s/object-detection-image-classification-and-semantic-segmentation-using-aws-sagemaker-e1f768c8f57d [Accessed 07 03 2022]

[12] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[13] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going Deeper with Convolutions," *arXiv preprint arXiv:1409.4842*, 2014.

[14] S. Ioffe, and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, pp. 448-456, June 2015.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016

[16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779-788, 2016.

doi:10.6342/NTU202203562

[17] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587, 2013.

[18] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440-1448, 2015.

[19] 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*, 39(6), 1137-1149, 2016.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740-755, September 2014.

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431-3440, 2015.

[22] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234-241, October 2015.

[23] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask R-CNN," in *Proceedings of the IEEE international conference on computer vision*, pp. 2961-2969, 2017.

[24] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset shift in machine learning*, The MIT Press, 2009.

doi:10.6342/NTU202203562

[25] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He, "A Comprehensive Survey on Transfer Learning," *arXiv preprint arXiv:1911.02685*, 2019.

[26] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

[27] Hao-Ping Lee, Ta-Wei Tang, Wan-Julin, Hakiem Hsu, and Kuan-Ming Li, "A CNN-based Defect Inspection for Smartphone Cover glass," *International Journal of Mechanical and Production Engineering (IJMPE)*, pp. 42-45, Volume-9, Issue-1, 2021

[28] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell, "Deep Domain Confusion: Maximizing for Domain Invariance," *arXiv preprint arXiv:1412.3474*, 2014.

[29] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky, "Domain-Adversarial Training of Neural Networks," *arXiv preprint arXiv:1505.07818*, 2015.

[30] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan, "Learning Transferable Features with Deep Adaptation Networks," *arXiv preprint arXiv:1502.02791*, 2015.

[31] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola, "A kernel two-sample test." *The Journal of Machine Learning Research,* 13(1), pp. 723-773, 2012

[32] Baochen Sun, Jiashi Feng, and Kate Saenko, "Return of Frustratingly Easy Domain Adaptation," *arXiv preprint arXiv:1511.05547*, 2015.

doi:10.6342/NTU202203562

[33] Baochen Sun and Kate Saenko, "Deep CORAL: Correlation Alignment for Deep Domain Adaptation," *arXiv preprint arXiv:1607.01719*, 2016.

[34] Yongchun Zhu, Fuzhen Zhuang, Jindong Wang, Guolin Ke, Jingwu Chen, Jiang Bian, Hui Xiong, and Qing He, "Deep Subdomain Adaptation Network for Image Classification," *arXiv preprint arXiv:2106.09388*, 2021.

[35] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative Adversarial Networks," *arXiv preprint arXiv:1406.2661*, 2014.

[36] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan, "Conditional Adversarial Domain Adaptation," *arXiv preprint arXiv:1705.10667*, 2017.

[37] Chaohui Yu, Jindong Wang, Yiqiang Chen, and Meiyu Huang, "Transfer Learning with Dynamic Adversarial Adaptation Network," *arXiv preprint arXiv:1909.08184*, 2019.

[38] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Chi Su, Qingming Huang, and Qi Tian, "Gradually Vanishing Bridge for Adversarial Domain Adaptation," *arXiv preprint arXiv:2003.13183*, 2020.

[39] Yaroslav Ganin and Victor Lempitsky, "Unsupervised domain adaptation by backpropagation," *arXiv preprint arXiv:1409.7495*, 2014.

[40] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," *arXiv preprint arXiv:1711.03213*, 2017.

[41] Yu He, Kechen Song, Qinggang Meng, Yunhui Yan, "An End-to-end Steel Surface Defect Detection Approach via Fusing Multiple Hierarchical Features," *IEEE Transactions on Instrumentation and Measuremente*, 2020, 69(4), 1493-1504.

doi:10.6342/NTU202203562

[42] Teledyne FLIR, "Blackfly GigE," [Online] https://www.flir.com/products/blackfly-gige/?model=BFLY-PGE-50S5C-C&vertical=machine+vision&segment=iis [Accessed 07 03 2022]

[43] Google Developer, " Imbalanced Data," [Online] https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data [Accessed 07 03 2022]

[44] Nvidia, "GEFORCE® GTX 1080 Ti." [Online] https://www.nvidia.com/zh-tw/geforce/products/10series/geforce-gtx-1080-ti/ [Accessed 07 03 2022]

[45] Pytorch, "RESNET," [Online] https://pytorch.org/hub/pytorch_vision_resnet/ [Accessed 07 03 2022]

[46] Jindong Wang, "Transferlearning Project," Git code. [Online] https://github.com/jindongwang/transferlearning [Accessed 07 03 2022]

[47] M. SIRSAT - Data Science and Machine Learning, "What is Confusion Matrix and Advanced Classification Metrics?," April 2019. [Online] https://manishasirsat.blogspot.com/2019/04/confusion-matrix.html. [Accessed 07 03 2022]

[48] 愷開，"淺談降維方法中的 PCA 與 t-SNE," July 2017. [Online] https://medium.com/d-d-mag/%E6%B7%BA%E8%AB%87%E5%85%A9%E7%A8%AE%E9%99%8D%E7%B6%AD%E6%96%B9%E6%B3%95-pca-%E8%88%87-t-sne-d4254916925b [Accessed 07 03 2022]

[49] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba, "Learning Deep Features for Discriminative Localization," *arXiv preprint arXiv:1512.04150*, 2015.

doi:10.6342/NTU202203562

[50] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," *arXiv preprint arXiv:1610.02391*, 2016.

[51] Herath, Samitha, Mehrtash Harandi, and Fatih Porikli, "Learning an invariant hilbert space for domain adaptation." *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[52] Poojan Oza, Vishwanath A. Sindagi, Vibashan VS, and Vishal M. Patel, "Unsupervised Domain Adaptation of Object Detectors: A Survey," *arXiv preprint arXiv:2105.13502*, 2021.

[53] P. P. Busto and J. Gall, "Open Set Domain Adaptation," *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 754-763, doi: 10.1109/ICCV.2017.88.

[54] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal Domain Adaptation," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2715-2724, doi: 10.1109/CVPR.2019.00283.

[55] Dong-Hyun LEE, "Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks." *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 2013

doi:10.6342/NTU202203562