

國立臺灣大學電機資訊學院電機工程學系



碩士論文

Department of Electrical Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

基於資料擴增與深度學習的改善磁控膠囊內視鏡辨識

腸道之研究

Study on the Improvement of Intestinal Identification by

Magnetic Controlled Capsule Endoscope Based on Data

Augmentation and Deep Learning

黃威銘

Wei-Ming Huang

指導教授：劉志文 博士

Advisor: Chih-Wen Liu, Ph.D.

中華民國 109 年 7 月

July, 2020

國立臺灣大學碩士學位論文
口試委員會審定書



基於資料擴增與深度學習的改善磁控膠囊內視鏡辨識
腸道之研究

Study on the Improvement of Intestinal Identification by
Magnetic Controlled Capsule Endoscope Based on Data
Augmentation and Deep Learning

本論文係黃威銘君（學號 R07921022）在國立臺灣大學電機工程
學系完成之碩士學位論文，於民國 109 年 07 月 15 日承下列考試委員
審查通過及口試及格，特此證明。

口試委員：

劉志文

（簽名）

（指導教授）

連志時

栗峰茹

系主任

吳忠中

（簽名）



致謝

光陰似箭，如今已經來到碩班二年級，快要畢業的時刻，回想當時推開實驗室大門的忐忑不安，到現今同學畢業之後的分道揚鑣，讓人不禁難過且不捨。在這兩年之中，實驗室給了我非常多的幫助。首先我要感謝我的指導教授 劉志文教授，讓我進入這間充滿人情味的實驗室，也帶領我踏入一個我從未踏入過的領域。在老師的指導下我依照自己的步調找尋到研究的方向，從完成的研究中收穫到一些感想及知識，相信在未来的職場中能夠依靠這些經驗不會手忙腳亂。

在這邊，我要感謝膠囊組強大的三位學長，浩恩、聲揚和謝忱。感謝浩恩學長每次和我交換很多的實驗想法，以及給予我們一些比賽經驗及參與展覽，還感謝他時常請我們吃點心；感謝聲揚學長給予我很多研究的方向，並告訴我可以如何呈現出研究成果，以及幫助我修改實驗需要的程式；感謝謝忱學長帶領我進入到影像處理的領域，讓我熟悉影像處理所需要的知識，也提供我一些可以繼續發展的研究方向。感謝有教授和三位學長的帶領，才能夠完成這篇論文。

除了教授和三位學長，我也要感謝跟我同一批進來的同學和隔壁實驗室的老師。感謝跟我同組的家澧常常鼓勵我們加油不要放棄，我和他常分享彼此的研究近況和進度；感謝馬達組的呈軒，和他說鬥嘴讓我的研究過程不那麼枯燥乏味；感謝電網組的仕凡、鄧維和伊斯拉，跟他們聊天可以減緩一些實驗中的煩悶；感謝隔壁實驗室的廷軒和加峻，時常來我們實驗室串門子，也告訴我很多人生經驗談。

除了學長和同學，最後還要感謝實驗室和隔壁的其他學長、學弟妹，感謝學長群：育儒、棹沅、育琦、宇航、歐靖、逸松、何羅伊、翊瑋、郁翔，感謝學弟、妹群：君翰、韋帆、仁宏、孟芸、鳳翎、張廖俊魁、昱全、祐寧。有他們在使得整個實驗室不死氣沉沉，顯得十分熱鬧。最後感謝我的父母，支持我繼續讀台灣大學電機系碩士，讓我能夠在學習階段中有一個完整的結尾。



摘要

近年來，在科技的不斷進步之下，電腦視覺將直接應用在各行各業中。於此同時，由於高效能圖形處理器 (Graphics Processing Unit, GPU) 的效能大幅進步，也使得深度學習的相關研究相當熱門。憑藉良好的訓練模型與充足的訓練數據，深度學習將電腦視覺之研究推進至一個新紀元。

在本論文中，我們將深度學習應用於辨識腸腔並取得其位置，達到磁控大腸膠囊內視鏡自動導航之方法。使用了最新的即時目標偵測深度學習模型—YOLO (You Only Look Once)。我們使用 KVASIR 之腸道圖像資料集來驗證 YOLO 模型之腸腔偵測效果，但是 KVASIR 之腸道圖像的數量仍然很少。

由於腸道圖像的不足，本文目的是使用資料擴增技術使 KVASIR 之腸道圖像增加，以便將其用於 YOLO 模型訓練以提高檢測效果。我們使用兩種擴增方式：傳統之資料擴增與生成對抗網路 (Generative Adversarial Networks, GAN)。傳統之資料擴增使用五種擴增方式：圖像平移、圖像旋轉、圖像縮放、圖像加入高斯雜訊與圖像加入動態模糊；而生成對抗網路使用 Cycle-GAN 將模擬腸道圖像轉換成 KVASIR 之腸道圖像。

關鍵字： 膠囊內視鏡、電腦視覺、資料擴增、生成對抗網路、人工智慧、深度學習、腸腔偵測

ABSTRACT



In recent years, with the continuous progress of science and technology, computer vision will be directly applied in all walks of life. At the same time, due to the efficiency of highly efficient graphics processing unit (GPU) greatly improved, the related research of deep learning is quite popular. With good training models and sufficient training data, deep learning advances the research of computer vision into a new era.

In this paper, we apply deep learning technology to identify the lumen and obtain its position, so as to achieve the novel navigation for magnetic field control endoscope. We use the state-of-the-art, real-time object detection deep learning model - YOLO (You Only Look Once). KVASIR dataset was used to evaluate the performance of lumen detection, but images of the KVASIR dataset are still few.

Due to insufficient intestinal images, this thesis aims at using data augmentation technology to increase the intestinal images of KVASIR dataset, so as to apply it to YOLO model training to improve the detection effect. We used two data augmentation methods: traditional data augmentation and Generative Adversarial Networks (GAN). Traditional data augmentation methods include image translation, image rotation, image scaling, image adding Gaussian noise and image adding motion blur; Cycle-GAN was used to transform the simulated intestinal images into the real intestinal images of KVASIR dataset.

Keywords: Capsule endoscope, Computer vision, Data augmentation, Generative Adversarial Network, Artificial neural networks, Deep learning, Lumen detection.

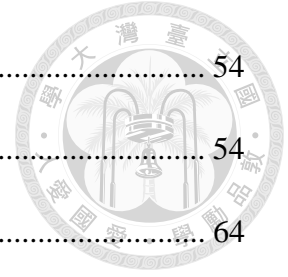
目錄



口試委員會審定書	i
致謝	ii
摘要	iii
ABSTRACT	iv
目錄	v
圖目錄	viii
表目錄	xii
第一章 緒論	1
1.1 研究背景	1
1.2 研究動機及研究目的	3
1.3 文獻回顧	4
1.4 章節摘要	7
第二章 磁牽引平台與磁控膠囊內視鏡之介紹	8
2.1 MFN 之概況與演進	8
2.1.1 手持式 MFN	8
2.1.2 磁牽引平台 (MFN Platform)	10
2.1.3 第一代 MFN Platform 與第二代 MFN Platform 之比較	12
2.2 磁控膠囊內視鏡之概況	13
2.3 動物試驗之成果	15
2.3.1 大腸鏡訓練模型	15
2.3.2 豬隻活體試驗	17



2.3.3	豬隻活體試驗之結果	19
2.3.4	結論與未來工作	22
第三章	資料擴增	24
3.1	幾何變換類	24
3.2	顏色變換類	28
第四章	生成對抗網路	31
4.1	生成對抗網路	31
4.2	Cycle-GAN	33
4.3	損失函數	35
第五章	腸腔辨識	39
5.1	YOLO	39
5.2	YOLO 演算法	41
5.2.1	網格單元 (Grid Cell)	41
5.2.2	YOLO 神經網路架構	43
5.2.3	非極大值抑制 (Non-max suppression, NMS)	43
5.2.4	損失函數 (Loss Function)	44
5.3	YOLOv3	47
第六章	實驗架構與成果討論	49
6.1	硬體與架構	49
6.2	資料集	49
6.3	資料處理	49
6.4	性能指標	50



6.5	實驗結果	54
6.5.1	資料擴增	54
6.5.2	Cycle-GAN.....	64
6.5.3	YOLOv3 腸腔辨識	69
第七章	結論與未來工作	72
7.1	結論	72
7.2	未來工作	73
參考文獻	74

圖目錄



圖 1.1	108 年十大主要癌症排名[1].....	4
圖 1.2	Medtronic (原 Imaging) PillCam[2]	2
圖 1.3	資料擴增與未經資料擴增之比較[3].....	4
圖 1.4	基於底圖與欲風格轉化之圖像所生成之合成圖像[4].....	5
圖 1.5	CNN 生成皮膚黑色素瘤之 32 種圖像[4].....	5
圖 1.6	MRI 經過 Cycle-GAN 轉化成 CT 之結果[5].....	6
圖 2.1	本團隊研製之手持式 MFN[6]	8
圖 2.2	本團隊研製之有線式磁控膠囊內視鏡 (a)示意圖 (b)實際圖[6].....	9
圖 2.3	手持式 MFN 操作示意圖[6]	9
圖 2.4	第一代 MFN Platform 之三視圖[7]	10
圖 2.5	第一代 MFN Platform 之實際圖[7]	11
圖 2.6	第二代 MFN Platform 之實際圖	11
圖 2.7	有線式磁控膠囊內視鏡之成品.....	13
圖 2.8	磁控大腸膠囊內視鏡.....	13
圖 2.9	微型腸鏡訊號處理主機.....	14
圖 2.10	M40 大腸鏡訓練模型[8]	15
圖 2.11	操作六種不同等級之大腸模型.....	16
圖 2.12	訓練模型之六種不同難度等級之操作時間長條圖.....	16
圖 2.13	豬隻活體試驗之過程圖.....	17
圖 2.14	豬隻活體試驗之拍攝位置.....	18
圖 2.15	磁控大腸膠囊內視鏡之光源透出示意圖.....	19



圖 2.16	MFN Platform 之機械手臂碰到豬隻腹部示意圖	19
圖 2.17	磁控大腸膠囊內視鏡影像擷取.....	20
圖 2.18	豬隻解剖示意圖.....	20
圖 2.19	磁控大腸膠囊內視鏡進入腸道實際深度之示意圖.....	21
圖 2.20	腸道傷痕之示意圖.....	21
圖 2.21	肉眼病理診斷報告.....	22
圖 3.1	圖像平移.....	25
圖 3.2	圖像縮放.....	26
圖 3.3	圖像旋轉.....	27
圖 3.4	任一點圍繞原點旋轉之示意圖.....	27
圖 3.5	高斯分佈.....	29
圖 3.6	加入高斯雜訊之圖像.....	29
圖 3.7	加入動態模糊之圖像.....	30
圖 4.1	生成對抗網路[10].....	31
圖 4.2	GAN 優化之訓練過程[10]	33
圖 4.3	資料成對之示意圖.....	33
圖 4.4	Cycle-GAN	34
圖 4.5	資料成對與非資料成對之示意圖[11].....	35
圖 4.6	循環一致性之示意圖[11].....	37
圖 4.7	Cycle-GAN 加入 $\mathcal{L}_{identity}$ 差異之示意圖[11].....	38
圖 5.1	磁控大腸膠囊內視鏡尖端之示意圖.....	39
圖 5.2	R-CNN 之示意圖[13].....	40

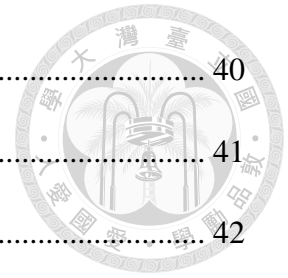


圖 5.3	YOLO 偵測系統[14].....	40
圖 5.4	YOLO 處理網格之過程[14].....	41
圖 5.5	邊界框位置標示示意圖[16].....	42
圖 5.6	YOLO CNN 模型結構[14]	43
圖 5.7	非極大抑制之示意圖.....	44
圖 5.8	Darknet-53 結構之示意圖[16].....	47
圖 5.9	YOLOv3 偵測腸腔之示意圖.....	48
圖 6.1	LabelImg 之介面	49
圖 6.2	Intersection Over Union.....	52
圖 6.3	傳統資料擴增之圖像.....	55
圖 6.4	原始資料集訓練過程之性能.....	56
圖 6.5	圖像平移訓練過程之性能.....	56
圖 6.6	圖像旋轉訓練過程之性能.....	57
圖 6.7	圖像縮放訓練過程之性能.....	57
圖 6.8	加入高斯雜訊訓練過程之性能.....	57
圖 6.9	加入動態模糊訓練過程之性能.....	58
圖 6.10	傳統資料擴增與原始資料集之結果長條圖.....	58
圖 6.11	傳統擴增方式之 PR-Curve.....	59
圖 6.12	幾何變換類擴增的訓練過程之性能.....	60
圖 6.13	顏色變換類擴增的訓練過程之性能.....	61
圖 6.14	幾何變換類、顏色變換類與原始資料集之結果長條圖.....	61
圖 6.15	幾何變換與顏色變換之 PR-Curve.....	62



圖 6.16	傳統資料擴增的訓練過程之性能.....	62
圖 6.17	傳統資料擴增與原始資料集之結果長條圖.....	63
圖 6.18	原始資料集與傳統資料擴增的 PR-Curve 之比較.....	63
圖 6.19	Cycle-GAN 訓練過程之損失	64
圖 6.20	在 Epoch 1 生成之圖像	65
圖 6.21	在 Epoch 50 生成之圖像	65
圖 6.22	在 Epoch 100 生成之圖像	66
圖 6.23	在 Epoch 147 生成之圖像	66
圖 6.24	在 Epoch 199 生成之圖像	66
圖 6.25	Cycle-GAN 擴增訓練過程之性能	67
圖 6.26	Cycle-GAN 與原始資料集之結果長條圖	68
圖 6.27	原始資料集與 Cycle-GAN 的 PR-Curve 之比較	68
圖 6.28	Cycle-GAN 生成失敗之案例	69
圖 6.29	全部擴增訓練過程之性能.....	70
圖 6.30	全部擴增與原始資料集的結果長條圖.....	71
圖 6.31	原始資料集與全部擴增的 PR-Curve 之比較.....	71



表目錄

表 2.1	第一代 MFN Platform 與第二代 MFN Platform 之比較.....	12
表 2.2	第一代 MFN Platform 與第二代 MFN Platform 的控制元件之比較.....	12
表 2.3	磁控大腸膠囊內視鏡之規格.....	14
表 2.4	內視鏡訓練模型之六種不同等級的操作時間.....	16
表 6.1	混淆矩陣 (Confusion Matrix).....	50
表 6.2	傳統資料擴增生成之數量.....	55
表 6.3	傳統資料擴增與原始資料集之結果比較.....	58
表 6.4	傳統資料擴增以變換分類之資料集數量.....	60
表 6.5	幾何變換類、顏色變換類與原始資料集之結果比較.....	61
表 6.6	傳統資料擴增與原始資料集之結果比較.....	63
表 6.7	Cycle-GAN 擴增之數量.....	67
表 6.8	Cycle-GAN 與原始資料集之結果比較.....	68
表 6.9	全部擴增方式擴增之數量.....	70
表 6.10	全部擴增與原始資料集的結果之比較.....	71



第一章 緒論

1.1 研究背景

在現今科技技術越來越發達的社會，人們的飲食習慣及生活作息逐漸不平衡，導致得到消化系統之病人逐年增加。消化系統之有關疾病無法從外表中觀察，因此需要由內視鏡來輔助醫師觀察病患身體內部狀況。內視鏡是醫生觀察病人消化系統不可或缺的工具之一，既可驗證其他診斷方式所得的診斷，亦可使不易察覺之早期病變得到確診，特別是對於消化道早期之惡性腫瘤的發現有獨特的作用。內視鏡隨著科技的進步，從笨重粗大的管狀內視鏡，轉換成了能夠讓病患吞嚥進入到人體的膠囊內視鏡，並且加上其他的科技，如影像處理、人工智慧、影像辨識等等研究方向導入至生醫領域來進行研究。

108年十大癌症死因

癌症別	死亡人數 (人)		死亡率 (每十萬人口)		標準化死亡率 (每十萬人口)	
	年增率 (%)	年增率 (%)	年增率 (%)	年增率 (%)	年增率 (%)	年增率 (%)
所有癌症死亡原因	50,232	1.8	212.9	1.8	121.3	-1.4
1.氣管、支氣管和肺癌	9,701	2.4	41.1	2.3	22.8	-0.8
2.肝和肝內膽管癌	7,881	-5.0	33.4	-5.1	18.8	-8.2
3.結腸、直腸和肛門癌	6,436	5.6	27.3	5.5	14.9	2.2
4.女性乳癌	2,633	4.9	22.2	4.7	13.1	1.4
5.口腔癌	3,425	4.5	14.5	4.4	8.9	1.4
6.前列腺(攝護腺)癌	1,538	5.8	13.1	5.8	7.1	1.9
7.胰臟癌	2,497	7.1	10.6	7.0	5.9	4.0
8.胃癌	2,379	-0.7	10.1	-0.7	5.5	-3.9
9.食道癌	1,983	-2.6	8.4	-2.7	5.1	-5.9
10.卵巢癌	683	6.1	5.7	5.8	3.5	4.5

圖 1.1 108 年十大主要癌症排名[1]

現代社會的十大死因，位居榜首的是癌症，而有關於消化道的癌症都進入到癌症死亡原因的前十名，尤其是大腸癌進入到前三名，如圖 1.1 [1] 所示。大腸癌早期發現時予以拔除會有極佳的治癒率，因此消化系統健檢的重要性近年來又再次提及。醫生藉由大腸內視鏡，來觀察患者的大腸，並做一些簡單的微型手術，處理簡單病變如息肉等。然而，內視鏡本身是一種入侵式之醫療器材，而大腸內

視鏡需要從患者的消化道出口進入，且其管徑約為 11-13 公厘，對於較敏感的患者而言有異物入侵的難受感。再者，大腸內視鏡在消化道推進時，若無經驗者之操作者操作，有可能會在腸壁表面造成損傷。實際上檢測時患者會選擇無痛麻醉檢查。但是患者無法改變姿勢來配合醫生，亦或是手術中無法告知疼痛，增加穿孔的風險性。為了克服這些缺點和局限性，無痛非侵入式膠囊內視鏡應運而生。

以色列 Given Imaging 公司 (現今的 Medtronic 公司) 在 2000 年結合了原為偵測用之微小攝影機與膠囊，並在 2001 年時發行了一種無線式膠囊內視鏡 (Wireless Capsule Endoscopy, WCE)。與傳統的胃鏡或結腸鏡相比，膠囊內視鏡(如圖 1.2 [2])藉助胃腸道的自然蠕動來通過人體，並將沿途的各個部位拍攝成影像且利用無線射頻方式傳送，以利醫生觀察病患完整的消化道系統。



圖 1.2 Medtronic (原 Imaging) PillCam[2]

然而，膠囊內視鏡無法透過外部操作拍攝特定的角度或目標，來判斷是否有腫瘤。到目前為止，膠囊內視鏡還不能夠完全取代傳統的大腸內視鏡。因此，如何讓膠囊內視鏡變得可受控制的研究，將是國際上研究膠囊內視鏡的團隊所面對的一個重要核心。



1.2 研究動機及研究目的

本團隊在這幾年來與萬芳醫院和群曜醫電公司發展磁控膠囊領域之研究。本團隊應用於上消化道之磁控膠囊研究於 2019 年製作成群曜醫電公司之產品—InsightEyes System。磁控膠囊內視鏡是由手持式磁牽引控制手把 (手持式 MFN) 進行控制，其內部有一個步進馬達以控制膠囊內視鏡旋轉角度之永久磁鐵，達到控制膠囊內視鏡的擺姿與移動，而這套系統在上消化道中有良好之表現。然而，由於較高的操作難度與較長的操作時間，並不適合在下消化道中使用手持式 MFN。因此，本團隊於 2017 年開發磁牽引平台 (MFN Platform) 以代替手持式 MFN。MFN Platform 是將手持式 MFN 改良並放置於機器手臂上以提高操作精度和減輕醫生手上之負擔，使得於下消化道之操作能夠順利進行。

本團隊計畫在未來之檢查過程中能夠用 MFN Platform 進行全自動導航，因此需要 MFN Platform 需要更加聰明與智慧，因此引用人工智慧以增強 MFN Platform 之能力。人工智慧 (Artificial Intelligence, AI) 使機器能夠在各種行業與應用程式中提供更多價值。近年來，研究人員已經利用人工智慧來分析大量的醫學資料並執行臨床任務，例如糖尿病引起視網膜之病變的識別，或是皮膚黑色素瘤之診斷。因此，利用人工智慧學習內視鏡辨別腸腔之位置是本團隊目前所研究的課題之一。

為了使得人工智慧能夠更加準確，本團隊需要完善度高且數量多的資料集。然而，完善度高且數量多的資料集需要耗時且耗人力進行標籤或分類。因此，本研究之目的是將數量少卻完善度高之資料集進行資料擴增。本研究所使用的方式有兩種：傳統之資料擴增與生成對抗網路以增加資料量，並將人工智慧訓練得更加準確，以利內視鏡透過人工智慧標籤出腸腔位置來控制擺姿至腸腔位置，使得全自動導航之行進更加順利。



1.3 文獻回顧

由於使用 MFN Platform 控制磁控膠囊內視鏡對準腸腔，來達成自動導航，因此本研究使用深度學習之方式進行腸腔判斷。在進行深度學習之方式進行辨識前，需要準備好訓練用資料集。然而，使用的資料集僅有 2680 張圖像，所以我們將研究進行資料擴增。

使用資料擴增之研究有許多篇。根據 Yu-Dong Zhang 等人 [3] 之研究，他們使用傳統的資料擴增方式進行研究，使用圖像旋轉、伽瑪校正 (Gamma Correction) 與加入噪聲，將原本 1800 張的 18 個種類之水果圖像擴增至 63000 張圖像，並將其輸入至他們建立 13 層深度卷積神經網路，使其能夠分類 18 種水果。他們分別對未經資料擴增之模型與經資料擴增之模型進行測試，發現經過資料擴增之後的模型較為良好，如圖 1.3 [3] 所示。

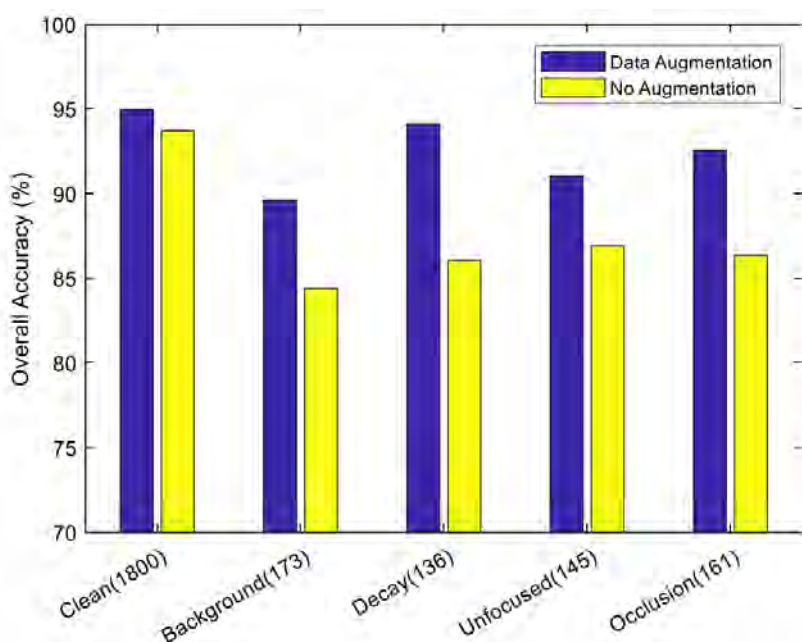


圖 1.3 資料擴增與未經資料擴增之比較[3]

另一個研究是由 Agnieszka Mikołajczyk 等人 [4] 提出，主要關注機器學習領域中最常出現之問題，訓練資料集不足或是資料集中的類平衡不均勻，並將現今之資料擴增方式進行整理。他們欲擴增之圖像有皮膚黑色素瘤、組織病理學之圖

像，與乳房 MRI (Magnetic Resonance Imaging)之圖像。他們提到傳統資料擴增之方法：幾何變換與顏色變換是現今被廣泛作為資料擴增方式使用，也提到一些較不流行的資料擴增方式，如生成對抗網路 (Generative Adversarial Network, GAN)、材質轉移 (Texture Transfer)，及隨機區塊消除資料擴增 (Random Erasing Data Augmentation)等。他們後續使用卷積神經網路將底圖與欲風格轉化之圖像轉換成跟底圖輪廓相近，但風格與另一張圖像相似之合成照片，如圖 1.4 [4] 所示，而利用此將原圖生成 32 種圖像，如圖 1.5 [4] 所示。



圖 1.4 基於底圖與欲風格轉化之圖像所生成之合成圖像[4]

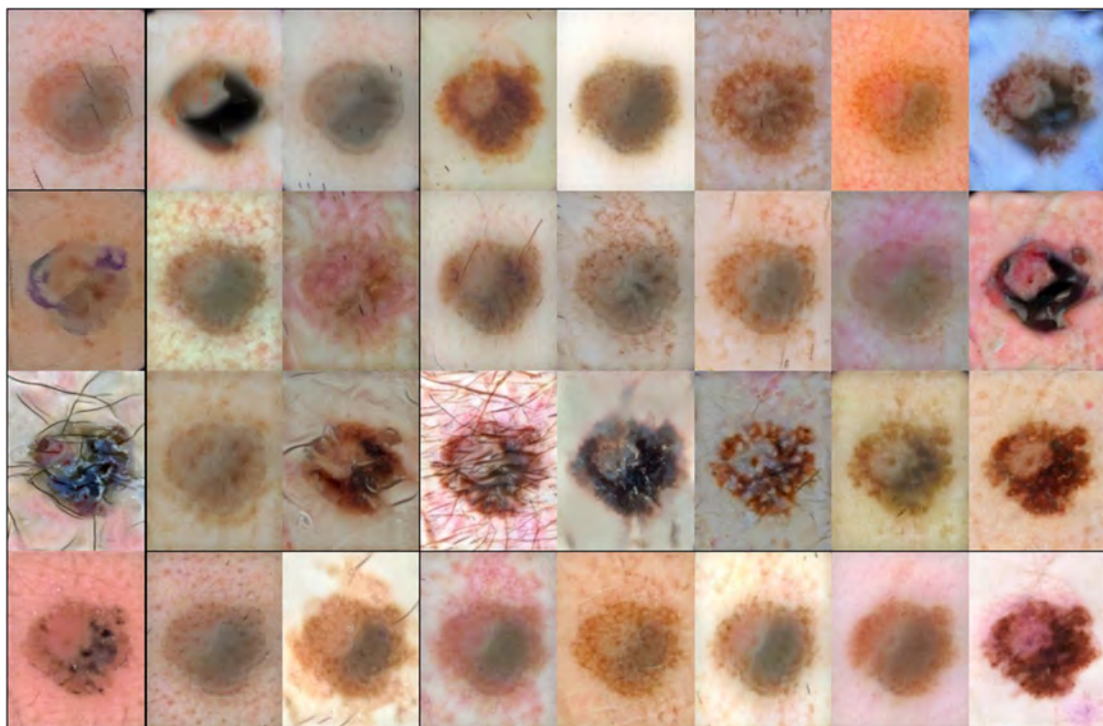


圖 1.5 CNN 生成皮膚黑色素瘤之 32 種圖像[4]

還有一個是由 Jelmer Wolterink 等人 [5] 提出關於 MRI 與 CT (Computed Tomography) 之研究。MRI 是非游離輻射、非侵入式之方式進行成像，成為一種廣泛應用於病患之診斷。然而，MRI 無法提供放射性治療計畫所需要的信息。因此，他們提出利用 Cycle-GAN 將 MRI 之圖像轉換成 CT 圖像，其結果如圖 1.6 [5] 所示， I_{MR} 為原始 MRI 圖像， $Paired$ 為配對訓練集之訓練結果， $Unpaired$ 為非配對訓練集之訓練結果， I_{CT} 為原始 CT 圖像。

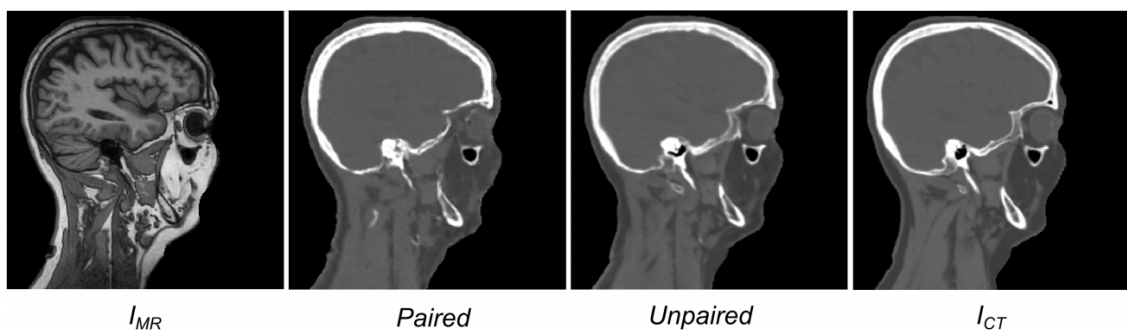


圖 1.6 MRI 經過 Cycle-GAN 轉化成 CT 之結果[5]

因此透過 [4] 整理出多種資料擴增方式，本研究採用在 [3] 中使用的傳統資料擴增與其設置一圖像旋轉、圖像加入噪聲，並新增額外的傳統擴增方式一圖像平移、圖像縮放與圖像加入動態模糊新增至訓練資料集，以及 [4] 所提出生成對抗網路之擴增方式的想法，並加入 [5] 所提出之 GAN 的延伸—Cycle-GAN，將模擬腸道進行風格轉換成真實腸道之擴增方式，來加入至訓練資料集中，使得最新的即時目標偵測深度學習模型—YOLOv3 能夠更加準確地判斷腸腔的位置。



1.4 章節摘要

本論文以如何利用資料擴增來增加資料集，使得膠囊內視鏡能夠透過 YOLOv3 來更精確地瞄準腸腔洞口，論文內容分成七個章節：

第一章：緒論

本章旨為說明本論文研究背景、動機與目的、文獻回顧，及介紹各章節。

第二章：磁牽引平台與磁控膠囊內視鏡之介紹

本章將介紹本團隊開發的磁牽引平台與磁控膠囊內視鏡開發過程與介紹其概況，以及利用此系統所做的動物實驗之成果。

第三章：資料擴增

本章將介紹第一種擴增方式：傳統資料擴增，其所使用之方法與原理。

第四章：生成對抗網路

本章將介紹使用的第二種擴增方式：生成對抗網路，以及後續介紹本研究使用生成對抗網路之延伸：Cycle-GAN 之原理與損失函數。

第五章：腸腔辨識

本章將介紹腸腔辨識所使用之方法，並介紹 YOLO 之原理與架構。

第六章：實驗架構與成果討論

本章主要說明本研究之過程及如何完成此研究，並討論不同的擴增方式下 YOLOv3 表現之性能，最後將所有擴增資料與原始資料集進行比較。

第七章：結論與未來工作

本章對未來可研究方向及改進方法進行總結並說明。

第二章 磁牽引平台與磁控膠囊內視鏡之介紹

本章節首先介紹本團隊研發的磁牽引平台 (Magnetic Field Navigator Platform, MFN Platform) 的開發過程與介紹其概況，接著介紹磁控膠囊內視鏡的開發過程與介紹其概況，最後介紹利用現今的磁牽引平台與磁控膠囊內視鏡操作的動物實驗之成果。

2.1 MFN 之概況與演進

2.1.1 手持式 MFN

本團隊由 劉志文 教授與萬芳醫院 連吉時 院長的帶領下，在 2012 年提出了一種概念創新的控制方式來牽引在上消化道裡的磁控膠囊內視鏡。該團隊所研製的裝置對於操作者在操作上的困難度大幅減少，變得十分容易。他們利用此裝置做了一系列的實驗 [6]，以下將簡單介紹手持式 MFN 動作原理。



圖 2.1 本團隊研製之手持式 MFN[6]

本團隊提出一個可以利用手持方式來操作之 MFN，如圖 2.1 [6]所示，在手持式 MFN 的頂端裝設有一個採用徑向充磁的圓柱式鈷鐵硼 (NdFeB) 永久磁鐵，該永久磁鐵的中心處連結一個步進馬達，進而使永久磁鐵具備轉動之效果。由於在操作時須將手把頂端處的永久磁鐵接觸到受檢者之皮膚上，因此在其外部頂端罩住一個固定在手把上的透明圓罩。此作法不僅可以避免掉永久磁鐵的圓柱稜角與

肌膚接觸的不適感，而且可以隔絕受檢者之肌膚與轉動中之永久磁鐵的接觸。

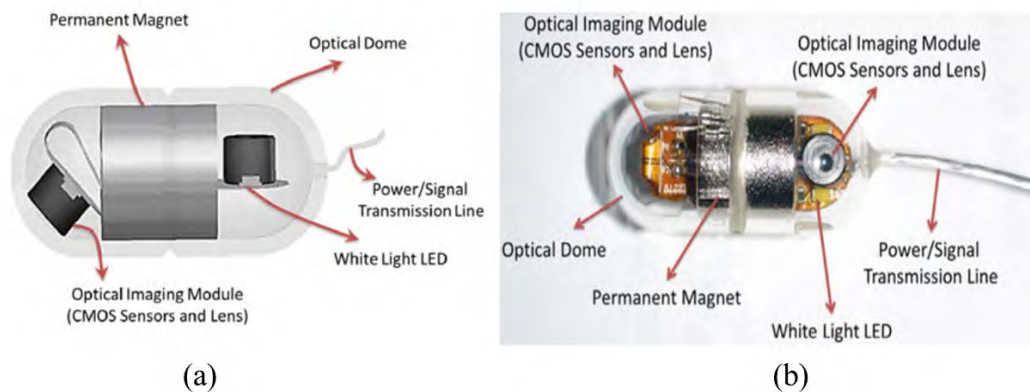


圖 2.2 本團隊研製之有線式磁控膠囊內視鏡 (a)示意圖 (b)實際圖[6]

另外，本團隊當時也自行研製出一款有線式膠囊內視鏡(如圖 2.2 [6] 所示)，中間放置一個採用徑向充磁的鈷鐵硼永久磁鐵，並附帶兩個攝影鏡頭：一個採取與磁鐵軸向垂直的方式放置；另一個採取與磁鐵軸向偏移 45 度的方式放置，如圖 2.2 (a) 所示。採取這樣的設計，當膠囊轉動時，這兩顆鏡頭能掃過的觀測面積就會增加，以利操作者在最短時間看到所欲觀測的範圍。

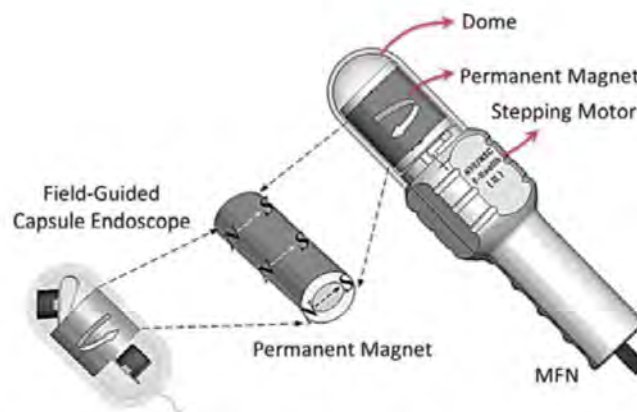


圖 2.3 手持式 MFN 操作示意圖[6]

將手持式 MFN 及有線式膠囊內視鏡以圖 2.3 [6] 方式操作，就能夠使操作者十分方便得在寬廣的胃部進行觀測。該操作概念簡單說明如下：移動方面利用兩個磁鐵靠近時互相吸引的特性，當其中一個磁鐵移動時，另一個磁鐵受到該移動

之磁鐵的影響而跟隨移動得磁鐵；旋轉方面，兩個磁鐵在相互靠近時，若轉動其中一個磁鐵，另一個磁鐵就會受到影響該轉動磁鐵的磁力影響而跟著一起轉動，如圖 2.3 [6]所示。從該示意圖中可知，當在一個旋轉的週期之內，當手持式 MFN 之永久磁鐵為逆時針旋轉時，有線式膠囊內視鏡會順著其轉動牽引的力量順時針旋轉。由此控制概念便能透過體外的手持式 MFN 之永久磁鐵，牽引體內同樣附有永久磁鐵的有線式膠囊內視鏡，進而達到控制有線式膠囊內視鏡的效果。

2.1.2 磁牽引平台 (MFN Platform)

本研究團隊在成功研發上消化道的手持式 MFN 之後，開始利用上消化道的開發經驗來研發下消化道的磁控膠囊內視鏡系統。然而，下消化道的大腸複雜度比起上消化道的胃更加複雜，使得操作手持式 MFN 的時間會更加長久。再加上手持式 MFN 是一件富有重量的醫療儀器，因此對於醫生而言，不希望手上長時間承受較大的負擔。因此，本團隊在 2017 年研製出一個可以牽引內視鏡，並具備 XYZ 移動與俯仰角 (Pitch Angle)、偏航角 (Yaw Angle) 兩軸轉動共五軸自由度的平台。而這就是第一代 MFN Platform [7] (如圖 2.4 [7]、圖 2.5 [7] 所示)。

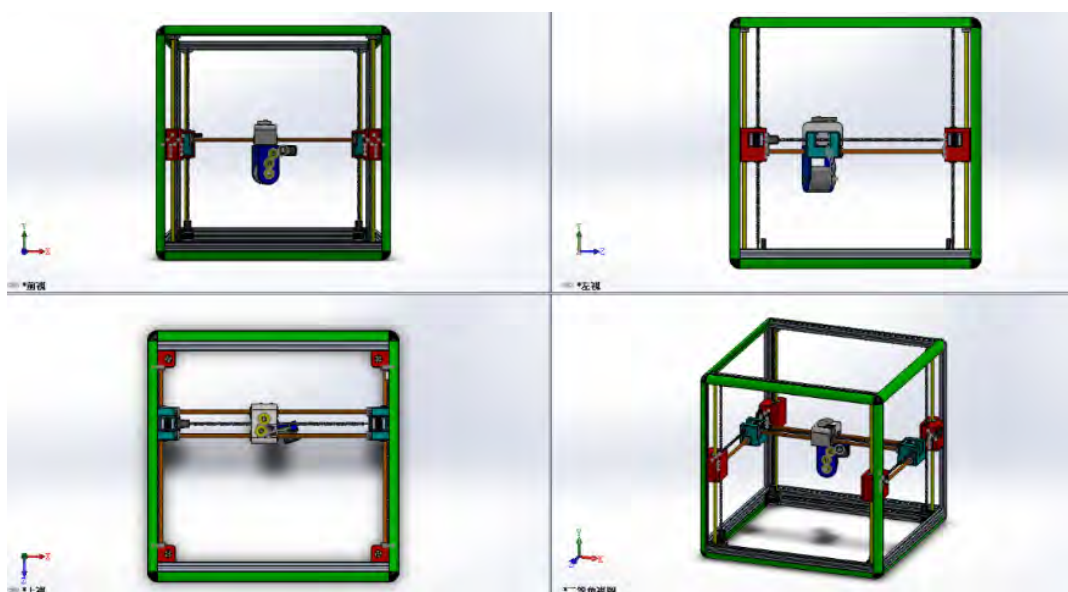


圖 2.4 第一代 MFN Platform 之三視圖[7]

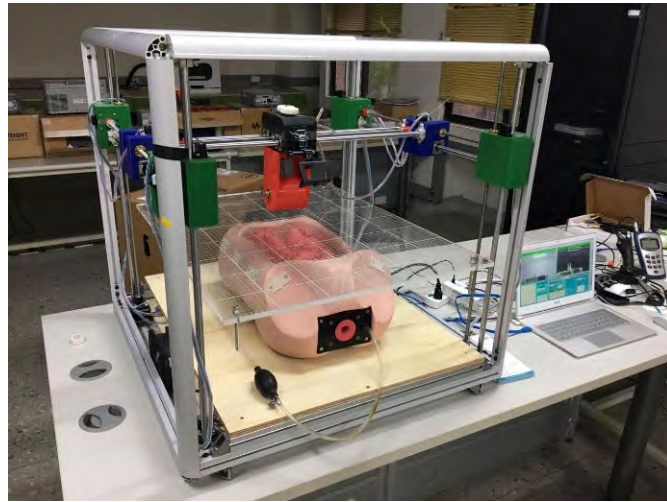


圖 2.5 第一代 MFN Platform 之實際圖[7]

儘管第一代 MFN Platform 可以在人體模型底下進行實驗(如圖 2.5 [7])，但是卻無法放置病床，以至於無法利用此平台來進行動物實驗與臨床實驗，因此在 2019 年本團隊研發出第二代 MFN Platform，如圖 2.6 所示。第二代 MFN Platform 除了可以進行動物實驗和臨床實驗之外，還增加磁控膠囊內視鏡可操作之有效範圍。

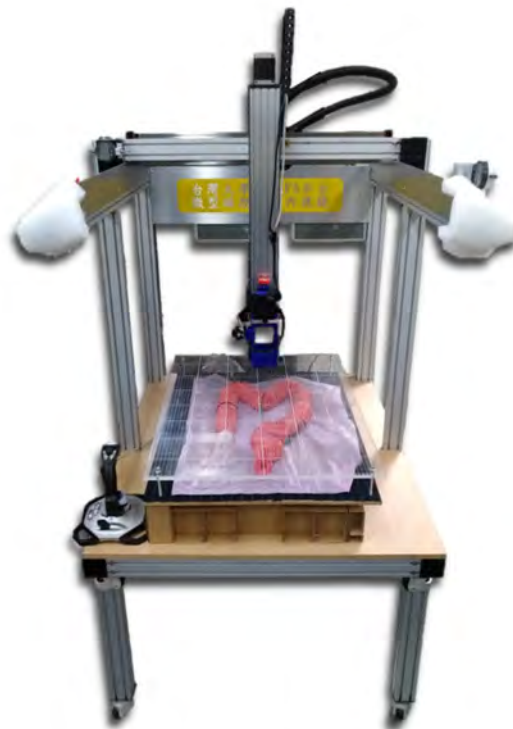


圖 2.6 第二代 MFN Platform 之實際圖

2.1.3 第一代 MFN Platform 與第二代 MFN Platform 之比較

第一代與第二代 MFN Platform 之間都擁有 XYZ 移動與 Pitch、Yaw 兩軸轉動共五軸自由度，然而不管是架構及硬體上，第二代 MFN Platform 都比起第一代來說擁有大幅度的提升，如表 2.1 所示。

表 2.1 第一代 MFN Platform 與第二代 MFN Platform 之比較

	第一代	第二代
機器大小 (cm ³)	80 * 80 * 80	92 * 108 * 180
移動範圍 (cm ³)	40 * 36 * 20	65 * 65 * 41
移動速度	40 mm/s	50 mm/s
移動精度	< 2mm	< 0.01mm
病床相容性	否	是

第二代 MFN Platform 採用嵌入式系統編成設計，實現即時控制，團隊還升級從第一代 MFN Platform 研製的驅動電路，提高第二代 MFN Platform 的穩定性和馬達速度。由於在控制精度上有了很大的提升，使得第二代 MFN Platform 更加精確，控制元件比較如表 2.2 所示。

表 2.2 第一代 MFN Platform 與第二代 MFN Platform 的控制元件之比較

	CPU	時脈	並行運算	馬達角度控制精度
第一代	ATmega2560	16 MHZ	無	0.9°
第二代 (雙處理器)	Teensy 3.6 Cortex-M4F Raspberry Pi 4 Cortex-A72	180 MHz 1.5 GHZ	有	0.05625°



2.2 磁控膠囊內視鏡之概況

本研究團隊在 2012 年研發出有線式磁控膠囊內視鏡之後，將其技術轉移至群曜醫電，並完善與製造出成品，如圖 2.7 所示。與研發並實驗用之膠囊內視鏡的不同，成品只有一顆鏡頭，在觀測上不需注意兩個視角，且減少其製造成本，達到一次性使用之醫療器材的目標。視角從原本的 112° 到 140° ，使得觀測的範圍更加廣闊，讓操作者更加容易觀察上消化道之整體狀況。



圖 2.7 有線式磁控膠囊內視鏡之成品



圖 2.8 磁控大腸膠囊內視鏡

在成功研發與上消化道相關之內視鏡之後，本團隊與鑫視科技公司著手研發大腸所使用之內視鏡。由於大腸含有糞便，無法像上消化道一樣不需考慮鏡頭整潔問題，需進行灑水動作以利鏡頭整潔。除此之外，大腸清理乾淨之後使得腸道

坍塌，需要藉由充氣的動作使得腸道能夠顯現以利觀測。因此醫師的建議下在有線式膠囊內視鏡中加裝一個工作通道，並開發出磁控大腸膠囊內視鏡之 Prototype，如圖 2.8 所示。除了藉由該通道來灑水清潔鏡頭，亦可利用該通道做腫瘤摘除或切片等等之簡單手術，並將影像透過微型腸鏡訊號處理主機 (如圖 2.9) 傳送至 PC，達成以體外磁場產生體內拉力，且能夠牽引長鏡、控制鏡頭擺向的效果。磁控大腸膠囊內視鏡規格如表 2.3 所示。



圖 2.9 微型腸鏡訊號處理主機

表 2.3 磁控大腸膠囊內視鏡之規格

項目	規格
磁力源	N52 鈹鐵棚永久磁鐵
體積	<ul style="list-style-type: none"> • 磁力源(最粗端)：外徑 12 mm、內徑 7.6 mm、高 12 mm • 內視鏡鏡身：外徑 7 mm、長度 2 m
光學鏡頭	解析度：1280 × 720 pixel 之 CMOS 感光元件
LED 照明	5 W 之高亮度 LED 白光照明
工作通道	直徑 3.2 mm
安全性測試	<ul style="list-style-type: none"> • Biocompatible：Endoscope tube • Electrical safety：1.5 kV for secondary side



2.3 動物試驗之成果

本研究團隊進行豬隻活體試驗之目的是為了驗證 MFN Platform 與磁控大腸膠囊內視鏡能夠進行於活體生物之可行性。在整個試驗分為三個部分：進行大腸鏡訓練模型、豬隻活體試驗之操作過程，及試驗之結果。以下將逐一說明。

2.3.1 大腸鏡訓練模型

本研究團隊於 2019 年操作 MFN Platform 與磁控大腸膠囊內視鏡進行豬隻活體試驗。在豬隻活體試驗之前，本團隊先利用人體大腸模型進行練習。人體大腸模型選用 M40 大腸鏡訓練模型，如圖 2.10 所示，其長寬約為成人大小，重約 6.5 公斤，材質為柔軟之特殊樹脂與硬樹脂，經由京都科學株式會社發明[8]。此大腸鏡訓練模型可以在肛門區域的入口處調整打開或閉合，且內部的大腸模型可進行填充空氣的動作使訓練上的視野更加有效。由於大腸模型固定位置可以進行調整，因此根據其說明書可設置多種不同的等級，使操作者在操作儀器方面更加熟能生巧。



圖 2.10 M40 大腸鏡訓練模型[8]

本團隊在大腸鏡訓練模型之中選用前六種不同的難度等級來做練習(如圖 2.11)，這些設置之等級皆按照模擬腸道之說明書進行設置。由於圖 2.11 (d)、(e)、(f)中的大腸模型稍微複雜，腸道有部分重疊在一起，使得操作時間較為略長，達到 30~40 分鐘，但整體平均花費時間仍在 30 分鐘之內，如表 2.4 與圖 2.12 所示。

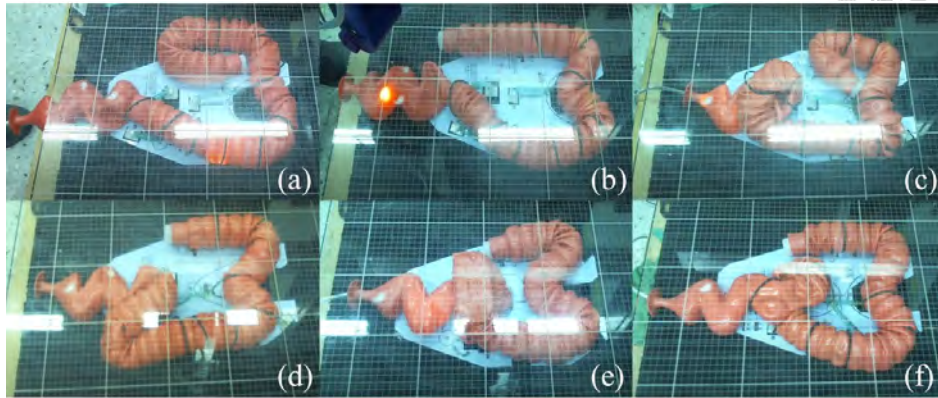


圖 2.11 操作六種不同等級之大腸模型

表 2.4 內視鏡訓練模型之六種不同等級的操作時間

等級	(a)	(b)	(c)	(d)	(e)	(f)	總時間	平均時間
花費時間 (sec)	387	1033	843	2130	2222	2853	9468	1578

六種不同等級的操作時間之長條圖

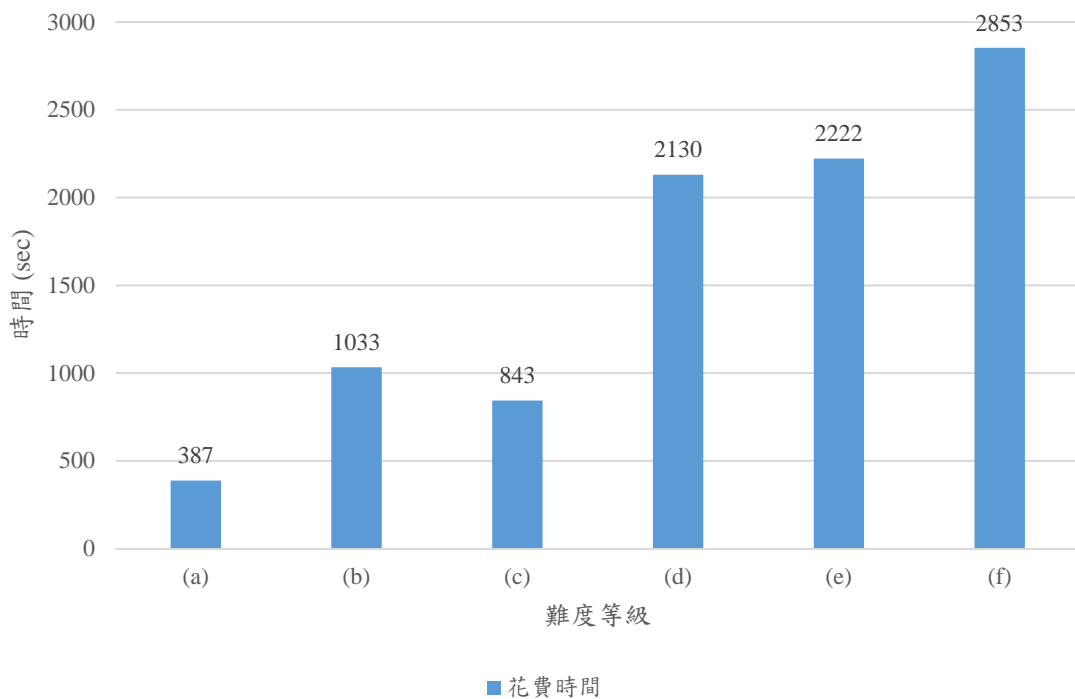


圖 2.12 訓練模型之六種不同難度等級之操作時間長條圖



2.3.2 豬隻活體試驗

本團隊之豬隻活體試驗於國立台灣大學實驗動物資源中心進行試驗，實驗時間從當日上午十時開始，進行至當日下午三時結束試驗，約五小時手術時間。實驗主要硬體配置一台個人電腦約五小時手術時間、MFN Platform、磁控大腸膠囊內視鏡及微型腸鏡訊號處理主機。豬隻活體試驗使用品種之物種為蘭嶼豬 (Lanyu Pig)，經由獸醫和麻醉醫師進行麻醉後由本團隊進行微型內視鏡牽引之操作。

在上午十時對實驗用豬隻進行麻醉，並在半小時之內進行前置作業，測試 MFN Platform 與磁控大腸膠囊內視鏡是否正常工作。在試驗過程中由本團隊之成員負責操作 MFN Platform、一位成員進行灑水充氣以利操作者進行操作、一位成員進行推動磁控大腸膠囊內視鏡、其他成員從旁協助並與上述成員輪流操作，並由萬芳醫院之合作醫師在旁進行指導，如圖 2.13 所示。



圖 2.13 豬隻活體試驗之過程圖

在實驗過程中本團隊的操作由四個攝影鏡頭進行拍攝豬隻活體試驗過程，如圖 2.14 所示。該圖之(a)和(b)為側邊錄影方式拍攝，(a)為手機錄影，(b)為相機錄影、(c)為俯視錄影拍攝，錄影設備為網路攝影機，以及(d)為磁控大腸膠囊內視鏡之影像。



圖 2.14 豬隻活體試驗之拍攝位置

豬隻之大腸與人體之大腸相比複雜許多，呈現螺旋狀之結構。本團隊利用 MFN Platform 之永久磁鐵帶動嵌於磁控大腸膠囊內視鏡之永久磁鐵，達到「拉力為主，推力為輔」的控制策略，不過因豬隻之大腸有些部分是需要進行向下牽引，礙於 MFN Platform 與磁控大腸膠囊內視鏡中兩永久磁鐵皆為互相吸引之現象，因此在試驗過程中本團隊遇到操作上的困難。為了克服操作上的困難，本團隊需要在鏡頭對準腸道之後釋放磁鐵並進行輔助推力行進，以通過向下方向之彎道。

在操作磁控大腸膠囊內視鏡過程中，當內視鏡鏡頭所在位置離腹部表面相近時，大腸鏡之 LED 光源會從腸道中透出皮膚，如圖 2.15 標示處所示，此時 MFN Platform 與磁控大腸膠囊內視鏡中兩永久磁鐵之吸引力為 500 公克，此數值由 MFN Platform 之機械手臂末端感測器所量測，且此吸引力大小尚在健康正常豬隻大腸可承受壓力範圍內。本團隊利用此現象在試驗過程中在一定程度上解決磁控大腸膠囊內視鏡在大腸中迷航之問題。



圖 2.15 磁控大腸膠囊內視鏡之光源透出示意圖

2.3.3 豬隻活體試驗之結果

在試驗過程中，豬隻之腹部並不是平坦的，因此操作 MFN Platform 時，除了操作成員需要時常調整高度之外，還需要其他成員幫忙提醒 MFN Platform 之機械手臂是否碰到豬隻之腹部表面，如圖 2.16 所示。此外本團隊也利用磁控大腸膠囊內視鏡拍攝大腸內部，並記錄整體實驗操作過程，如圖 2.17 所示，左上角為標記出該試驗時間點所拍攝之影像。



圖 2.16 MFN Platform 之機械手臂碰到豬隻腹部示意圖

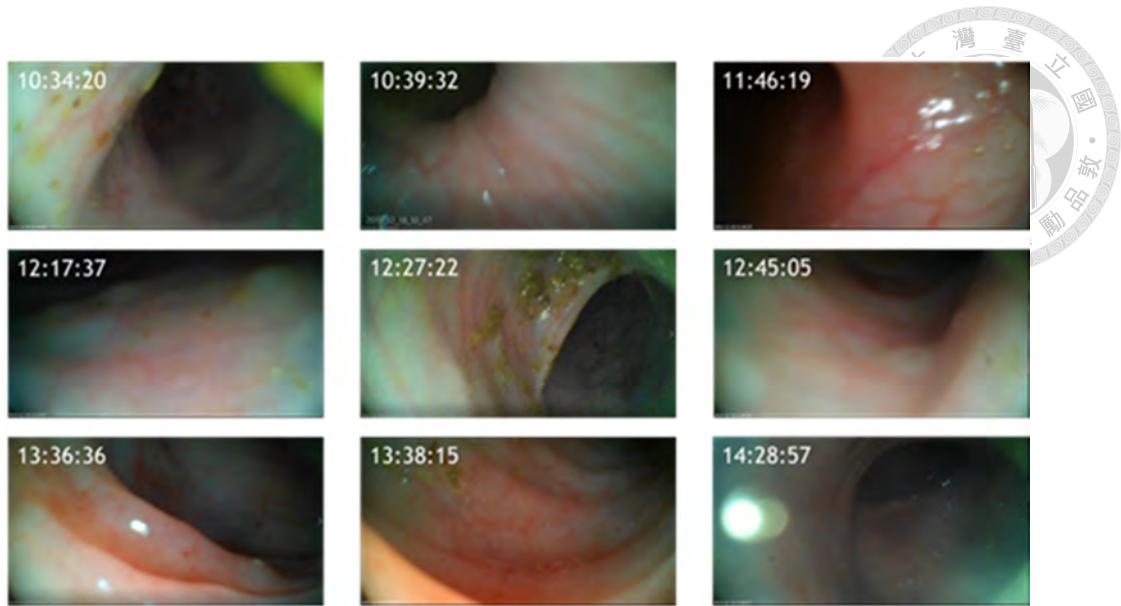


圖 2.17 磁控大腸膠囊內視鏡影像擷取

在結束試驗之後，為了瞭解 MFN Platform 與磁控大腸膠囊內視鏡之運動對大腸壁之影響，本團聘請專業獸醫對豬隻進行人道安樂死並進行術後解剖分析。在上述內容中提及，豬隻大腸與人體之大腸結構相比複雜許多，呈現螺旋狀之結構，故在操作上較為困難。於是在試驗中，為了使磁控大腸膠囊內視鏡方便操作，故將腸道內部注入適量的水與空氣，使內視鏡能夠擁有良好視野，解剖時從活體內部中拿取的大腸如同灌氣過後之氣球一般，如圖 2.18 所示。

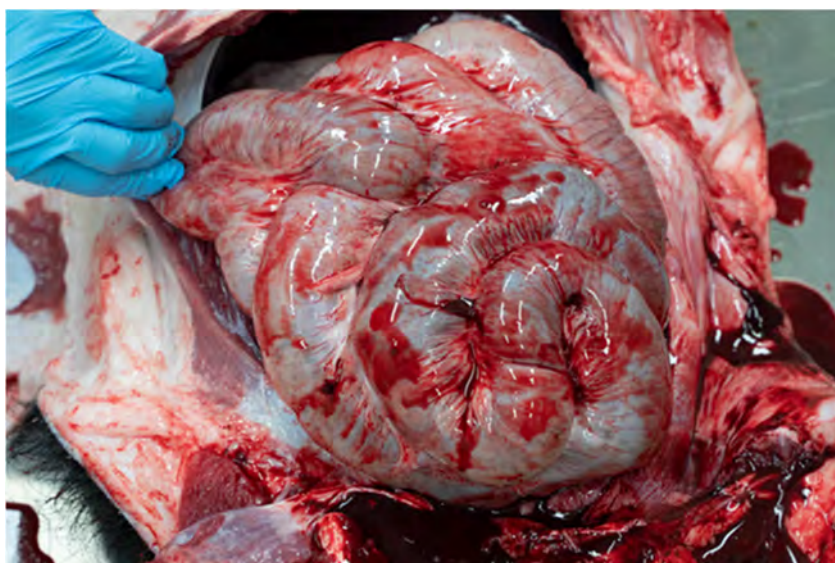


圖 2.18 豬隻解剖示意圖

接著將提取出的大腸進行解剖，使本團隊了解磁控大腸膠囊內視鏡實驗行進距離。本團隊之磁控大腸膠囊內視鏡因適用於人體之大腸，故整體試驗沒有走到豬隻大腸之二分之一位置。磁控大腸膠囊內視鏡進去腸道之長度約為 115 公分，但由於內視鏡在腸道中可能有彎曲的現象，以致內視鏡實際走進腸道深度約為 100 公分，如圖 2.19 所示，黑色標線約為磁控大腸膠囊內視鏡進入腸道之最終位置。

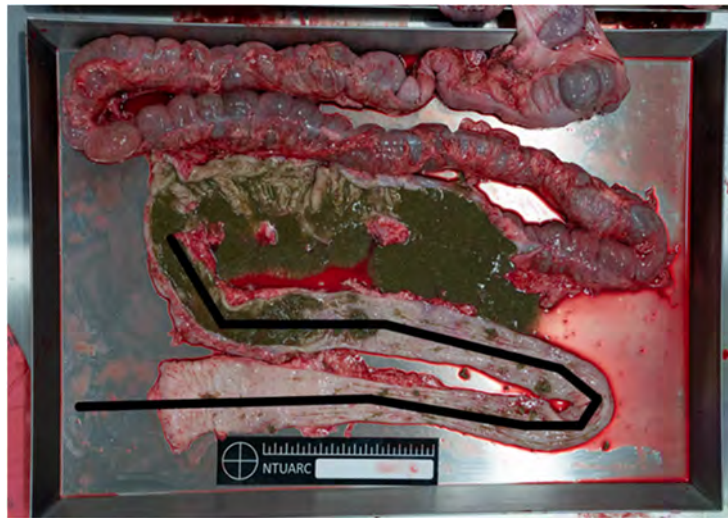


圖 2.19 磁控大腸膠囊內視鏡進入腸道實際深度之示意圖

清理腸道糞便後，觀察腸道中有以肉眼看到之傷痕存在，尤其是於直腸端往大腸方向約 1 公尺處的部分有約 1*5 平方公分的病變，如圖 2.20 所示。經過獸醫之判斷此傷口在用內視鏡過程中是可接受之範圍，若須確定病理診斷仍需要進行組織病理學檢查，獸醫病理診斷報告書如圖 2.21 所示。



圖 2.20 腸道傷痕之示意圖



肉眼病理診斷報告 Gross Pathology Report

日期：2019/12/18
地點：實驗動物資源中心
物種：蘭嶼豬

安樂死方法：
深度麻醉後給予KCl 3mmol/kg。

解剖流程：
打開胸腹腔後，將腹腔腸道分離出體外進行檢查。腸道檢查步驟為從直腸端剪開直腸與大腸腸道，並暴露出粘膜面，檢查長度約1.5公尺。

肉眼病理發現：
於直腸端往大腸方向約1公尺處的大腸黏膜有約 1 x 5 公分的紅色輕微異常病變。此病變可能為黏膜潮紅或輕微粘膜出血。如需確定病理診斷仍需進行組織病理學檢查。
其他部位腸道，肝臟，脾臟，腎臟，肺臟與心臟皆無明顯異常。

獸醫師：

圖 2.21 肉眼病理診斷報告

2.3.4 結論與未來工作

從試驗結果可證明運用 MFN Platform 搭配磁控大腸膠囊內視鏡做動物實驗之可行性。外部永久磁鐵除了可以有效牽引內視鏡之外，Yaw 及 Pitch 皆能有效的被控制。以下是本團隊在動物實驗中所發現的問題與提出之改進方式。

- (1) MFN Platform 的伺服馬達之控制尚有缺陷，造成 Pitch 有問題。

改進方式：更換馬達及更新控制馬達之程式

- (2) Yaw 控制因線長與佈線方式之限制，無法 360 度旋轉，造成控制上較不直覺。

改進方式：增加滑環後並更新程式以達到 360 度旋轉。

- (3) 內視鏡尋找腸腔耗時 (大約佔了本試驗之八成時間)，需要靠大量人工操作搜尋來搜尋。

改進方式：討論鏡頭從 100°之視野改成廣角以利搜尋腸腔。



- (4) 目前機械手臂定位內視鏡的方式為被動式，雖然可以有效地定位到內視鏡，但當助手推內視鏡時、或是機台搖晃的時候，則立即遺失其位置，並另須重新定位。

改進方式：提高機台運動時之穩定性，以及整合本團隊研發之磁定位技術。

- (5) 此試驗使用 N52 灌磁之永久磁鐵，磁力可以控制方向、擺姿，但操作於大腸較下方的空間區域，磁鐵牽引力稍嫌不足，僅能靠助手在後方推動內視鏡來前進。

改進方式：提高磁力強度，並考慮以方形磁鐵取代圓柱形磁鐵以增加特定方向之可控性。

未來希望能將上述提出之改進方式完善，並達成 MFN Platform 與磁控大腸膠囊內視鏡之電器安全與生物相容性之試驗，以利後續的磁控大腸膠囊內視鏡臨床試驗能夠有更佳的操作體驗。



第三章 資料擴增

在進行深度學習訓練時，經常需要海量的資料以確保訓練資料時不會產生過度擬合(over-fitting)的現象。然而，需要海量的資料意味著需要更多的人力與時間之成本進行分類與標籤。資料擴增是一種透過現有資料集，利用一些方法使得其圖像數量增加，可以讓有限的資料產生等價於更多資料之價值。原圖像經過資料擴增處理後，雖然在人眼能夠辨識與原圖為相同之圖像，但是對於神經網路訓練時兩者為不同之圖像。資料擴增無須利用過多的時間與人力蒐集新的圖像，而且可以透過擴增之方法將標籤之邊界框連同圖像一起擴增。在本研究中，使用兩種方法擴增圖像，一種是監督式之方法進行，在採用預設之資料上變換規則，在已有資料的基礎上進行資料擴增，為單樣本資料擴增，也就是傳統之資料擴增；另一種為非監督式之方式進行，通過模型學習資料分步，隨機生成與訓練資料集分佈一致之圖片，為生成對抗網路 (GAN)。

本研究進行圖像平移、圖像縮放、圖像旋轉、加入高斯雜訊與加入動態模糊這五種傳統資料擴增方式的原因為這五種現象皆為即時影像中較為常見之現象。圖像平移對應於內視鏡在腸道之中進行小幅度地擺頭動作；圖像縮放對應於內視鏡向前之後的目標會進行放大；圖像旋轉對應於內視鏡鏡頭進行旋轉；加入高斯雜訊對應於操作者使用使用者介面進行操作時介面會摻雜高斯雜訊，最後動態模糊對應於操作者結束操作時將內視鏡取出會需要進行觀察，可能抽取過程過於快速導致錯過物體之位置，如腫瘤或傷口等。以下將簡單介紹研究使用之傳統資料擴增的原理。

3.1 幾何變換類

在擴增圖像中，經常需要對圖像進行各種操作，如平移、縮放、旋轉等，而這些操作都屬於圖像之仿射變換。仿射變換是一種二維坐標轉到另一個二維坐標之間的線性轉換，可保持二維圖像之平直性、平行性。以下為其數學式：



$$\begin{cases} u = a_1x + b_1y + c_1 \\ v = a_2x + b_2y + c_2 \end{cases} \quad (3.1)$$

將公式 (3.1) 對應的齊次坐標矩陣表示：

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.2)$$

利用這個齊次坐標矩陣對，並利用原圖像處理以生成多種新圖像，本文所使用之方式包括平移、縮放、與旋轉：

- 圖像平移：

圖像平移是將圖像的所有像素坐標進行水平或垂直方向之移動，也就是所有像素按照給定的偏移量在水平方向上沿 x 軸、垂直方向上沿 y 軸移動，如圖 3.1 所示，左圖為原圖，右圖為經過向右平移與向上平移之新圖像。以下為其數學式：

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & dx \\ 0 & 1 & dy \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.3)$$

其中

dx ：x 軸移動之偏移量。

dy ：y 軸移動之偏移量。

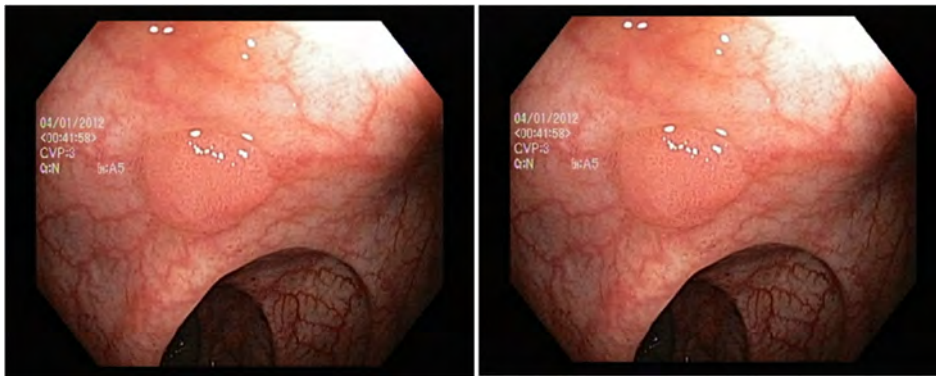


圖 3.1 圖像平移



- 圖像縮放：

圖像縮放是將圖像的所有坐標進行放大或縮小，並在水平方向沿著 x 軸縮放 s_x 倍、垂直方向沿著 y 軸縮放 s_y 倍，如圖 3.2 所示，左圖為原圖，右圖為經放大過後之新圖像。以下為其數學式：

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.4)$$

其中

s_x ：在 x 軸縮放之倍率。

s_y ：在 y 軸縮放之倍率。



圖 3.2 圖像縮放

- 圖像旋轉：

圖像旋轉是將圖像的所有坐標進行順時鐘旋轉或逆時鐘旋轉，如圖 3.3 所示，左圖為原圖，右圖為經過逆時針旋轉 15° 之新圖像。以下為圖像旋轉之原理。

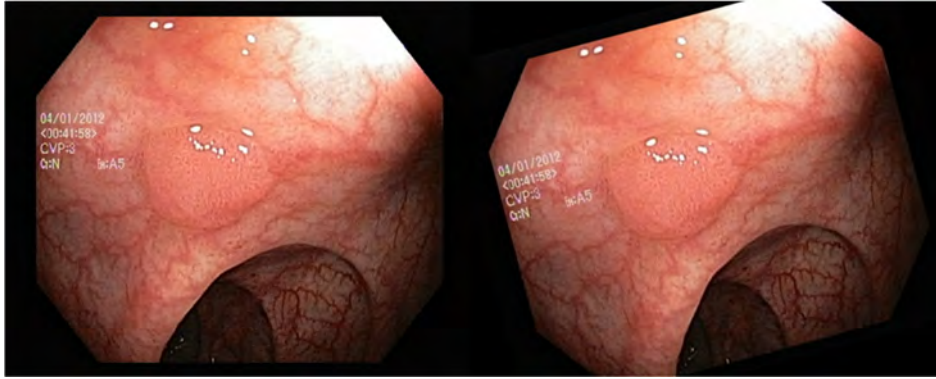


圖 3.3 圖像旋轉

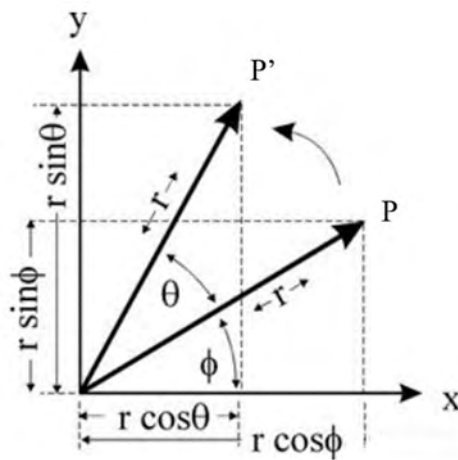


圖 3.4 任一點圍繞原點旋轉之示意圖

將圖像之像素點 P 點在圍繞原點旋轉 θ 至 P' 點，如圖 3.4 所示，P 點與 P' 點以極坐標表示為：

$$P = \begin{cases} x = r \cos \phi \\ y = r \sin \phi \end{cases} \quad (3.5)$$

$$P' = \begin{cases} u = r \cos(\theta + \phi) \\ v = r \sin(\theta + \phi) \end{cases} \quad (3.6)$$

其中

θ : P 點和 P' 點之間的角度。

ϕ : P 點以極坐標表示之角度。



將公式 (3.6) 利用正弦與餘弦之展開式展開，並將公式 (3.5) 代入至展開式，可得公式 (3.7)、(3.8)：

$$\begin{aligned} u &= r \cos(\theta + \phi) \\ &= r \cos \theta \cos \phi - r \sin \theta \sin \phi \end{aligned} \quad (3.7)$$

$$\begin{aligned} &= x \cos \theta - y \sin \theta \\ v &= r \sin(\theta + \phi) \\ &= r \sin \theta \cos \phi + r \cos \theta \sin \phi \\ &= x \sin \theta + y \cos \theta \end{aligned} \quad (3.8)$$

將公式(3.7)、(3.8)轉換成矩陣以表示可得：

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.9)$$

其中

θ ：圖像旋轉之角度。

3.2 顏色變換類

上面的幾何變換之操作，沒有改變圖像本身之內容，可能是選擇圖像之一部分或者對畫素進行了重分佈。如果要改變圖像本身內容，就可以使用顏色變換之資料擴增，常見之擴增方式包括雜訊、模糊、顏色變換等，而本文所使用之擴增方式為高斯雜訊與動態模糊。

- 雜訊：

雜訊在圖像上常引起較強視覺效果的孤立像素點或像素塊。一般來說，雜訊信號與需研究圖像內容不相關，它以無用之信息方式表現，以擾亂圖像的可觀測信息。常見之雜訊包括椒鹽雜訊、高斯雜訊等，本文所使用之擴增方式為圖像加入高斯雜訊。

高斯雜訊是圖像中經常見到的一種雜訊。它是一種機率密度函數為常態分佈（見公式 (3.11)）之雜訊。換句話說，將雜訊轉換到頻域，其分佈就會呈現高斯分佈，如圖 3.5 所示。高斯雜訊之發生原因通常是由於不良照明和高溫引起的傳感器而產生。圖像加入高斯雜訊之成果如圖 3.6 所示。

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3.10)$$

其中

μ ：期望值，決定雜訊之位置。

σ ：標準差，決定雜訊分佈之幅度。

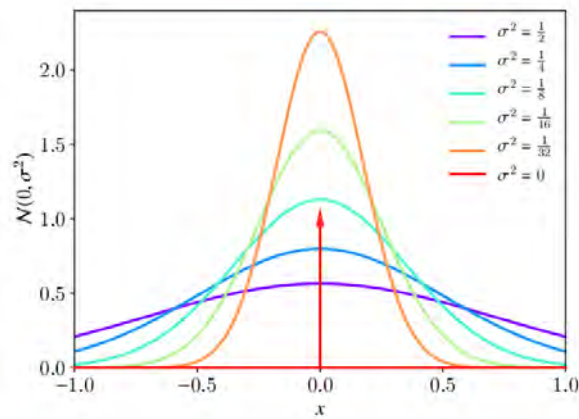


圖 3.5 高斯分佈

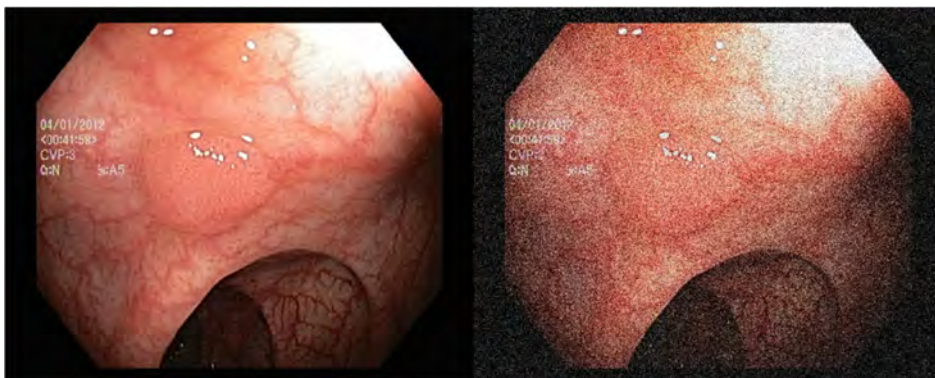


圖 3.6 加入高斯雜訊之圖像



- 模糊：

模糊是圖像處理中最簡單和常用的操作之一，其產生模糊之原因主要有：光學系統的像差、光學成像衍射、成像系統的非線性畸變、成像過程之相對運動等。使用圖像模糊之擴增方式的原因之一是為了給圖像預處理時候降低噪音。圖像模糊處理方式可以使用平均或積分運算，使得圖像中的邊界、輪廓變得模糊。

動態模糊是由於相機快門的有限擷取時間與相機和物體間之相對運動所產生。在評估使用此模糊之原因是由於 MFN Platform 帶動膠囊內視鏡，而內視鏡之運動過程中影像所擷取之圖像為動態模糊，因此藉由動態模糊之擴增方式改變圖像之內容達到模糊效果（如圖 3.7 所示），並加進資料擴增之資料集。

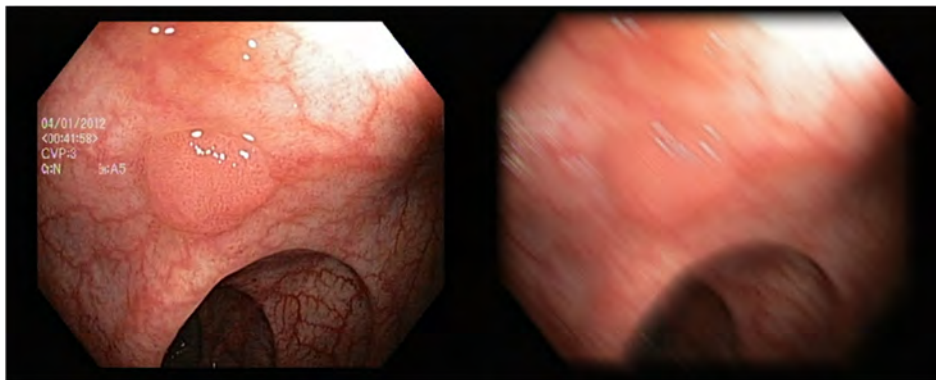


圖 3.7 加入動態模糊之圖像

第四章 生成對抗網路



在進行傳統上的資料擴增之後，對於資料集的數量增加是有幫助的。除了利用傳統資料擴增這種有監督的資料擴增之外，本研究加入一種非監督式的資料擴增方式進行，為利用深度學習之神經網路幫助資料集生成更多資料，而最典型之方法為生成對抗網路。

4.1 生成對抗網路

生成對抗網路 (GAN) 是一種非監督式學習的一種方法，由 Ian Goodfellow 等人 [9] 於 2014 年提出。GAN 通過讓兩個神經網路相互博弈的方式進行學習，其背後最基本的目的是為了使訓練資料集生成更多的資料集。GAN 啟發於博弈論中的二人零和博弈，其原理在於參與博弈的雙方在嚴格競爭下，一方的收益必然為另一方之損失，博弈雙方的收益和損失相加總合永遠為零，不存在合作之可能性。換句話說，GAN 由兩個神經網路組成：生成網路(Generative Network) G 和判別網路(Discriminating Network) D 來進行博弈。

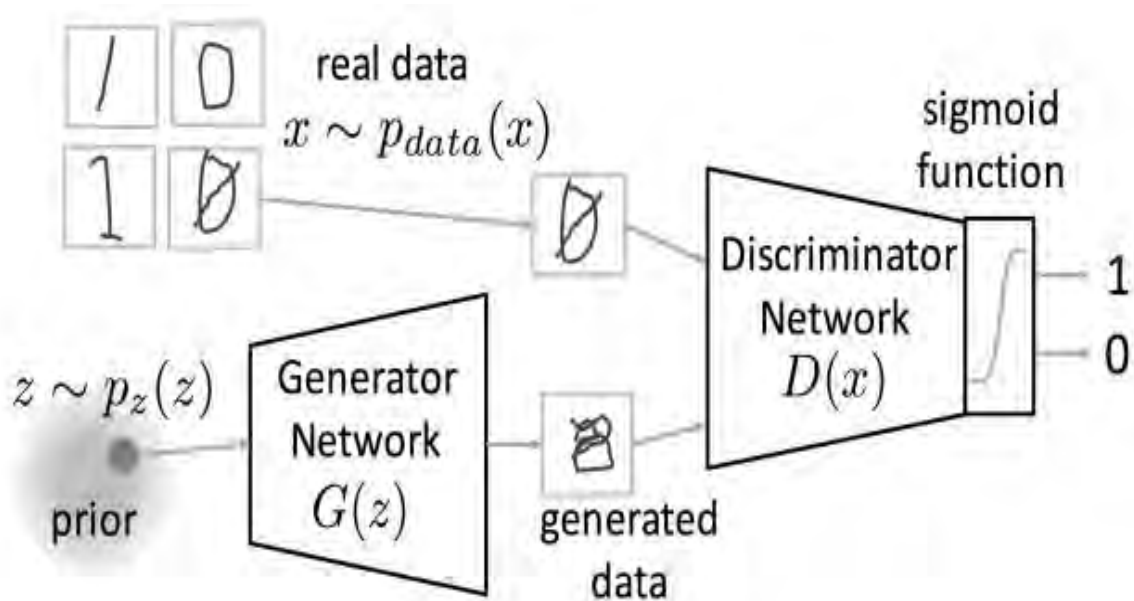


圖 4.1 生成對抗網路[10]



GAN 的兩個網路各司其職：生成網路 G 負責捕捉樣本數據的分布；判別模型為一個二分類器，判別輸入是真實的數據還是生成的樣本，如圖 4.1 [10] 所示。其過程中 x 為真實圖像，而真實圖像符合 $p_{data}(x)$ 之分布； z 為雜訊數據，而雜訊向量符合 $p_z(z)$ 分布，如高斯分布或均勻分布。接著從雜訊 z 進行抽樣並通過生成網路 G 生成數據 $x = G(z)$ 。最後將真實數據及生成數據一起送入判別網路 D ，來輸出判定結果。

在訓練過程中，生成網路 G 的目標就是盡量生成真實的圖像去欺騙判別網路 D ，而判別網路 D 的目標為盡量把生成網路 G 生成之圖像和真實圖像分別開來。因此， G 和 D 構成一個動態的博弈過程。以下將博弈過程之目標函數（亦為損失函數）用數學式表示：

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (4.1)$$

其中

$V(D, G)$ ：為 G 和 D 之目標函數加總（對抗損失）。

x ：符合真實資料的分布 $p_{data}(x)$ 之真實圖像。

$D(x)$ ：判別網路 D 判斷真實圖像是否真實之概率。

z ：符合高斯分布或均勻分布 $p_z(z)$ 之雜訊向量。

在整個博弈過程中，如圖 4.2 [10] 所示， D 會設法增加 $V(D, G)$ 之值，而 G 會減少 $V(D, G)$ 之值。優化 D 的時候需繪製很好的黑色虛線，使它能夠區分真實圖像和生成圖像，因此希望 $D(x) \approx 1$ ， $D(G(z)) \approx 0$ 。相反地，優化 G 的時候，生成圖像能夠更加接近真實圖像的樣子，將 $D(G(z)) \approx 1$ ，使得 D 難以區分圖像的真實性。持續反覆博弈直到最後再也畫不出區分兩者之黑色虛線，也就是 $V(D, G)$ 之值達到一個平衡點。

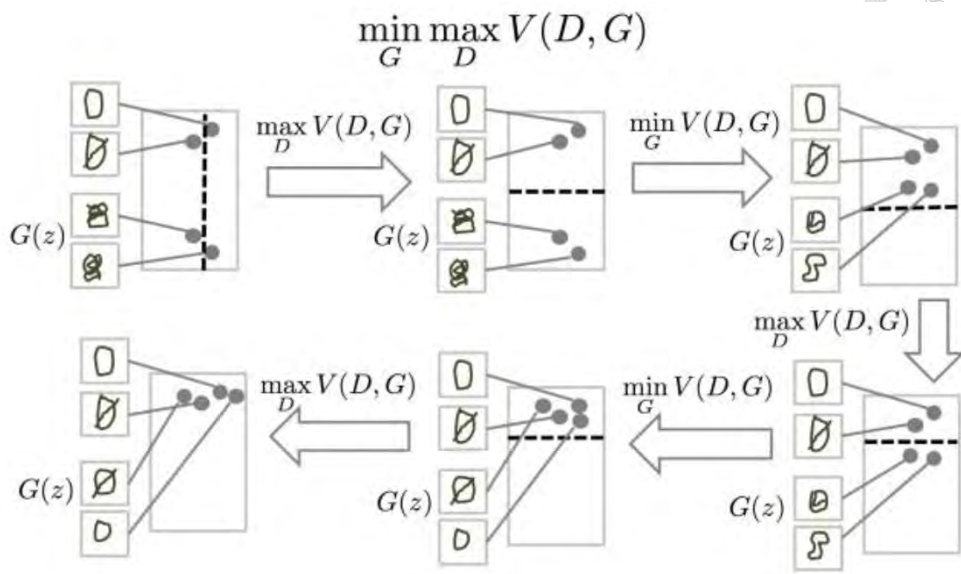
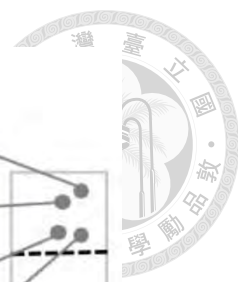


圖 4.2 GAN 優化之訓練過程[10]

4.2 Cycle-GAN

GAN 在增加資料集上是符合研究上之需求，但是由於在訓練資料時給一個輸入就需要相匹配的另外一個輸入，也就是模型訓練時需要依賴兩兩匹配的圖像，使得 GAN 學習的適應性有限，如圖 4.3 所示，所以在準備訓練用資料集時需要人力去準備。其次，所使用的資料集中除了有真實腸道之資料集之外，還有模擬腸道的資料集，因此欲利用模擬腸道轉換成真實腸道，以增加真實腸道之資料集。

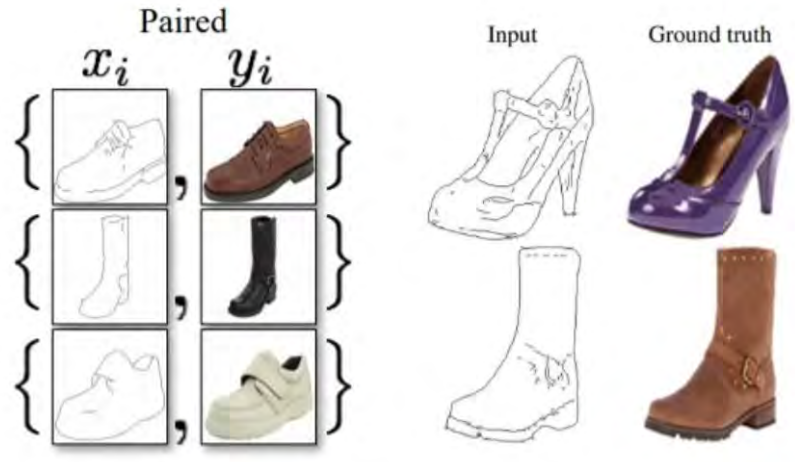


圖 4.3 資料成對之示意圖

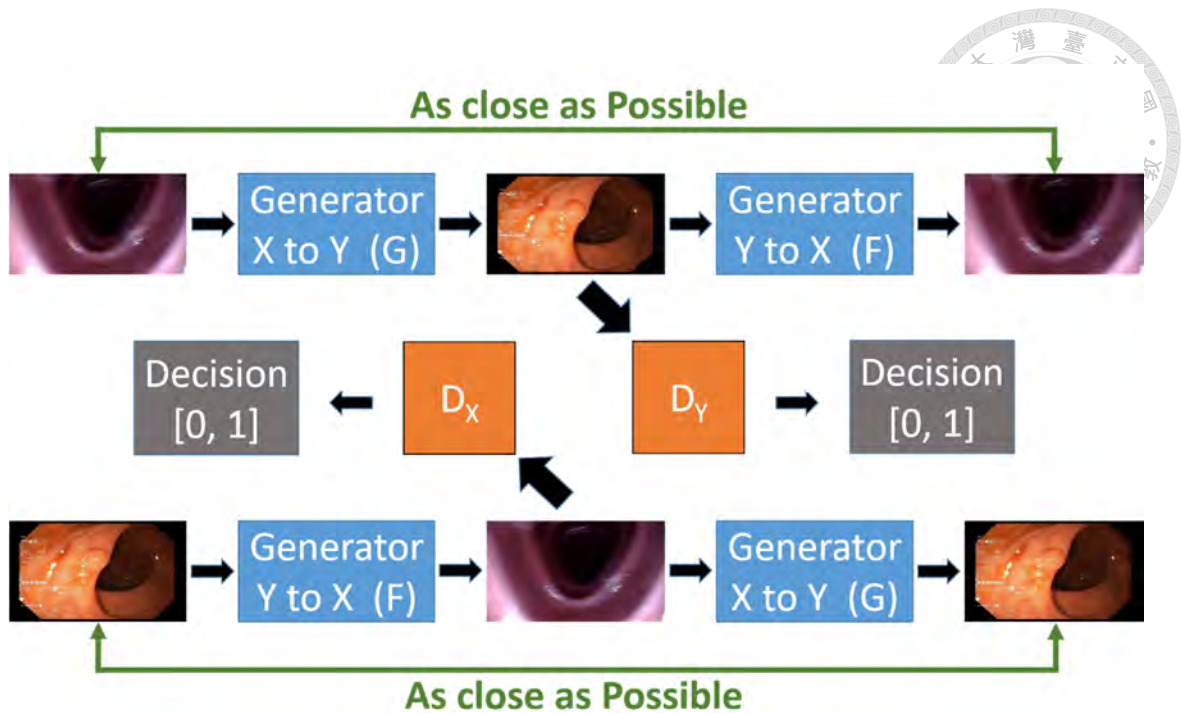


圖 4.4 Cycle-GAN

Cycle-GAN，如圖 4.4 所示，在 2017 年由 Jun-Yan Zhu 等人 [11] 發表，並建立一個網站提供開源碼 [12]。Cycle-GAN 的學習目標是將資料集 X 和 Y 風格之間的轉換，而非具體之資料集 X 和 Y 之間的映射關係，如圖 4.5 [11] 所示。因此，Cycle-GAN 對於資料互相對應的依賴性不存在，有效規避模型訓練時需要依賴匹配的圖像之困難。

從圖 4.4 可知，與傳統的 GAN 最大之不同在於 Cycle-GAN 多出了一組生成與判斷的模型，並形成一個循環。在缺乏循環的情況下，雖然可以照常生成出圖像，然而卻難以優化判別網路。因此，利用這個循環可以使得資料正規化，確保前後向傳播一致，以利模型驗證並提高質量。在訓練過程中，會同時訓練這兩組不同的生成網路和判斷網路，並將兩組的對抗之損失函數與循環之損失函數做結合，達到非成對圖像之目標轉換，即風格轉換，如圖 4.5 [11] 右圖所示。

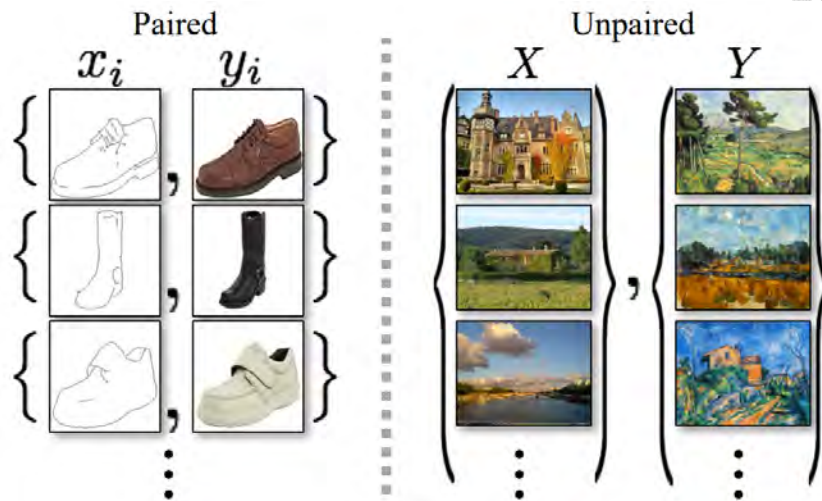


圖 4.5 資料成對與非資料成對之示意圖[11]

4.3 損失函數

Cycle-GAN 生成之圖像需要滿足兩個條件：一為生成之圖像應盡量是真實的；二為輸入之資料集 X 的圖像和生成之資料集 Y 的圖像應該要相互對應。生成之圖像盡量是真實的部分需要 D_X 與 D_Y 進行判別，因此這部分需由 GAN 的損失來處理，而 GAN 的損失函數與前面章節 4.1 類似。

第二種條件由於傳統 GAN 是將資料集 X 映射到資料集 Y，能夠找到應對之特徵，因此不存在此問題。若 Cycle-GAN 忽略此問題而直接訓練，會造成生成網路生成出與原圖不符合之圖像，且能騙過判別網路，導致前後無法匹配。因此，需要引入循環一致性以防止學習到之映射關係將使得生成網路 G 與 F 相互矛盾。

Cycle-GAN 之總損失函數由多個損失函數組合而成，以下為數學式表示：

$$\begin{aligned} \mathcal{L}(G, F, D_X, D_Y) = & \mathcal{L}_{GAN}(G, D_Y, X, Y) \\ & + \mathcal{L}_{GAN}(F, D_X, Y, X) \\ & + \lambda \mathcal{L}_{cyc}(G, F) \end{aligned} \quad (4.2)$$

其中：

\mathcal{L}_{GAN} ：GAN 對抗之損失函數。

G、F：GAN 之生成網路。



D_X 、 D_Y ：GAN 之判別網路。

X 、 Y ：資料集。

λ ：控制兩種損失函數之比重。

\mathcal{L}_{cyc} ：循環一致性之損失。

從公式 4.2 可知，損失函數分成兩部分，為 GAN 對抗之損失函數與循環一致性之損失函數，以下將詳細解釋：

- GAN 對抗之損失函數：

$$\begin{aligned}\mathcal{L}_{GAN}(G, D_Y, X, Y) = & \mathbb{E}_{y \sim p_{data}(y)} [\log D_Y(y)] \\ & + \mathbb{E}_{x \sim p_{data}(x)} [\log(1 - D_Y(G(x)))]\end{aligned}\quad (4.3)$$

$$\begin{aligned}\mathcal{L}_{GAN}(F, D_X, Y, X) = & \mathbb{E}_{x \sim p_{data}(x)} [\log D_X(x)] \\ & + \mathbb{E}_{y \sim p_{data}(y)} [\log(1 - D_X(F(y)))]\end{aligned}\quad (4.4)$$

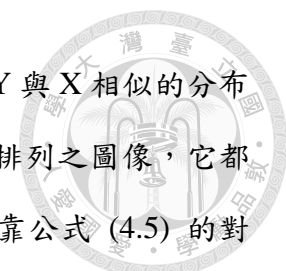
其中：

x ：符合真實資料的分布 $p_{data}(x)$ 之真實圖像。

y ：符合真實資料的分布 $p_{data}(y)$ 之真實圖像。

\mathcal{L}_{GAN} 由於有兩組形成循環，因此就有兩個損失函數。以公式 (4.3) 為例，當生成網路 G 試圖生成與資料集 Y 相似之圖像 $G(x)$ 時，判別網路 D_Y 也試著將生成之圖像從原圖中區分出來。生成網路 G 希望通過優化減少與資料集 Y 之差別，以及判別網路 D_Y 也希望優化增強判別能力，另一組 F 與 D_X 也是如同 G 與 D_Y 。這兩組互相博弈之結構用公式表達如下：

$$\begin{aligned}\min_G \max_{D_Y} \mathcal{L}_{GAN}(G, D_Y, X, Y) \\ \min_F \max_{D_X} \mathcal{L}_{GAN}(F, D_X, Y, X)\end{aligned}\quad (4.5)$$



理論上對抗訓練可以學習到網路 G 與 F，並生成與資料集 Y 與 X 相似之分布之輸出。然而，當一個網路擁有足夠大之容量，輸入任何隨機排列之圖像，它都可以映射到與目標圖像相匹配之輸出分布。因此，無法單獨依靠公式 (4.5) 的對抗損失將每一個單獨輸入 x_i 映射到 y_i ，因此利用循環特徵找到其關係(見公式 (4.6))，可衍生出另一部分損失函數。

$$\begin{aligned} x &\rightarrow G(x) \rightarrow F(G(x)) \approx x \\ y &\rightarrow F(y) \rightarrow G(F(y)) \approx y \end{aligned} \tag{4.6}$$

其中：

$G(x)$ ：圖像 x 經過生成網路 G 形成 \hat{y} ， $\hat{\cdot}$ 為生成之符號。

$F(y)$ ：圖像 y 經過生成網路 F 形成 \hat{x} ， $\hat{\cdot}$ 為生成之符號。

- 循環一致性之損失 (Cycle Consistency Loss) 函數：

$$\begin{aligned} \mathcal{L}_{cyc}(G, F) &= \mathbb{E}_{x \sim p_{data}(x)} \left[\|F(G(x)) - x\|_1 \right] \\ &+ \mathbb{E}_{y \sim p_{data}(y)} \left[\|G(F(y)) - y\|_1 \right] \end{aligned} \tag{4.7}$$

為了進一步減少對抗網路能夠得到之空間大小，因此加入了循環一致性，如圖 4.6 [11] 所示。以圖 4.6 [11] 左圖為例，資料集 X 中的每一張圖像 x 在轉換過程中最後要回到原來之圖像，反之亦然 (如公式 (4.6) 所示)，即前向、後向循環一致，皆使用 L1 正則化。

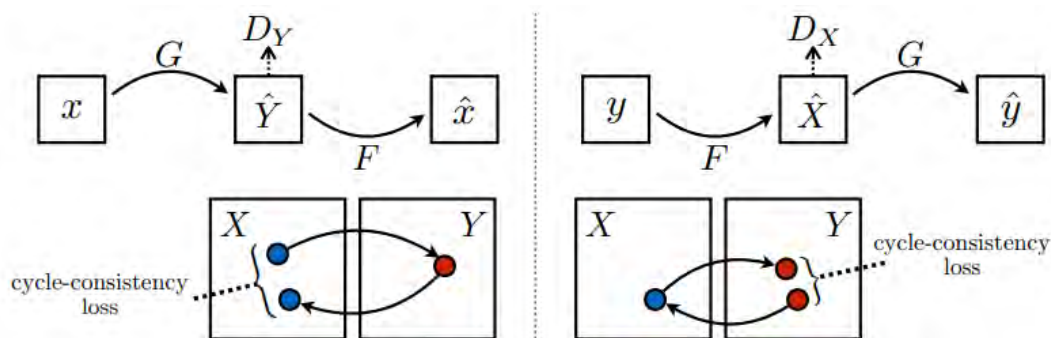


圖 4.6 循環一致性之示意圖[11]

最後，Jun-Yan Zhu 等人再多加一組使顏色成分保留之損失函數 $\mathcal{L}_{identity}$ 。由於生成網路 G 和 F 可以自由地改變輸入圖像之色調，可能循環一致性相同，但顏色與原圖像不符合。因此提供目標之資料集之圖像當作生成網路之輸入以對生成網路進行正則化，使得原本的 Cycle-GAN 更加準確，如圖 4.7 [11] 所示。從圖 4.7 [11] 可知，加入 $\mathcal{L}_{identity}$ 之 Cycle-GAN 與原圖對比，天空色彩較為相似；若無 $\mathcal{L}_{identity}$ 之 Cycle-GAN 與原圖對比，天空色彩有些微不同。以下為 $\mathcal{L}_{identity}$ 之數學式：

$$\begin{aligned} \mathcal{L}_{identity}(G, F) = & \mathbb{E}_{y \sim p_{data}(y)} \left[\|G(y) - y\|_1 \right] \\ & + \mathbb{E}_{x \sim p_{data}(x)} \left[\|F(x) - x\|_1 \right] \end{aligned} \quad (4.8)$$

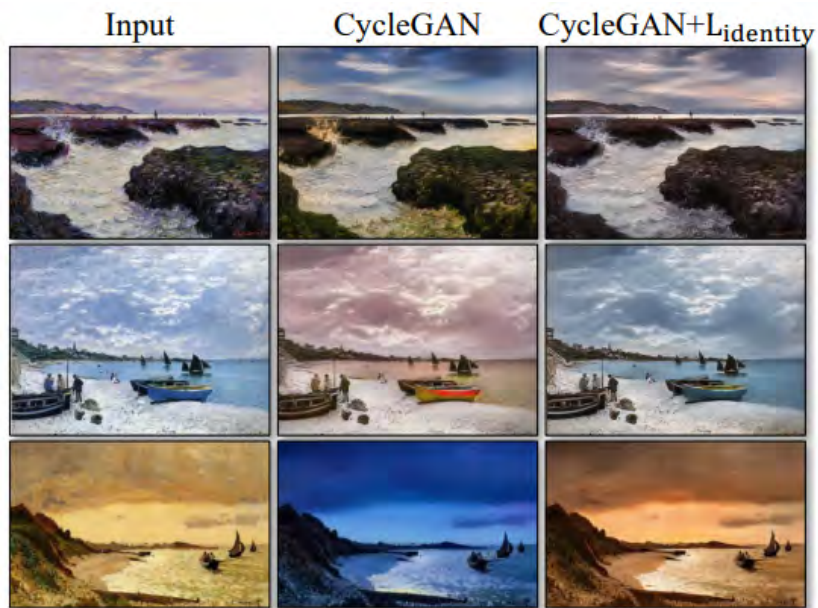


圖 4.7 Cycle-GAN 加入 $\mathcal{L}_{identity}$ 差異之示意圖[11]

第五章 腸腔辨識



磁控大腸膠囊內視鏡是一種可以進行大腸癌檢測、切除腫瘤等手術之診斷程序，其中以膠囊大小以並利用安裝在尖端上之微型鏡頭以獲取腸腔的畫面，如圖 5.1。而整個操作內視鏡的過程藉由外部的永久磁鐵以磁性控制之方式來控制其通過直腸並進入大腸等能夠在磁性控制下之區域。



圖 5.1 磁控大腸膠囊內視鏡尖端之示意圖

在本論文的研究中，改善腸腔辨識之目的是為了使操作者在操作本團隊研究之 MFN Platform 上減少像是一直對 MFN Platform 搖桿連續操作內視鏡擺姿等等的繁瑣操作，並利用 MFN Platform 能夠更準確地找到腸腔之方向。腸腔之檢測可以用於重新定位微型鏡頭並將鏡頭與腸道平行對齊。換句話說，當腸腔出現在畫面上，MFN Platform 之導航系統會對內視鏡進行轉向，使得腸腔會顯示在畫面之正中央。

5.1 YOLO

近期對於物體偵測之方法，像是 2014 年之 R-CNN (如圖 5.2 [13] 所示)，所使用候選區域之方法來生成圖像上之潛在邊界框(bounding boxes)，然後在這些邊界框上執行分類器去分類這些邊框所得出之種類。這套系統具有後處理方式可以優化邊界框、消除重複的邊界框，以及透過其他邊界框評估目標之邊界框 [13]。然而，由於評估及標籤的邊界框、分類邊界框之分類器、及一些參數是需要分開做訓練，因此在這些複雜的處理上就會顯得緩慢且在優化上難以最佳化。

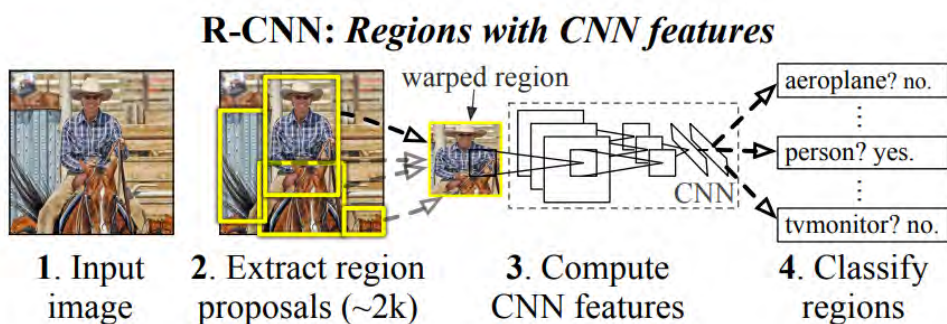
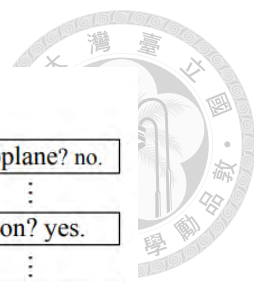


圖 5.2 R-CNN 之示意圖[13]

YOLO (You Only Look Once, 如圖 5.3 [14] 所示) 由 Redmon 等人 [14] 在 2015 年時發表並建立了一個開源碼的神經網路架構—Darknet [15]。Darknet 相比基於分類器的系統必須分開訓練的缺點，它使用了不同方法，將原本分散的目標偵測步驟融合，使其預測方式是藉由讀取整個圖像，並計算每個邊界框對於每一個類別的機率，並只需要單一神經網路進行評估，實現 end-to-end 之訓練，使其運算速度大幅度提升。

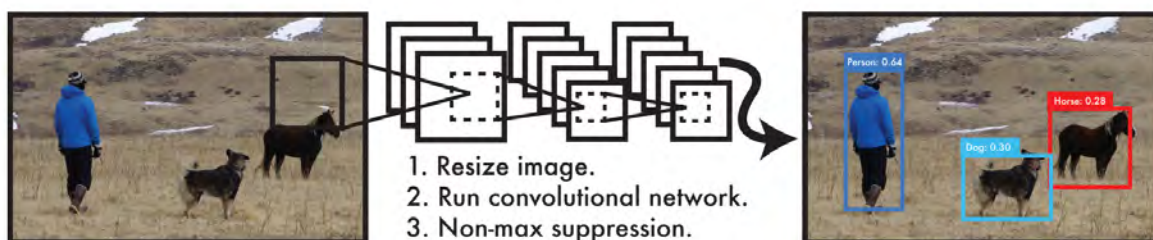


圖 5.3 YOLO 偵測系統[14]

從圖 5.3 [14] 可知，YOLO 會先將輸入圖像進行縮放，並將縮放圖像輸入至多層之卷積神經網路 (Convolutional Neural Network, CNN)，透過其訓練與偵測，並使用非極大值抑制 (Non-max suppression, NMS) 去做篩選。最後，再處理預測結果並得到坐標與類別都顯示至輸入圖像之中後成為輸出圖像。一般的 YOLO 版本可以在單一顆 Titan X GPU 上達到 45 fps (frame per second)。對於本團隊研究中，輸入影片可以達到即時腸腔辨識的效果。以下將一一詳細說明 YOLO 之原理。



5.2 YOLO 演算法

5.2.1 網格單元 (Grid Cell)

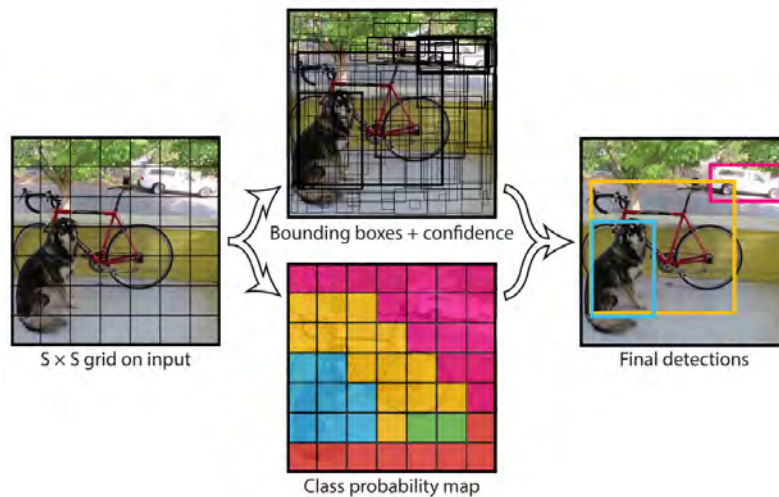


圖 5.4 YOLO 處理網格之過程[14]

YOLO 將輸入圖像切割成 $S \times S$ 的網格(grid)，如圖 5.4 [14] 所示，如果網格中間有物體則該網格會負責去偵測該物體。單元格會預測出 B 個邊界框與邊界框之置信度 (box confidence scores)，其分數越高就代表格子中越有物體存在，以下為其數學式：

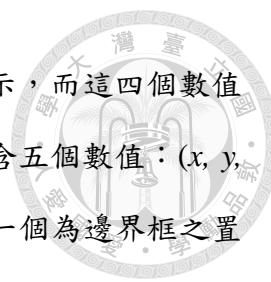
$$\text{box conf. score} = \text{Pr}(\text{Object}) * IOU_{pred}^{\text{truth}} \quad (5.1)$$

其中

$\text{Pr}(\text{Object})$ ：邊界框含有物體之機率：若邊界框裡不含物體， $\text{Pr}(\text{Object}) = 0$ ；反之則為 $\text{Pr}(\text{Object}) = 1$ 。

$IOU_{pred}^{\text{truth}}$ ：邊界框之準確度，預測框與真實實際框的交並比 (Intersection Over Union, IOU) 來表示。

邊界框可以用 4 個數值表示： (x, y, w, h) ，其中 (x, y) 為邊界框之中心坐標，而 (w, h) 為邊界框之寬與高。中心坐標為相對於每個格子左上角坐標點之偏移值，



且寬與高為相對於整個圖像之寬與高的比例，如圖 5.5 [16] 所示，而這四個數值大小範圍就落在 $[0, 1]$ 之間。因此，邊界框的預測值實際上就包含五個數值： $(x, y, w, h, box\ conf. score)$ ，其中前四個為邊界框大小及位置，而最後一個為邊界框之置信度。

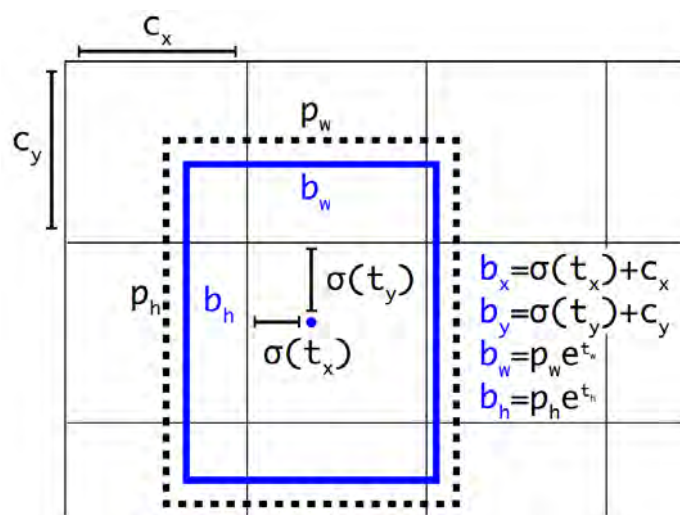


圖 5.5 邊界框位置標示示意圖[16]

前面處理完邊界框之問題，還有分類問題。對於每個單元格還要預測出 C 個類別之機率，在預測出邊界框之置信度時也同時計算出各個邊界框類別之置信度 (class-specific confidence scores)：

$$\Pr(class_i | object) * \Pr(object) * IOU_{pred}^{truth} = \Pr(class_i) * IOU_{pred}^{truth} \quad (5.2)$$

其中

$\Pr(class_i | object)$ ：物體有在該類別出現之機率 (conditional class probabilities)

$\Pr(object) * IOU_{pred}^{truth}$ ：邊界框之置信度，如公式(5.1)所示。

因此，每個邊界框皆有類別之置信度，並且藉由類別之置信度來過濾網格的預測框。



每個網格需要預測 $(B*5 + C)$ 個值，而每個邊界框有 5 個數值： $(x, y, w, h, box\ conf.\ score)$ ，最終預測值為 $S \times S \times (B*5 + C)$ 大小之張量。Redmon 等人在 PACAL Visual Object Classes (VOC) 中驗證第一代 YOLO 時，所設立之參數： $S = 7, B = 2$ ，在 PACAL VOC 中有 20 個類別， $C = 20$ ，使得最終得到之張量為 $7 \times 7 \times (2 * 5 + 20) = 7 \times 7 \times 30$ ，而這也是模型最後所需要輸出之結果。

5.2.2 YOLO 神經網路架構

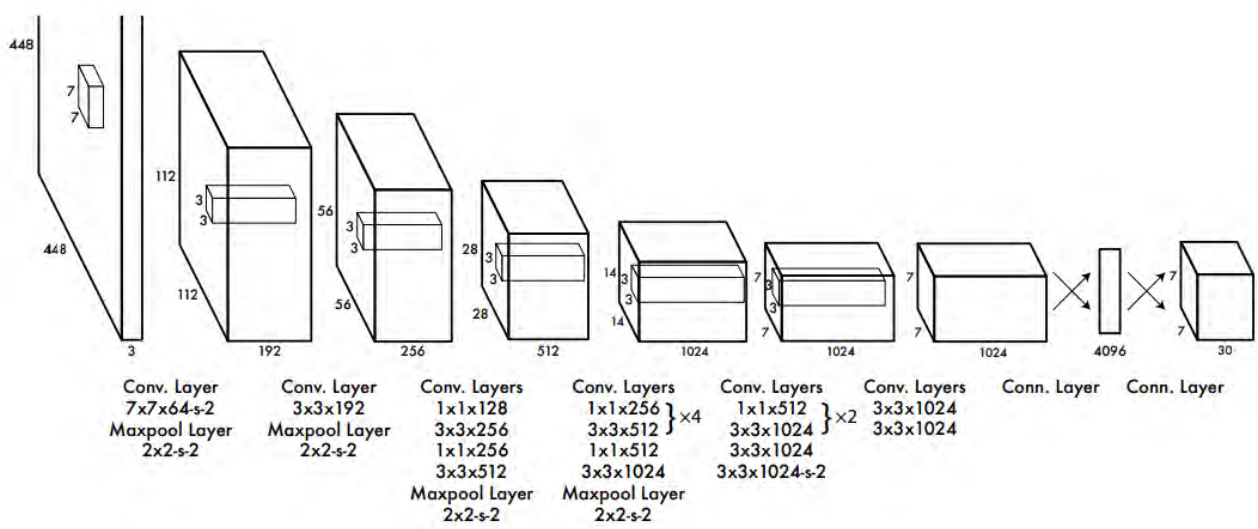


圖 5.6 YOLO CNN 模型結構[14]

YOLO 有 24 層卷積層 (Convolutional Layers) 和 2 層全連接層 (Fully Connected Layers) 所組成之神經網路，如圖 5.6 [14] 所示，最終輸出為 $7 \times 7 \times 30$ 之張量大小。在 CNN 中最後一層使用之激勵函數為線性激勵函數 (Linear Activation Function)，其餘的激勵函數則是使用 Leaky ReLU，該數學式表示如下：

$$\phi(x) = \begin{cases} x, & x > 0 \\ 0.1x, & otherwise \end{cases} \quad (5.3)$$

5.2.3 非極大值抑制 (Non-max suppression, NMS)

在圖像的物體偵測中，都會先預測出含有物體的候選邊界框是否有物體。然而，在選出過程中會遇到單個物體被多個候選之邊界框所標記，如圖 5.7 所示。

因此，在這裡需要用到非極大值抑制 (Non-max suppression, NMS)，此方法透過每個邊界框之置信度來做篩選，可以有效地消除多餘的邊界框，並找到最佳預測之邊界框。最後，對最佳之邊界框進行邊界框類別之置信度的判斷，判斷出最佳之邊界框的分類。



圖 5.7 非極大抑制之示意圖

5.2.4 損失函數 (Loss Function)

YOLO 在預測之每個單元格皆預測多個邊界框但其相對應之類別只有一個。如果該單元格內確實有物體存在，就只選擇與真實實際框中具有最大 IOU 之邊界框負責與該物體，而其它邊界框則設為不存在物體。這樣的設置方式是為了使每個單元框所對應之邊界框能夠更加專業化，以便其可適用不同大小，不同長寬比之物體。YOLO 之損失函數 [14] 皆使用平方和誤差公式 (Sum-squared Error)，以下為損失函數之表示式，符號上面有 $\hat{\cdot}$ 為神經網路之預測值：

$$\begin{aligned}
 & \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{obj} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
 & + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
 & + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{obj} (C_i - \hat{C}_i)^2 \\
 & + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{noobj} (C_i - \hat{C}_i)^2 \\
 & + \sum_{i=0}^{S^2} \mathbf{1}_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2
 \end{aligned} \tag{5.4}$$



從公式 (5.4) 可知，損失函數可由四個部分所組成：

- 預測邊界框之中心點 (x, y) 的損失：

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{obj} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \quad (5.5)$$

其中

$\mathbf{1}_{ij}^{obj}$ ：控制函數，如果在第 i 個單元格中第 j 個邊界框裡出現物體則為

$\mathbf{1}_{ij}^{obj} = 1$ ；若該處無物體則為 $\mathbf{1}_{ij}^{obj} = 0$ 。

λ_{coord} ：增加含有物體時的損失之權重係數。

因此中心點之損失只對有包含真實物體所屬之單元格中的邊界框進行損失計算，若無物體則預測函數不對損失函數造成影響。 (x, y) 之值與預測值之間採用簡單的平方和誤差進行計算。

- 預測邊界框寬和高 (w, h) 的損失：

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \quad (5.6)$$

其中

$\mathbf{1}_{ij}^{obj}$ ：控制函數，如果在第 i 個單元格中第 j 個邊界框裡出現物體則為

$\mathbf{1}_{ij}^{obj} = 1$ ，也是使得只有真實物體之部分進行損失計算。

λ_{coord} ：增加含有物體時的損失之權重係數。

在這邊對 (w, h) 處理分別取根號，在於不取根號會使得損失函數更傾向於調整尺寸較大之預測框。例如，20 個像素之偏差對於 $800*600$ 之預測框沒有影響，



但如果是 30*40 之預測框影響較大。

- 置信度 (C) 的損失：

$$\sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbf{1}_{ij}^{noobj} (C_i - \hat{C}_i)^2 \quad (5.7)$$

其中

$\mathbf{1}_{ij}^{obj}$ ：控制函數，如果在第 i 個單元格中第 j 個邊界框裡出現物體則為

$\mathbf{1}_{ij}^{obj} = 1$ 。

$\mathbf{1}_{ij}^{noobj}$ ：相反之控制函數，如果在第 i 個單元格中第 j 個邊界框有物體則

為 $\mathbf{1}_{ij}^{noobj} = 0$ ；反之為 1。

C：邊界框之置信度，透過式子(5.1)，其分數數值在有物體時為 IOU 數值，無物體則為 0。

λ_{noobj} ：降低不含有物體時的損失之權重係數。

- 物體類別機率 (p) 的損失：

$$\sum_{i=0}^{S^2} \mathbf{1}_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2 \quad (5.8)$$

其中

$\mathbf{1}_i^{obj}$ ：控制函數，如果在第 i 個單元格中出現物體則為 $\mathbf{1}_i^{obj} = 1$ 。

$p_i(c)$ ：在第 i 個單元格中物體為 c 之機率。

YOLO 面對物體檢測問題，是一個典型的類別數目不均衡之問題。其中 7*7 = 49 格單元格中，含有物體之單元格可能只有 3、4 格，其餘是不含有物體之單



元格。此時不採取策略的話會造成 mAP (mean Average Precision) 不會太高，原因是模型更傾向於不含有物體之單元格。 λ_{coord} 、 λ_{noobj} 之作用讓含有物體之網格在損失函數中的權重更大，使得模型更加重視含有物體之網格所造成之損失。在預設上， λ_{coord} 與 λ_{noobj} 分別為 5 與 0.5。

5.3 YOLOv3

YOLOv3 [16] 在 2018 年出世，是一個用於物體偵測之卷積神經網路，其藉由 Darknet-53 (總共 106 層之神經網路，其中包含 53 層卷積層) 來達成判別物體特徵之圖像分類，如圖 5.8 [16] 所示。

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1x	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2x	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8x	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8x	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4x	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

圖 5.8 Darknet-53 結構之示意圖[16]

YOLOv3 之神經網路在前 74 層共進行了五次的下採樣，75 層以後共進行 3 種尺度之輸出。在第 82 層進行第一次輸出最小尺度之 YOLO 層 (13*13)，將原本輸入之圖像：416*416*3 轉換成 13*13*255 (因 B 設為 3 且使用 COCO 之數據類

別 C 共 80 種類別) 之張量，其步長為 32 的步幅；在第 94 層中第二次輸出中型尺度之 YOLO 層 (26*26)，將第 79 層之輸出透過卷積後加入第 61 層輸出，並透過卷積形成 26*26*255 之張量，其步長為 16 的步幅；最後在第 106 層中輸出最大尺度之 YOLO 層 (52*52)，將第 91 層之輸出透過卷積後加入第 36 層輸出，並透過卷積形成 52*52*255 之張量，其步長為 8 的步幅，上述之過程透過圖 5.9 來簡單表示。在過程中加入前期處理之輸出併入到後續卷積之輸出的方式是為了防止通過多次卷積而失去有效特徵，其方式為區域性特徵融合。

YOLO 與 YOLOv3 之輸出部分有些許不同：YOLO 共有 $S \times S \times (B * 5 + C)$ 之張量；而 YOLOv3 將 B 個邊界框之數值部分除了有 $(x, y, w, h, box\ conf.\ score)$ 這五個數值之外，還包含 C 個類別，故總共有 $S \times S \times [B * (5 + C)]$ 之張量。

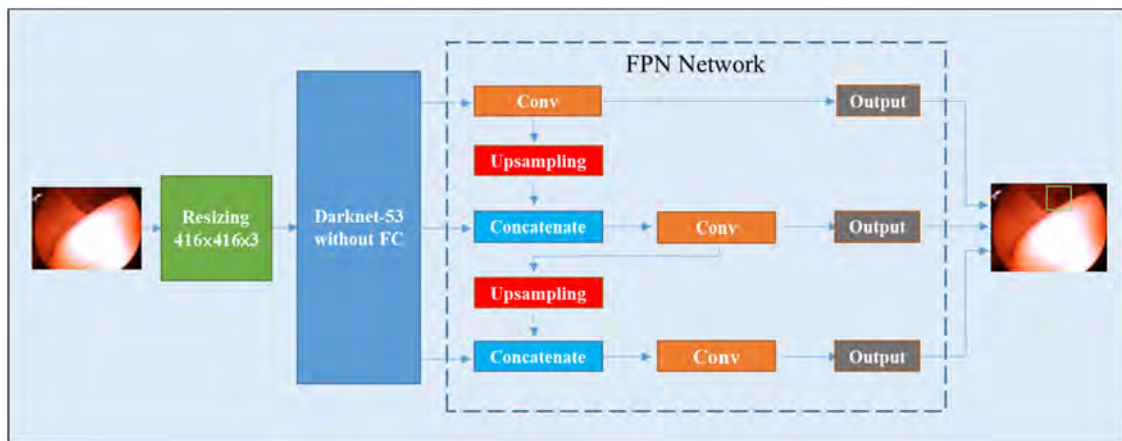


圖 5.9 YOLOv3 偵測腸腔之示意圖

第六章 實驗架構與成果討論



6.1 硬體與架構

整個實驗皆使用 Python 語言，並透過 Pytorch 架構建立深度學習之神經網路。在整個實驗過程中，硬體部分使用桌上型電腦，其配置有 NVIDIA GeForce GTX1080Ti GPU。使用的作業系統為 Windows 10。

6.2 資料集

在整個實驗中，KVASIR 資料集不僅是進行資料擴增，還進行 Cycle-GAN 之實驗。KVASIR 資料集是從大腸鏡檢查是視頻中擷取畫面之資料集，包括腸道中的各種位置。在 YOLOv3 進行腸腔辨識中選擇了 2680 張圖像當作實驗用之資料集，其中 2144 張圖像作為訓練用，536 張圖像作為驗證用。在 Cycle-GAN 訓練所使用模擬腸道之圖像有 1015 張，真實腸道為 KVASIR 資料集之圖像共 1200 張圖像。

6.3 資料處理

LabelImg (如圖 6.1 所示) 用於標記圖像中之腸腔位置，其可以產生出符合 YOLO 格式所需要之位置文件。

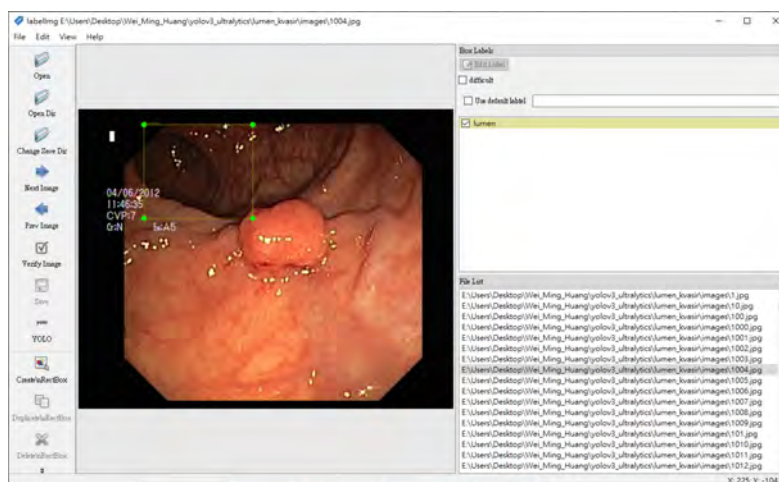


圖 6.1 LabelImg 之介面



6.4 性能指標

為了評估物體偵測之模型在測試資料集中的表現如何，需要用一些指標去判斷模型的好壞，以作為選擇模型的依據。這邊使用分類問題中常使用之指標—混淆矩陣 (Confusion Matrix)，它可以衍生出很多不同的指標。混淆矩陣包含四個參數：真陽 (True Positive, TP)、真陰 (True Negative, TN)、偽陽 (False Positive, FP) 及偽陰 (False Negative, FN)，見表 6.1。

表 6.1 混淆矩陣 (Confusion Matrix)

Confusion Matrix		Predict	
		Yes	No
Real	Yes	TP	FN
	No	FP	TN

其中

TP：實際為正例且被模型劃分為正例之樣本數

FP：實際為反例但被模型劃分為正例之樣本數

FN：實際為正例但被模型劃分為反例之樣本數

TN：實際為反例且被模型劃分為反例之樣本數

混淆矩陣之參數可以衍生出多種指標，在本研究所使用之指標有：精確率 (Precision)、召回率 (Recall)、F1 分數 (F1-score)、平均精度均值 (mean Average Precision, mAP)、及本研究使用 Precision-Recall 曲線下之面積 (the Area Under the Curve of P-R Curve, AUC of P-R Curve)。利用這些指標，以判斷擴增 KVASIR



資料集對 YOLO 模型是否有提升性能之效果。

Precision 表示模型預測時預測為正例之樣本數中有多少是真正的實際正例，主要針對預測結果。預測為正例之樣本數的可能性有兩種：一種是將正例預測成正例之 TP；另一種為將反例預測成正例之 FP，以下為 Precision 之數學式：

$$precision = \frac{TP}{TP + FP} \quad (6.1)$$

Recall，又稱為靈敏度 (Sensitivity)，表示樣本中之實際正例有多少是被預測正確，主要針對實際樣本。實際正例樣本也有兩種可能：一種是將正例預測成正例之 TP；另一種為將正例預測成反例之 FN，以下為 Recall 之數學式：

$$recall = \frac{TP}{TP + FN} \quad (6.2)$$

在最佳情況下，Precision 和 Recall 會擁有很高的數值。然而，兩者之間之數值之間具有一定的關係，造成高 Precision 會導致 Recall 降低，反之亦然。因此，有一種評估指標期望 Precision 和 Recall 都同時兼顧，其指標名為 F1-score。F1-score 從 F_β -score 推算出來， F_β -score 主要將 Precision 和 Recall 合併為一個值，以下為 F_β -score 之數學式：

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall} \quad (6.3)$$

當公式 (6.3) 中的 $\beta = 1$ 時，就是 F1-score，基本上是模型預測正確之機率。F1-score 是 Precision 和 Recall 兩種數值之平均值。進一步來說，為 Precision 和 Recall 的調和平均數，以下為其數學式：

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (6.4)$$

當網路模型生成預測之邊界框，不一定會百分之百與真實實際之邊界框符合，因此將其與真實實際之邊界框形成一定面積之重疊區域以找到一個 TP。而這個重疊區域之大小是由交並比 (Intersection Over Union, IOU) (如圖 6.2 所示) 來



當作判斷標準。一般 IOU 之設定值都設定為 50%，高於這個值，則表示網路模型偵測到一個物體。

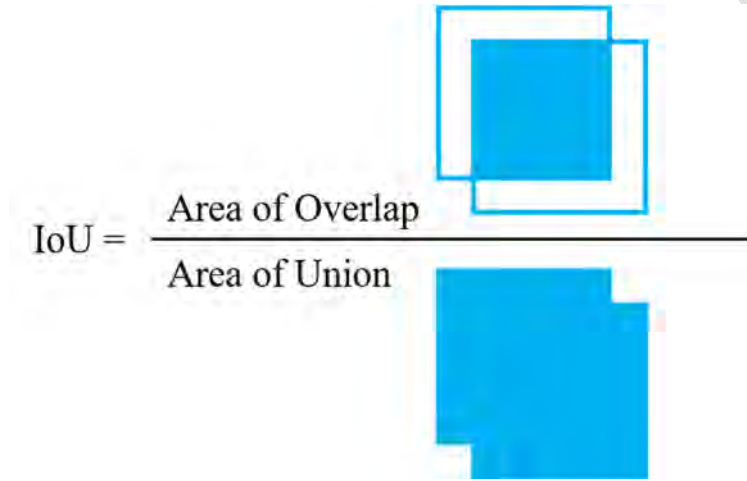


圖 6.2 Intersection Over Union

為了解決 Precision、Recall 及 F1-score 的單點值侷限性，欲得到一個能夠反映全局性能之指標。因此透過 Precision 和 Recall 之應用，增加新的判斷指標。mAP (mean Average Precision) 是預測物體位置及類別之性能度量標準，為取多個類別之 AP (Average Precision) [17] 進行平均。針對算法繪製 Precision-Recall 曲線，並將 11 個等距之 Recall 全部計算後曲平均為每一類之 AP，以下為 AP 之數學式：

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, 0.2, \dots, 1\}} p_{interp}(r) \quad (6.5)$$


$$p_{interp}(r) = \max_{\tilde{r}: \tilde{r} \geq r} p(\tilde{r}) \quad (6.6)$$

其中

r : Recall 之階段， $[0, 0.1, 0.2, \dots, 1]$ 共 11 個階段。

$p_{interp}(r)$: 採用相應 Recall 超過 r 之方法測得最大 Precision。

$p(\tilde{r})$: 在 \tilde{r} 之 Recall 所測得之 Precision。



AUC of P-R Curve 是利用上述 AP 的方式繪製出 Precision-Recall 曲線，並計算這條曲線下與 x 座標軸和 y 座標軸所圍成的面積。當面積值越來越接近 1，也就是曲線彎曲的程度越來越靠近左上角時，表示物體偵測之模型在測試資料集的表现越來越好。

由網路模型所預測之邊界框並不是絕對的。每一個預測之邊界框都有各自的概率或置信度，以衡量網路模型所預測之邊界框的確定性。網路模型通過選擇最小置信度當作閾值，使得預測的總數減少。而藉由預測的總數減少，使錯誤的偵測數量降低，可以使網路模型的 Precision 增加。然而，選擇較高的閾值也意味著 Recall 會跟著降低。因此，需要選擇適當的閾值以兼顧平衡，並獲得好的 Precision 跟高的 Recall。



6.5 實驗結果

6.5.1 資料擴增

本研究進行五種資料擴增：圖像平移、圖像旋轉、圖像縮放、圖像加入高斯雜訊與圖像加入動態模糊，而其中一部分擴增方式參考 [3] 來設定範圍。每一種擴增方式將擴增 KVASIR 資料集之數量為 2680 張圖像，最終共計 13400 張擴增圖像並加入至訓練集，如表 6.2 所示。以下為五種擴增方式之設置，其結果如圖 6.3 所示，圖 6.3 (a) 為原始圖像。

幾何變換類：

- 圖像平移：x 軸方向向左或向右隨機移動 10 個像素，y 軸方向向上或向下隨機移動 10 個像素，如圖 6.3 (b) 所示，向右移動 10 個像素。
- 圖像旋轉：六種旋轉角度任選一種進行旋轉，有 -15° 、 -10° 、 -5° 、 $+5^\circ$ 、 $+10^\circ$ 和 $+15^\circ$ ，共六種，其中 - 為逆時針旋轉，+ 為順時針旋轉，如圖 6.3 (c) 所示，進行順時針旋轉 10° 。
- 圖像縮放：八種縮放方式任選一種進行縮放，有 0.8、0.85、0.9、0.95、1.05、1.1、1.15 與 1.2，共八種，其中小於 1 為縮小，大於 1 為放大，如圖 6.3 (d) 所示。

顏色變換類：

- 圖像加入高斯雜訊：將高斯分佈之 μ 設定為 0， σ 設定為 $255 * 0.2 = 51$ ，如圖 6.3 (e) 所示。
- 圖像加入動態模糊：將模糊程度之比例 k 數值越高，其模糊程度越高。將 k 調整至 50，達到模糊之效果，且有八個方位進行動態模糊，有 0° 、 45° 、 90° 、 135° 、 180° 、 225° 、 270° 與 315° ，共八個，如圖 6.3 (f) 所示。

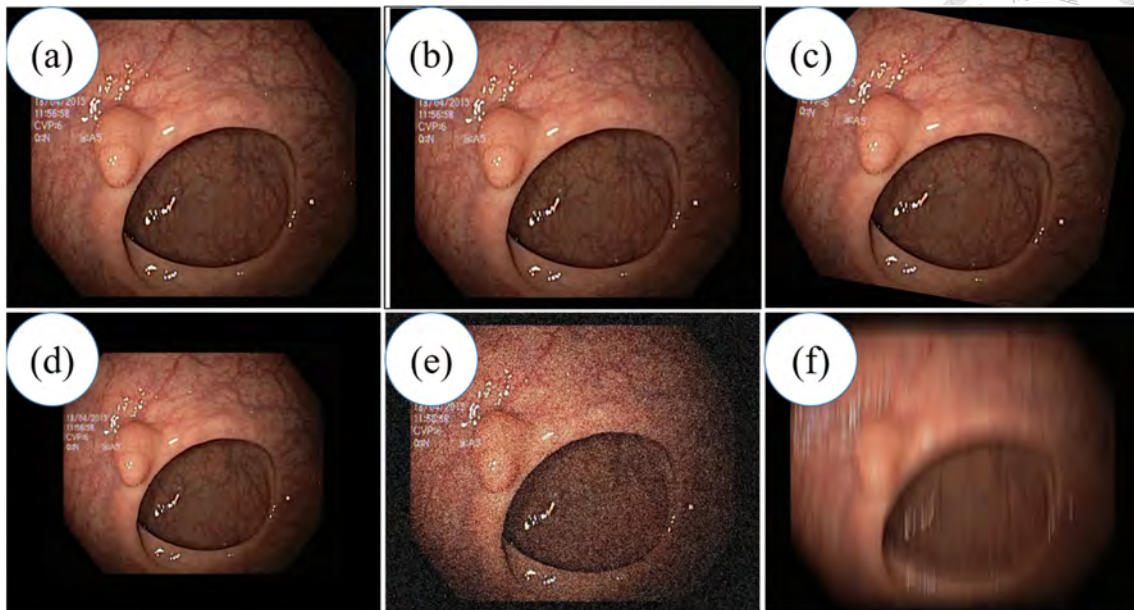


圖 6.3 傳統資料擴增之圖像

表 6.2 傳統資料擴增生成之數量

	擴增新圖片之數量
資料擴增生成之訓練集	15544
原始訓練集	2144
圖像平移	2680
圖像旋轉	2680
圖像縮放	2680
高斯雜訊	2680
動態模糊	2680
原始驗證集	536
總計	16080

一開始先對每一種擴增方式進行訓練，每一種擴增方式之訓練集包含原始訓練集與各個擴增方式之資料集，例如在進行圖像平移之訓練，原始訓練集有 2144 張，而圖像平移生成 2680 張，訓練集總共使用 $2144 + 2680 = 4824$ 張圖像，驗證

方式為使用原始驗證集 536 張圖像進行。在每一個訓練，模型的參數都是由隨機梯度下降法 (stochastic gradient descent, SGD) 進行優化訓練損失。如果有 $\text{IOU} > 0.5$ ，從 SGD 優化器生成的二元分類之結果將其歸類為腸腔。訓練過程中訓練 200 Epoches、batch size 為 8，且學習率皆為 0.001。對原始資料集與每一種擴增方式進行 YOLOv3 訓練，其訓練過程如圖 6.4、圖 6.5、圖 6.6、圖 6.7、圖 6.8，與圖 6.9 所示。測試結果之結果如表 6.3 與圖 6.10 所示，每種擴增方式之 PR-Curve 如圖 6.11 所示。

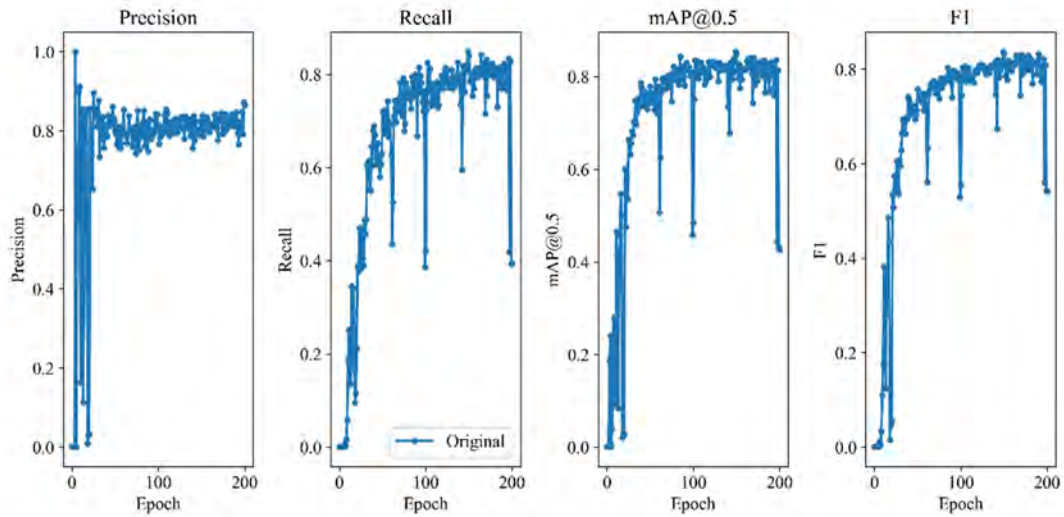


圖 6.4 原始資料集訓練過程之性能

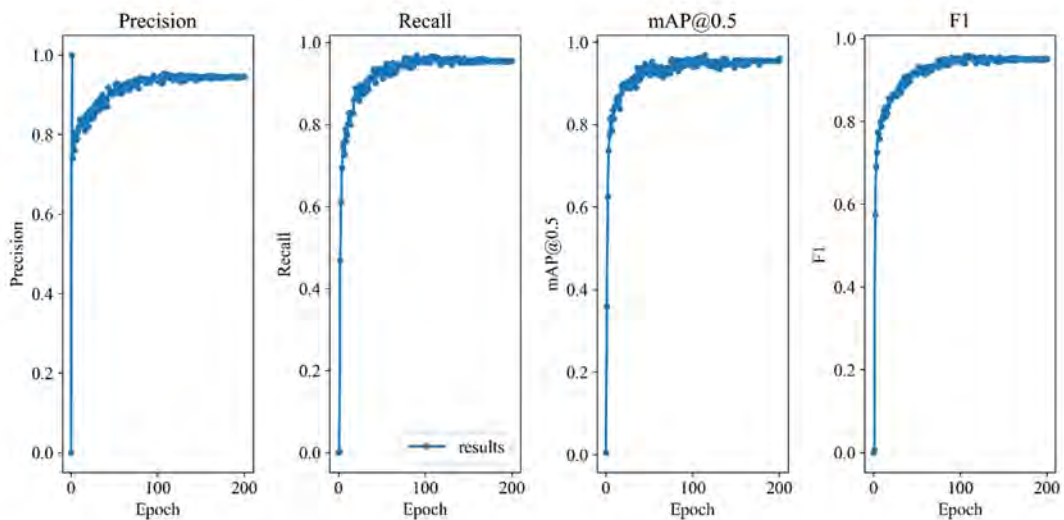


圖 6.5 圖像平移訓練過程之性能

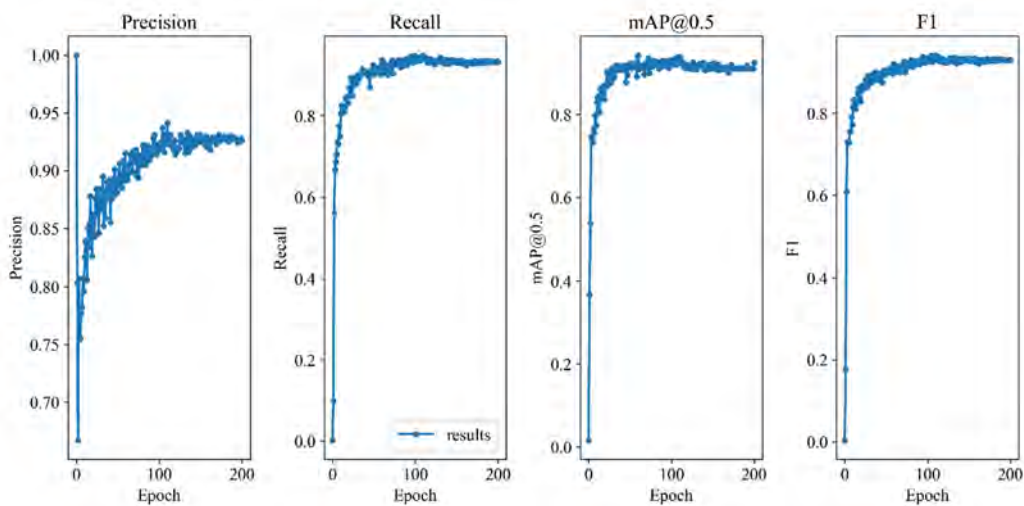


圖 6.6 圖像旋轉訓練過程之性能

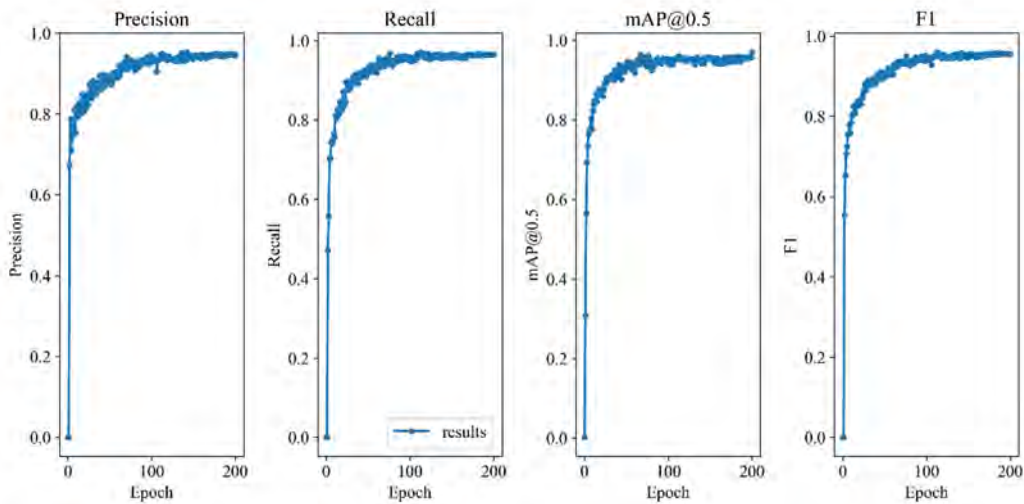


圖 6.7 圖像縮放訓練過程之性能

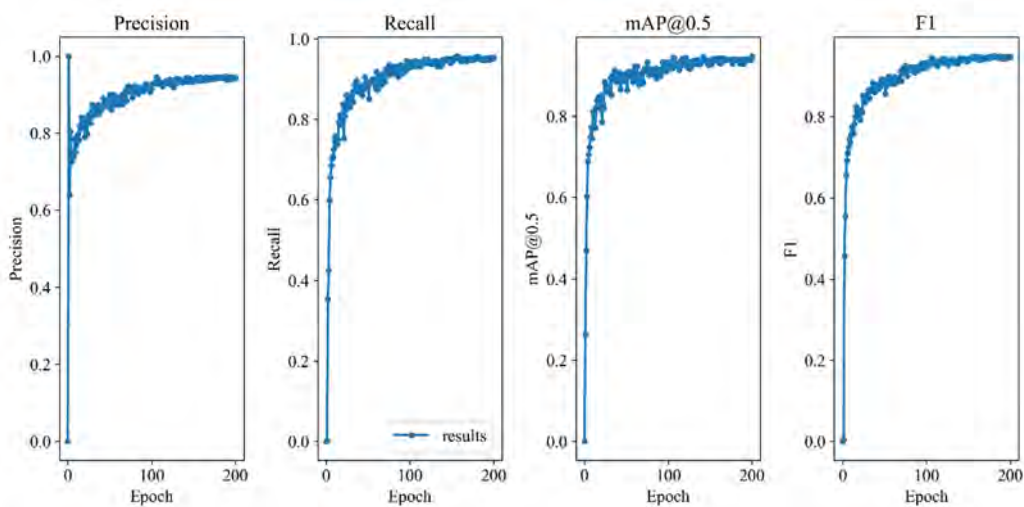


圖 6.8 加入高斯雜訊訓練過程之性能

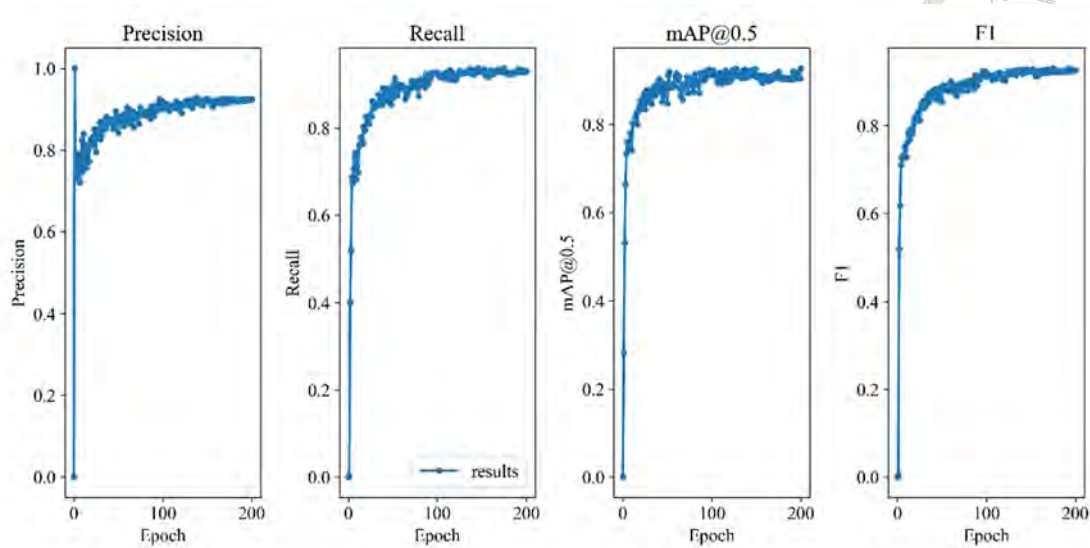


圖 6.9 加入動態模糊訓練過程之性能

表 6.3 傳統資料擴增與原始資料集之結果比較

	Precision	Recall	mAP@0.5	F1	AUC
Origin	0.764	0.86	0.857	0.809	0.854
Translation	0.942	0.965	0.973	0.953	0.972
Rotation	0.87	0.939	0.946	0.904	0.946
Scaling	0.946	0.965	0.968	0.955	0.969
Gaussian Noise	0.944	0.953	0.952	0.949	0.951
Motion Blur	0.922	0.931	0.926	0.927	0.921

傳統資料擴增與原始資料集之結果長條圖

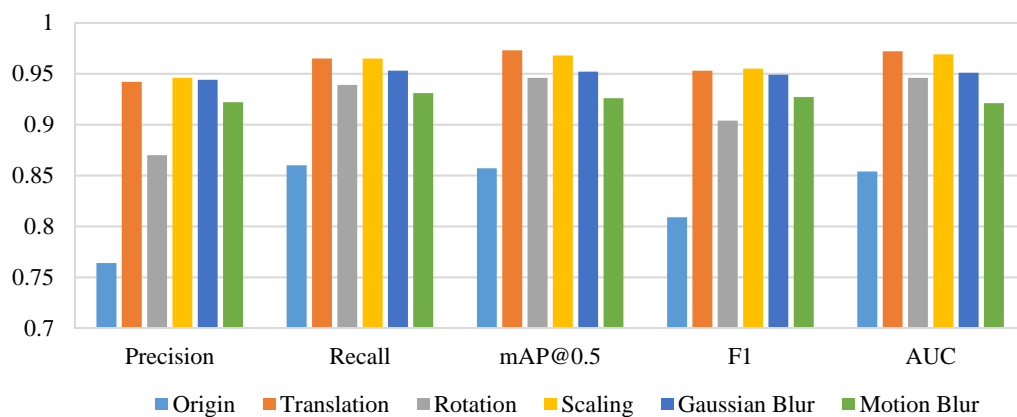


圖 6.10 傳統資料擴增與原始資料集之結果長條圖

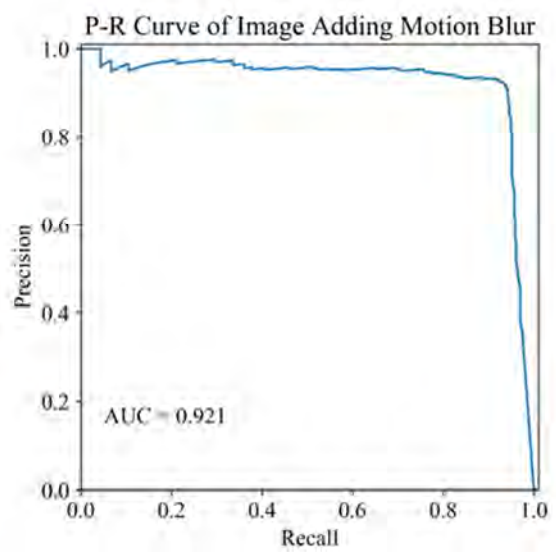
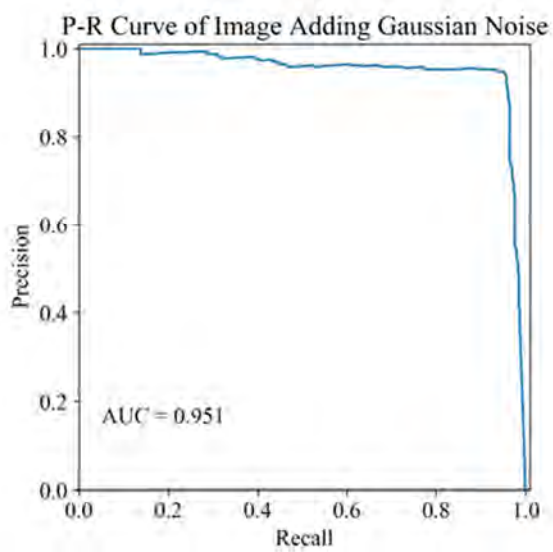
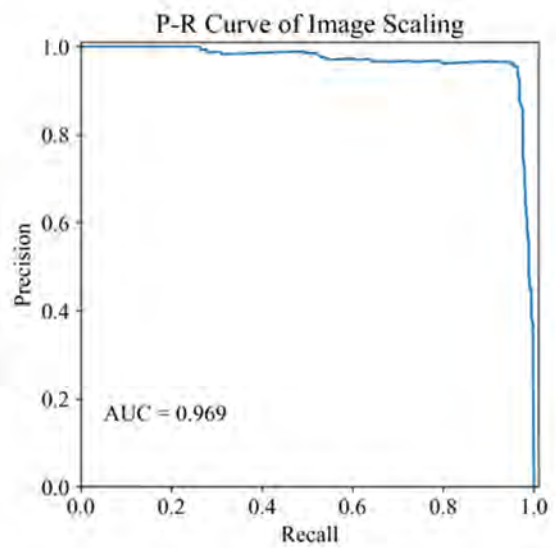
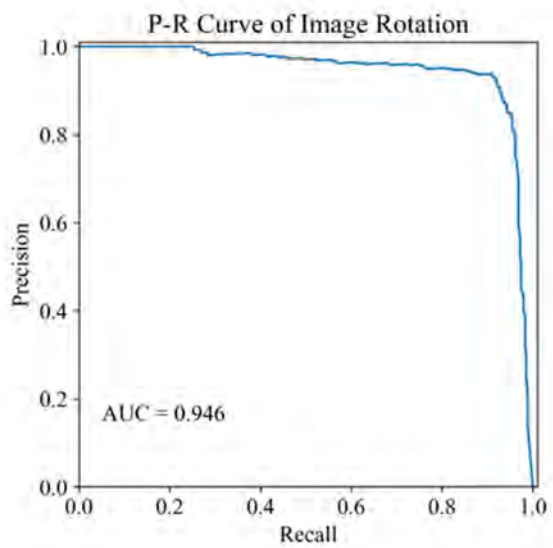
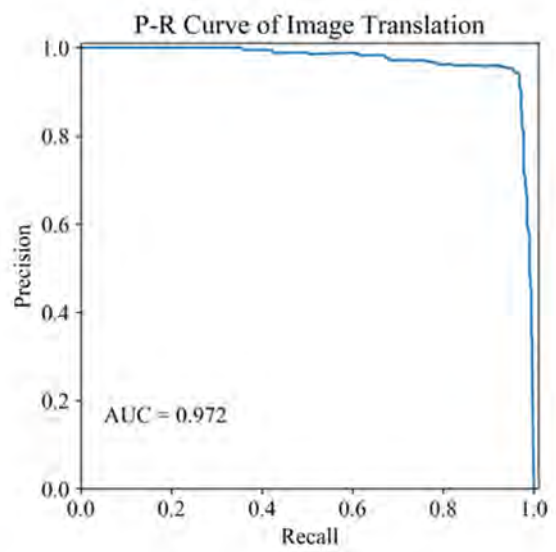
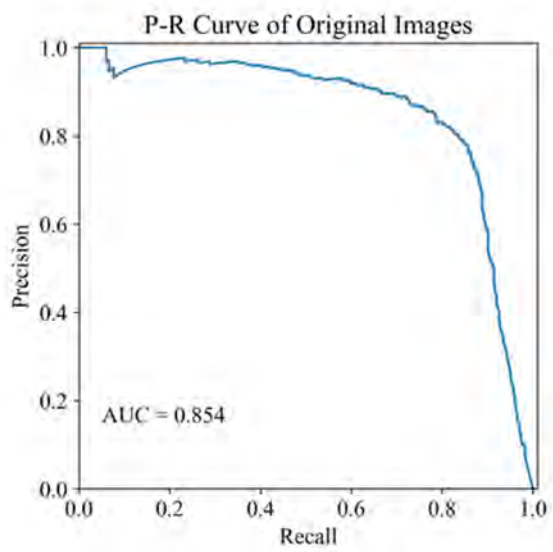


圖 6.11 傳統擴增方式之 PR-Curve

除了訓練每一種擴增方式，亦進行分類成幾何變換與顏色變換之訓練，如表 6.4 所示。訓練之訓練參數如同章節 6.5.1 之設置方式，以進行 YOLOv3 訓練。訓練過程如圖 6.12、圖 6.13 所示。測試之結果與 PR-Curve 如表 6.5、圖 6.14 與圖 6.15 所示。從表 6.5、圖 6.14 與圖 6.15 可知，幾何變換類與顏色變換比原始資料集之性能較為優異。

表 6.4 傳統資料擴增以變換分類之資料集數量

擴增新圖片之數量	
幾何變換之資料集	10184
原始訓練集	2144
圖像平移	2680
圖像旋轉	2680
圖像縮放	2680
原始驗證集	536
顏色變換之資料集	7504
原始訓練集	2144
高斯雜訊	2680
動態模糊	2680
原始驗證集	536

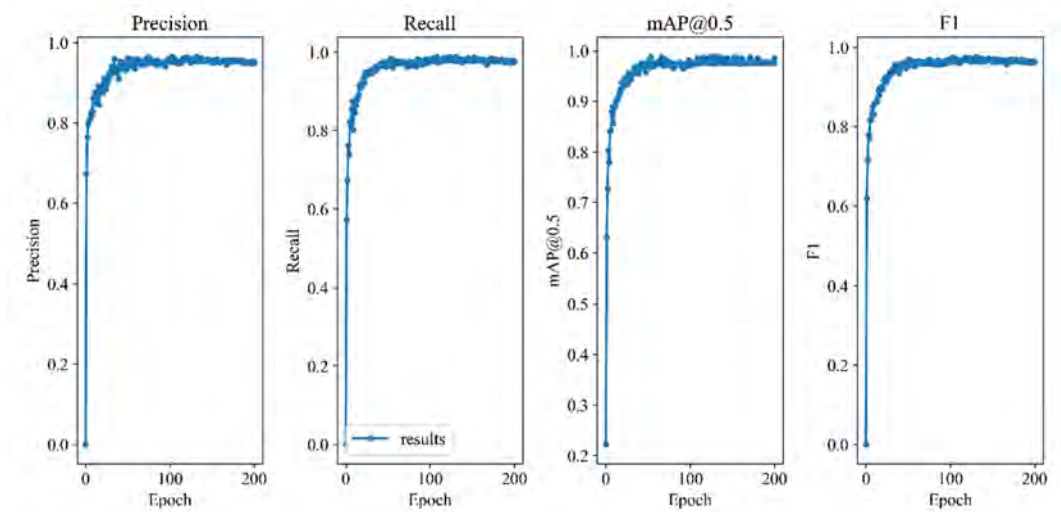


圖 6.12 幾何變換類擴增的訓練過程之性能

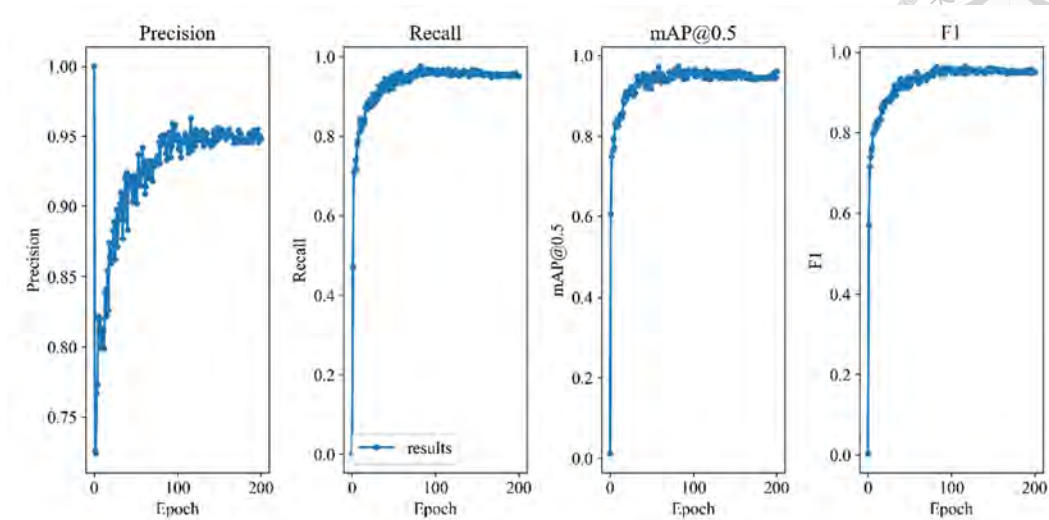


圖 6.13 顏色變換類擴增的訓練過程之性能

表 6.5 幾何變換類、顏色變換類與原始資料集之結果比較

	Precision	Recall	mAP@0.5	F1	AUC
原始資料集	0.764	0.86	0.857	0.809	0.854
幾何變換類	0.958	0.993	0.989	0.975	0.993
顏色變換類	0.943	0.974	0.976	0.958	0.977

幾何變換類、顏色變換類與原始資料集之結果長條圖

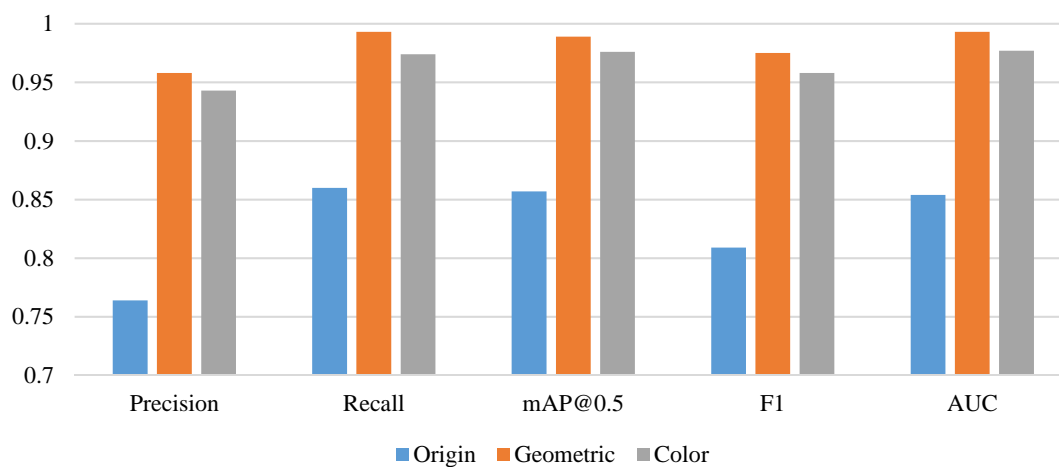


圖 6.14 幾何變換類、顏色變換類與原始資料集之結果長條圖

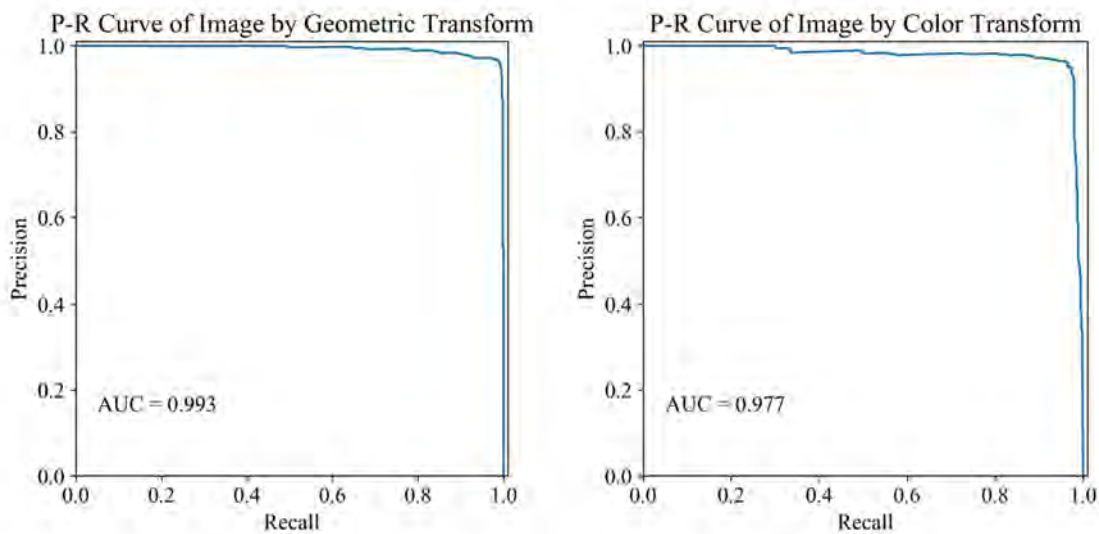


圖 6.15 幾何變換與顏色變換之 PR-Curve

最後將所有傳統資料擴增方式 (如表 6.2 所示) 進行 YOLOv3 訓練，訓練之參數如同章節 6.5.1 之設置方式。訓練過程如圖 6.16 所示，測試結果如表 6.6 與圖 6.17 所示，其結果比原始資料集相比性能較為優異，傳統資料擴增之 PR-Curve 如圖 6.18 右圖所示。

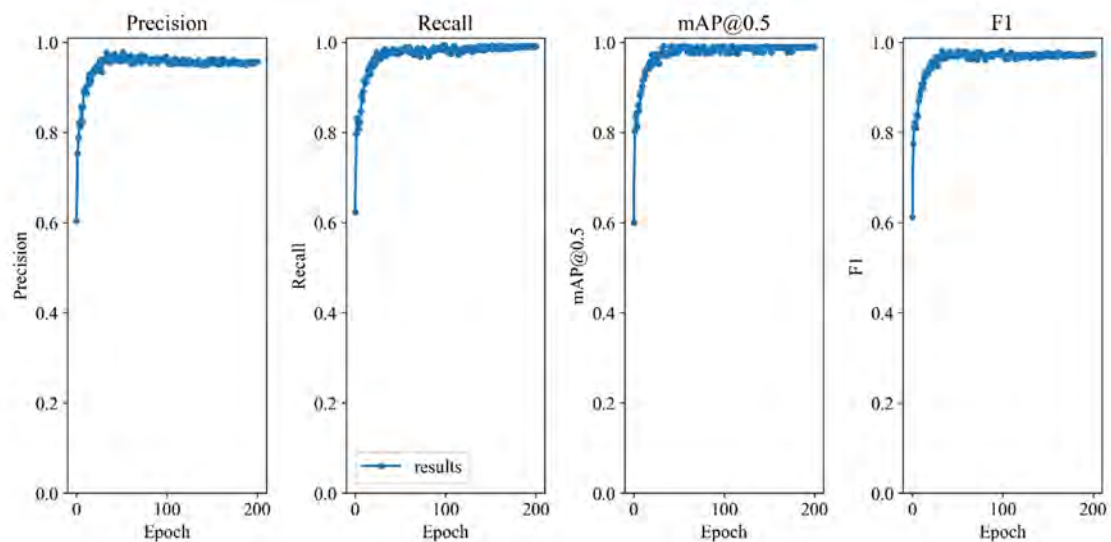


圖 6.16 傳統資料擴增的訓練過程之性能

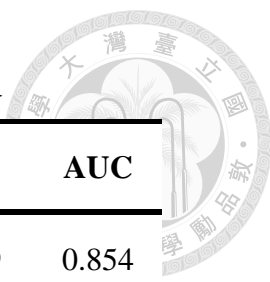


表 6.6 傳統資料擴增與原始資料集之結果比較

	Precision	Recall	mAP@0.5	F1	AUC
原始資料集	0.764	0.86	0.857	0.809	0.854
傳統資料擴增	0.962	0.993	0.992	0.977	0.996

傳統資料擴增與原始資料集之結果長條圖

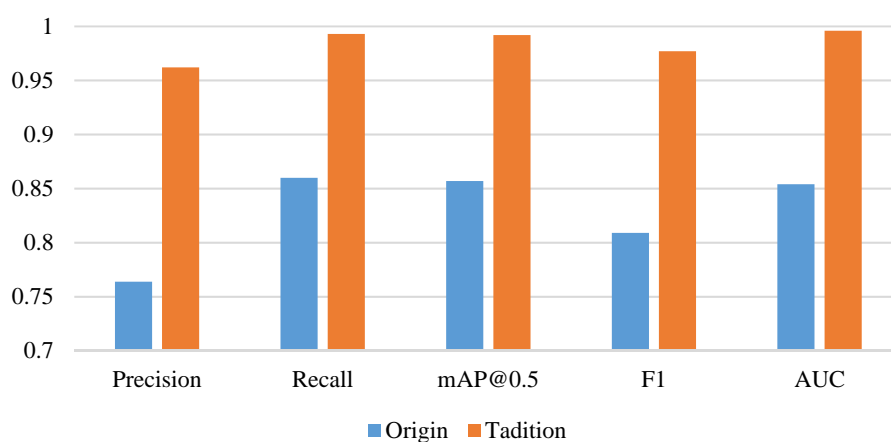


圖 6.17 傳統資料擴增與原始資料集之結果長條圖

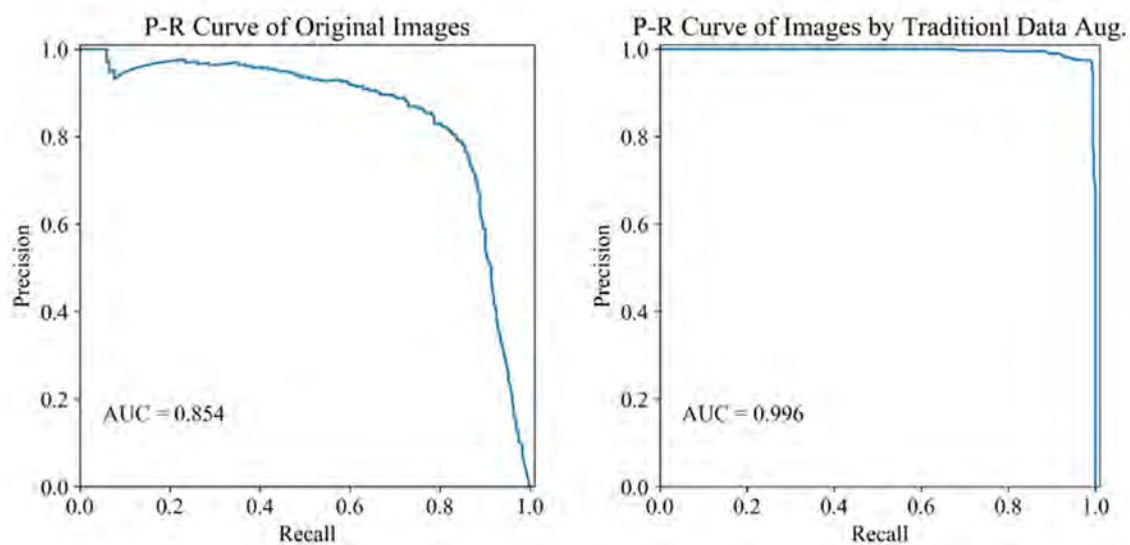


圖 6.18 原始資料集與傳統資料擴增的 PR-Curve 之比較



6.5.2 Cycle-GAN

這部分進行 Cycle-GAN 的訓練，以獲得 KVASIR 圖像之分佈，並將模擬腸道轉換成真實腸道。我們有自製之模擬腸道資料集，其資料處理方式與 KVASIR 資料集相同，共 8935 張圖像。在 Cycle-GAN 訓練所使用模擬腸道之圖像有 1015 張，並設為 X 資料集，真實腸道為 KVASIR 資料集之圖像共 1200 張圖像，設為 Y 資料集；驗證之 X 資料集為 185 張模擬腸道圖像，Y 資料集為 200 張真實腸道圖像。模型的參數由 Adam 優化器進行優化損失，並訓練 200 Epoches、batch-size 為 1，學習率為 0.0002，訓練過程之損失如圖 6.19 所示。

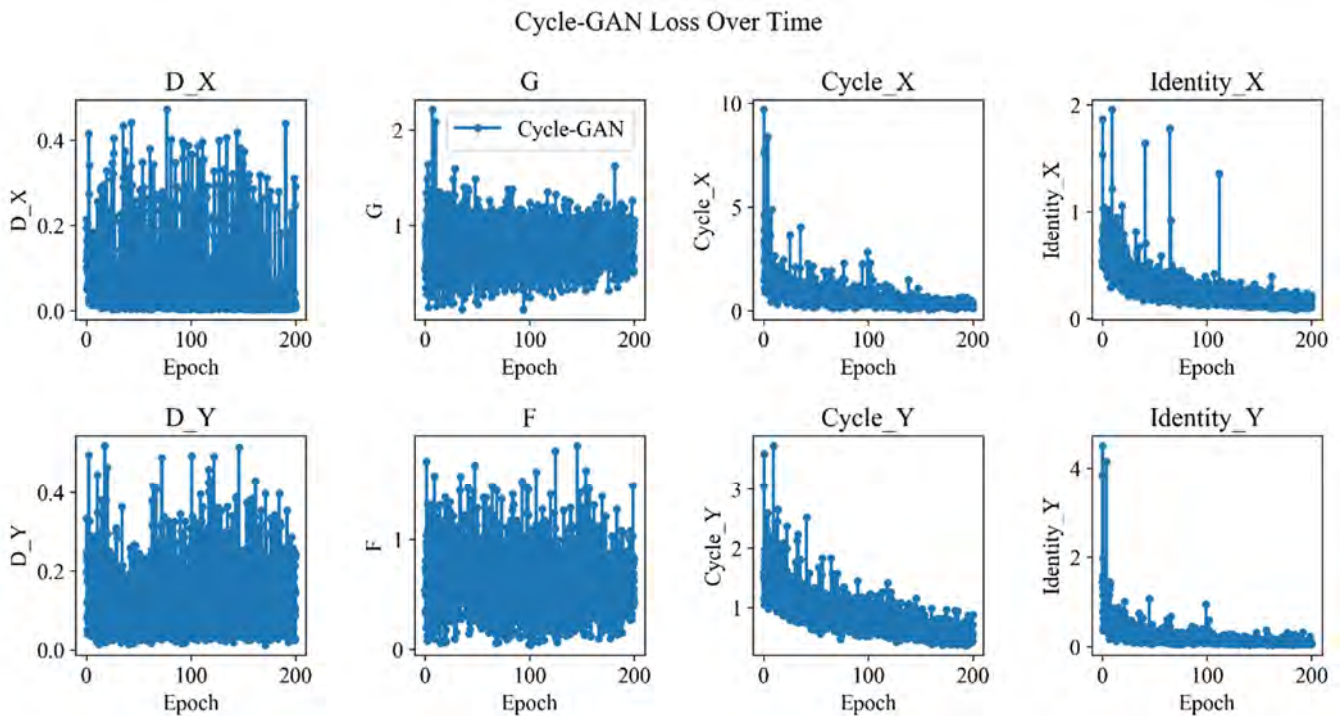


圖 6.19 Cycle-GAN 訓練過程之損失

在 Epoch 1 訓練所產生之圖像，如圖 6.20 所示，圖(a)為 X 資料集之原圖像 x 。圖(a)經過生成網路 G 後轉換成圖(b)，此為 $G(x)$ 。將圖(b)經過生成網路 F 後轉換成圖(c)，此為 $F(G(x))$ 。從圖(a) → 圖(b) → 圖(c) 形成循環。圖(d)為 X 資料集之圖像 x 經過生成網路 F 後生成，使得生成網路 F 在生成過後能與原圖之顏色成分保留。另一方面，圖(e)為 Y 資料集之原圖像 y 。圖(e)經過生成網路 F 後轉換成

圖(f)，此為 $F(y)$ 。將圖(f) 經過生成網路 G 後轉換成圖(g)，此為 $G(F(y))$ 。從圖(e) → 圖(f) → 圖(g) 形成循環。圖(h)為 Y 資料集之圖像 y 經過生成網路 G 後生成。

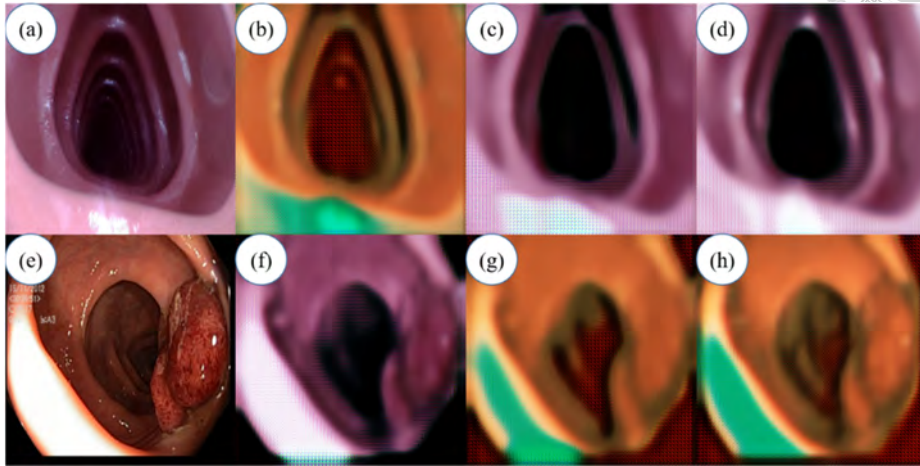


圖 6.20 在 Epoch 1 生成之圖像

在訓練過程進行到 Epoch 50，如圖 6.21 所示，生成網路 G 從圖(a) 轉換成圖(b)與生成網路 F 從圖(e) 轉換成圖(f) 尚未轉換較為失真。在循環回來之後，圖(c) 與圖(a) 比較尚有模糊之處理，但到了後續的 Epoch 100、Epoch 147、Epoch 199 (如圖 6.22、圖 6.23、圖 6.24 所示)，可以看出圖(a) 到圖(b) 資料集 X 轉換到資料集 Y 轉換得非常成功，故利用此結果將模擬腸道 8935 張圖像轉換成 KVASIR 資料集之風格。

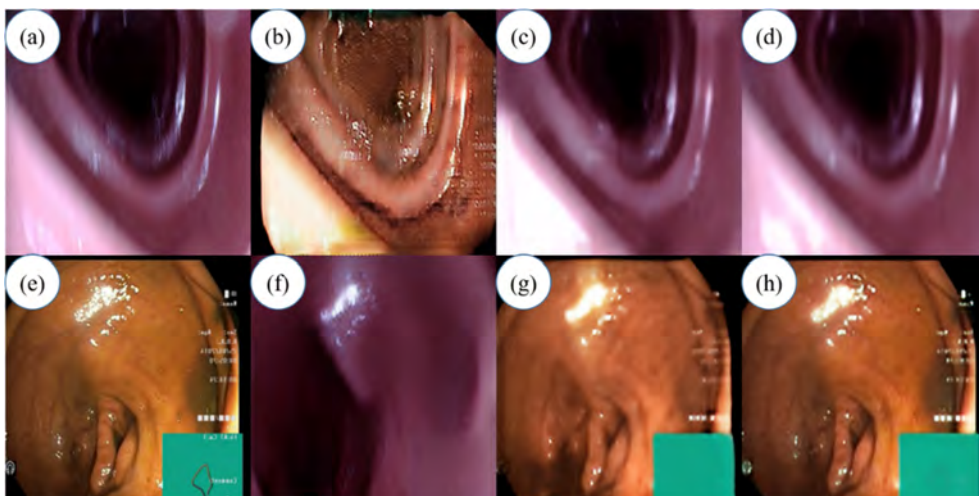


圖 6.21 在 Epoch 50 生成之圖像

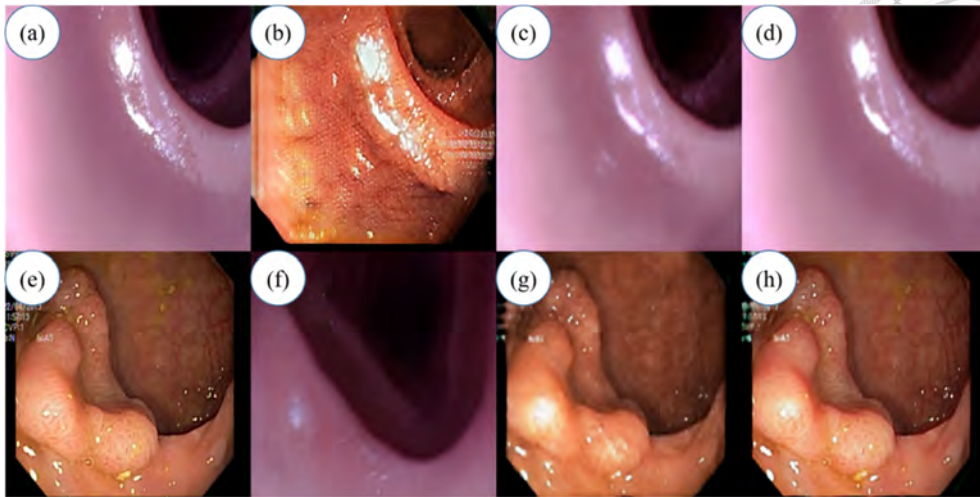
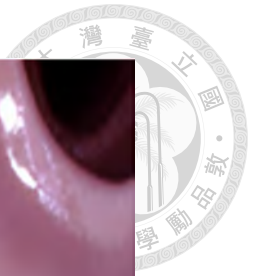


圖 6.22 在 Epoch 100 生成之圖像

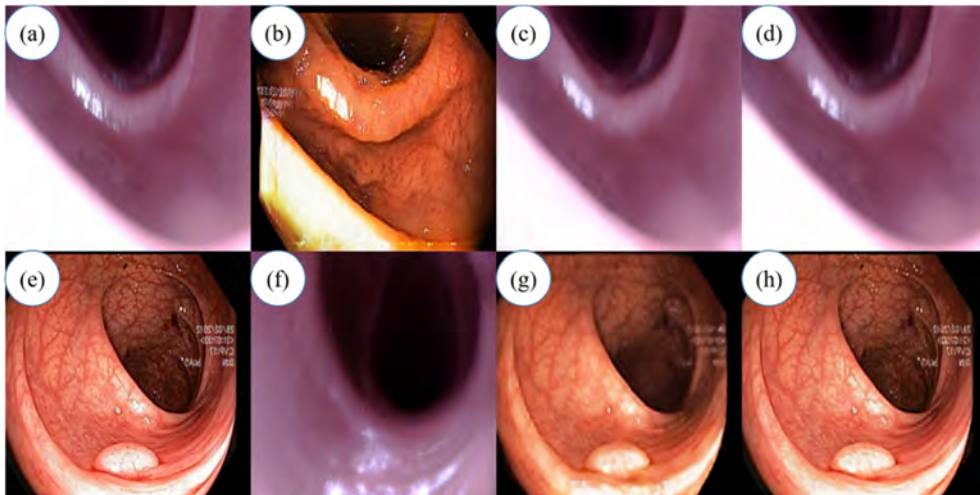


圖 6.23 在 Epoch 147 生成之圖像

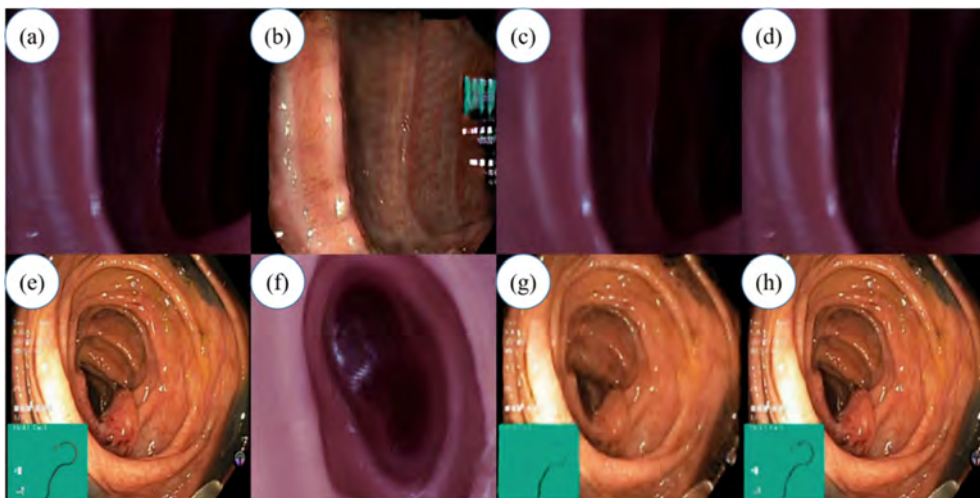


圖 6.24 在 Epoch 199 生成之圖像

將 8935 張新生成之圖像與原始資料集進行整理，如表 6.7 所示，加入訓練集之中。訓練過程之訓練參數如同章節 6.5.1 之設置方式，並進行 YOLOv3 訓練，如圖 6.25 所示。測試結果與原始資料集互相比較，如表 6.8 與圖 6.26 所示，Cycle-GAN 之 PR-Curve 如圖 6.27 右圖所示。Cycle-GAN 比原始資料集之驗證些微良好，但仍低於傳統資料擴增驗證之性能指標。

表 6.7 Cycle-GAN 擴增之數量

擴增新圖片之數量	
資料擴增生成之訓練集	11079
Cycle-GAN 生成之圖像	8935
原始訓練集	2144
原始驗證集	536
總計	11615

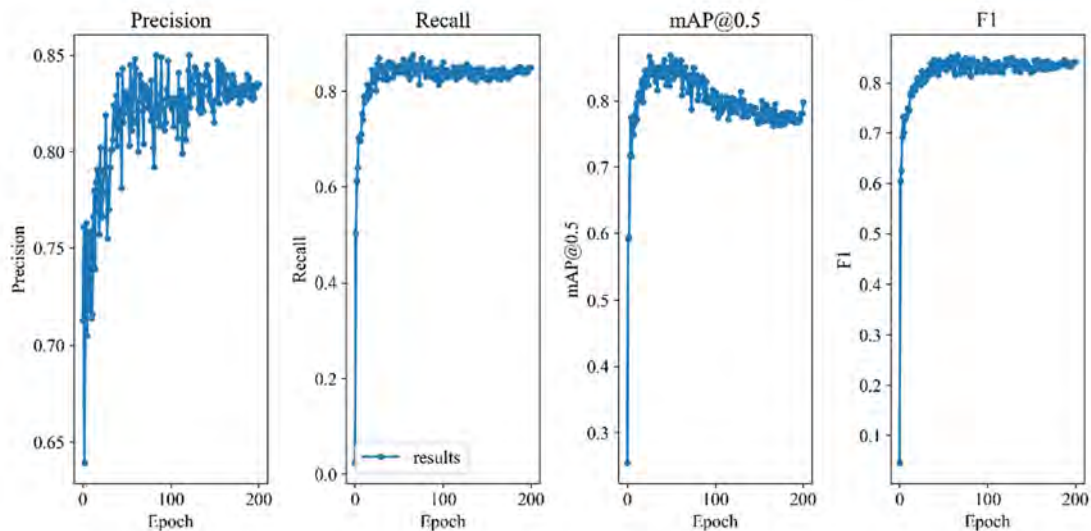


圖 6.25 Cycle-GAN 擴增訓練過程之性能

表 6.8 Cycle-GAN 與原始資料集之結果比較

	Precision	Recall	mAP@0.5	F1	AUC
原始資料集	0.764	0.86	0.857	0.809	0.854
Cycle-GAN 擴增	0.79	0.856	0.877	0.822	0.874
傳統資料擴增	0.957	0.991	0.99	0.974	0.996

Cycle-GAN與原始資料集之結果長條圖

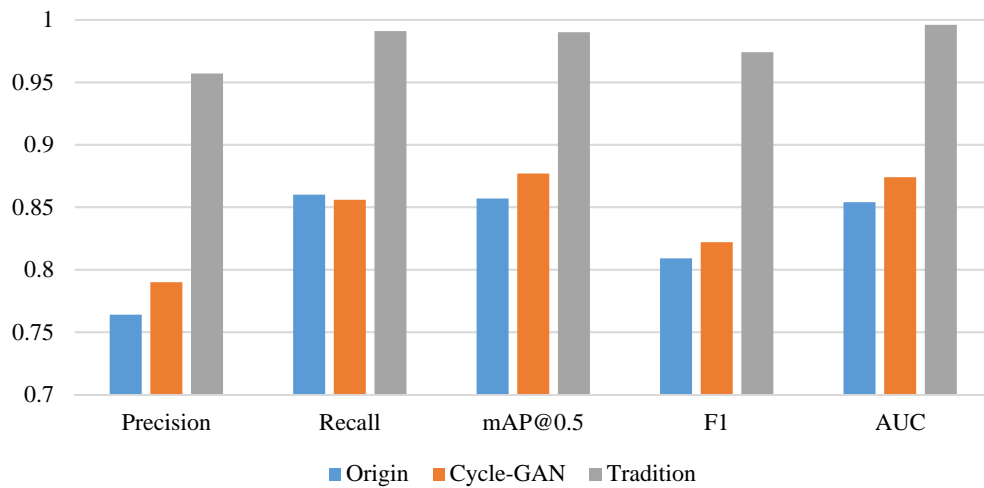


圖 6.26 Cycle-GAN 與原始資料集之結果長條圖

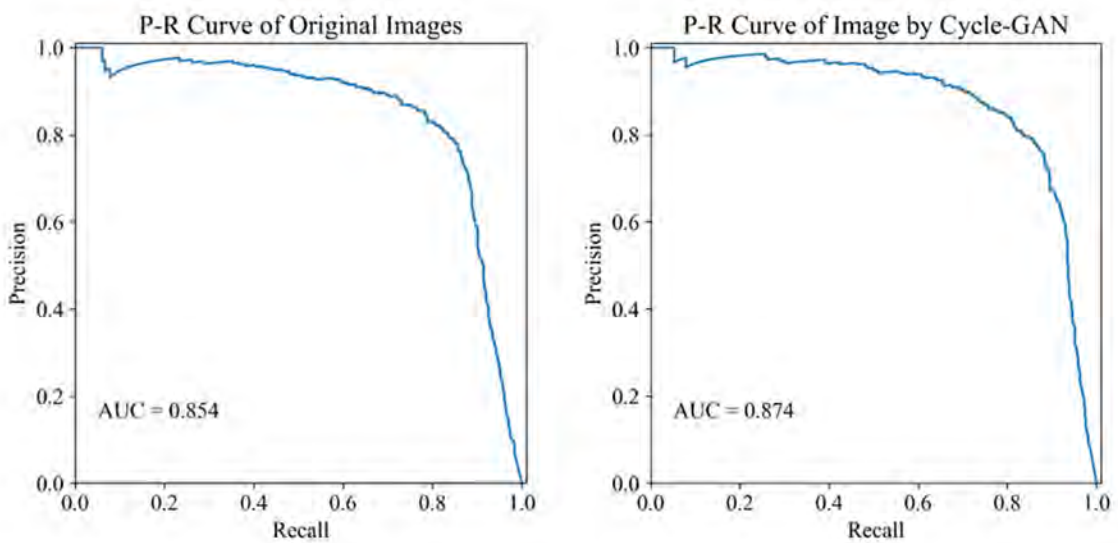


圖 6.27 原始資料集與 Cycle-GAN 的 PR-Curve 之比較

Cycle-GAN 成功將模擬腸道圖像進行轉換。然而，Cycle-GAN 擴增之圖像轉換有些許問題，如圖 6.28 所示。從圖 6.28 可知，右圖所生成之真實腸道圖像皆有綠色圖案，其原因為原始資料集有部分圖像左下角或右下角部分有綠色框，因此 Cycle-GAN 所學習之分佈會產生綠色圖案，導致圖像之腸腔部分皆被擋住。亦有些部分之腸壁表現不完善，因模擬腸道之圖像較昏暗，也使得其分佈無法跟真實腸道之分佈比較，所獲得之圖像才會導致失真。

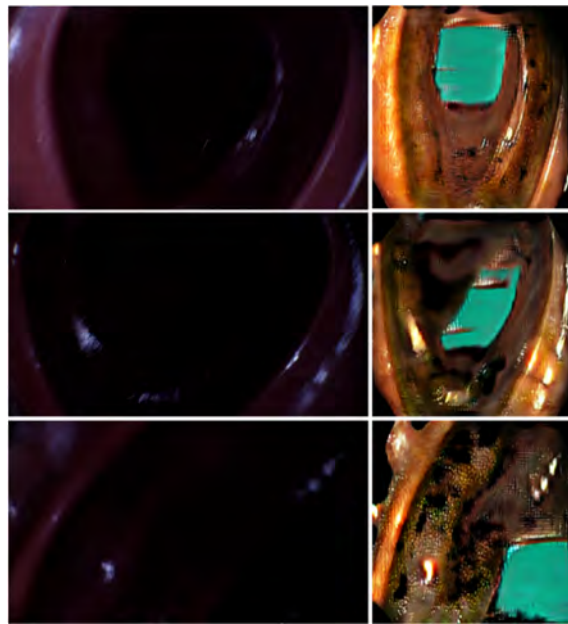


圖 6.28 Cycle-GAN 生成失敗之案例

6.5.3 YOLOv3 腸腔辨識

最後，透過傳統資料擴增和 Cycle-GAN 擴增的資料，並與原始資料集之性能互相比較。原始資料集與所生成之圖像數量共計 25015 張，如表 6.9 所示，最終擴增方式之數量增加為原本的九倍。訓練過程之訓練參數如同章節 6.5.1 之設置方式，並進行 YOLOv3 訓練，如圖 6.29 所示。可以從圖 6.29 知道，每個性能指標在訓練過程中都達到 0.95 以上。最後測試結果與原始資料集進行比較，如表 6.10 與圖 6.30 所示，其性能指標有顯著增加，且全部資料擴增之 PR-curve 曲線越接近右上角，數值高達 0.995，如圖 6.31 所示。

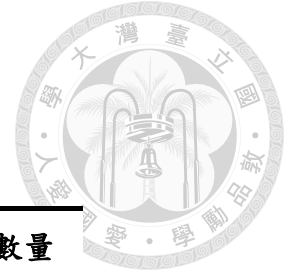


表 6.9 全部擴增方式擴增之數量

擴增新圖片之數量	
資料擴增生成之訓練集	24479
原始訓練集	2144
圖像平移	2680
圖像旋轉	2680
圖像縮放	2680
高斯雜訊	2680
動態模糊	2680
Cycle-GAN	8935
原始驗證集	536
總計	25015

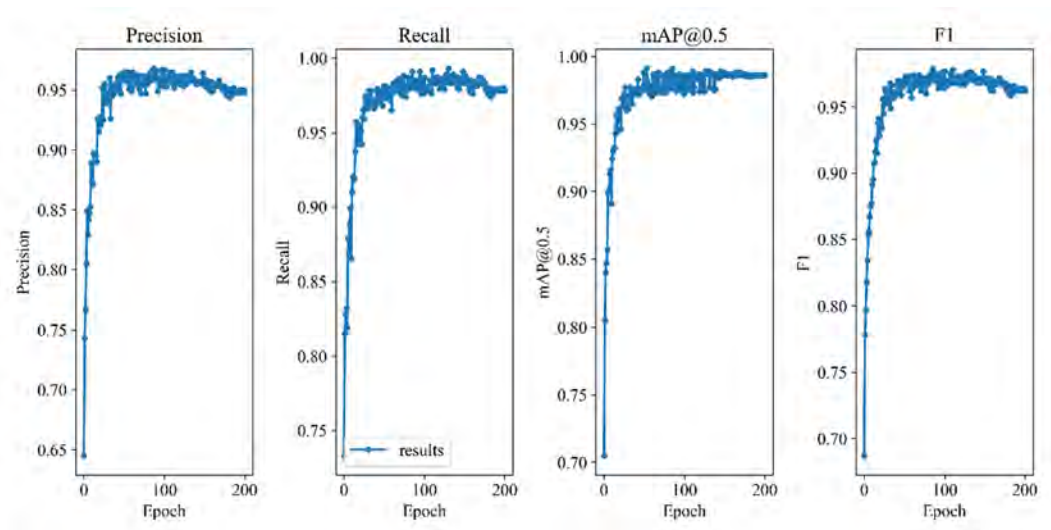


圖 6.29 全部擴增訓練過程之性能

表 6.10 全部擴增與原始資料集的結果之比較

	Precision	Recall	mAP@0.5	F1	AUC
原始資料集	0.764	0.86	0.857	0.809	0.854
全部擴增	0.963	0.993	0.991	0.978	0.995

全部擴增與原始資料集的結果長條圖

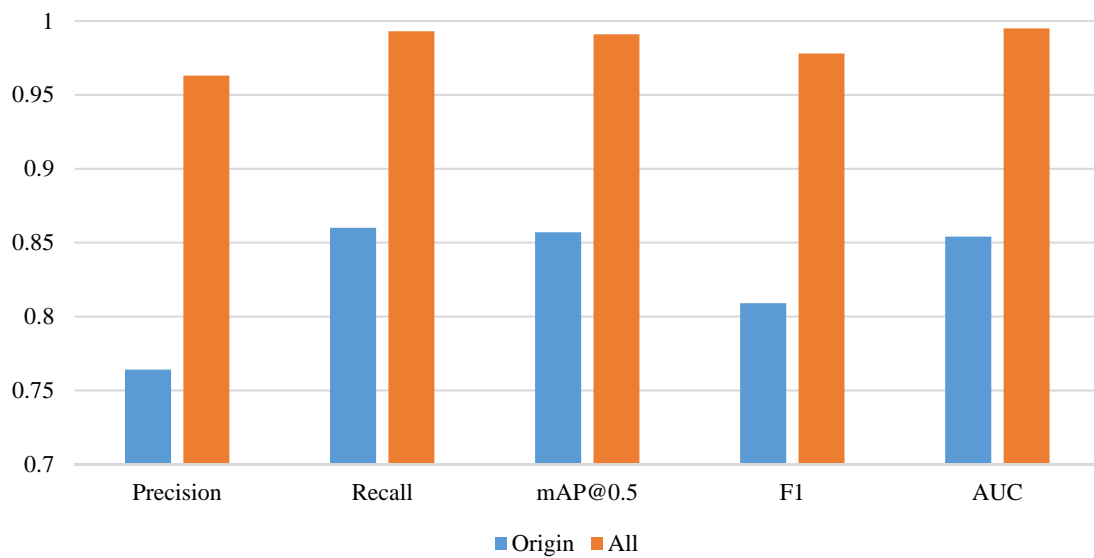


圖 6.30 全部擴增與原始資料集的結果長條圖

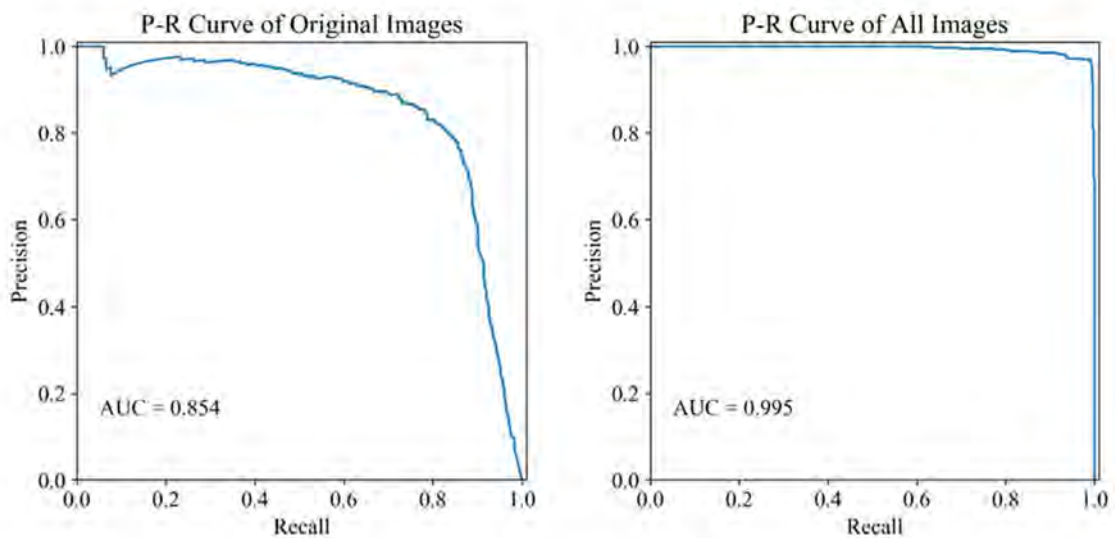


圖 6.31 原始資料集與全部擴增的 PR-Curve 之比較

第七章 結論與未來工作



7.1 結論

首先，在這研究中提出改善腸腔辨識以利磁控膠囊內視鏡進行導航之方法。這個方法是藉由資料擴增的方式進行，利用兩種擴增方式：傳統資料擴增與 Cycle-GAN，將少量卻精確的資料集 2680 張圖像擴增至其 9 倍數量之資料集 25015 張圖像，以增強其辨識能力。在這研究中先訓練原始資料集，其測試結果 Precision 為 0.764，Recall 為 0.86，mAP 為 0.857、以及 F1-score 為 0.809，AUC of PR-curve 為 0.854。

傳統資料擴增利用現有的圖像進行平移、旋轉、縮放、加入高斯雜訊與加入動態模糊。每一種傳統之擴增方式增加的數量為原始資料集之總數量，因此擴增了 13400 張圖像，並與原始資料集進行訓練。每一種擴增方式訓練過後與原始資料集相比都有著顯著提升。之後將傳統資料擴增方式之資料集進行訓練，測試結果 Precision 為 0.962，Recall 為 0.993，mAP 為 0.992、以及 F1-score 為 0.977，AUC of PR-curve 為 0.996。

另一種擴增方式是利用 Cycle-GAN 進行擴增，將 8935 張模擬腸道之圖像轉換成真實腸道之圖像。Cycle-GAN 擴增之圖像由於訓練資料之真實圖像有綠色框，導致其生成之圖像有些微失真。因此與原始資料集互相比較，Cycle-GAN 擴增對於 YOLOv3 只有些微的改善。Recall 部分原始資料集為 0.86，而 Cycle-GAN 只有 0.856，兩者數值仍較為相近。雖然 Recall 變低，但 Precision、mAP 及 F1-score 都有上升。

最後將所有擴增過後之圖像與原始資料集進行結合，利用 YOLOv3 訓練模型，所有性能指標與原始資料集比較皆增長 1.1 ~ 1.2 倍，Precision 為 0.963，Recall 為 0.993，mAP 為 0.991、以及 F1-score 為 0.978，AUC of PR-curve 為 0.995。



因此利用兩種資料擴增方式擴增，可達到提升腸腔辨識率之效果。

7.2 未來工作

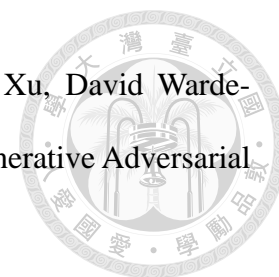
本論文採用傳統資料擴增方式與 Cycle-GAN 進行資料擴增來實現。然而，未來在 MFN Platform 操作磁控膠囊內視鏡上進行真實腸道之導航有很多不確定之因素，導致腸腔無法被辨識，或是無法自動導航。為了使 MFN Platform 與磁控膠囊內視鏡在導航上更加良好，以下是未來可以著手得部分：

- (1) 增加更多的腸腔圖像至訓練之資料集，如圖像上有糞便、腸液等。面對更多的不確定之因素，YOLOv3 亦能夠辨識出腸腔位置。
- (2) 以往皆無採取影像來輔助自動導航進行，都是由 MFN Platform 之機械手臂上的感測器進行判斷方向與行走。因此希望在自動導航之演算法中能夠加入 YOLOv3 之影像，使其更加清楚表現出腸腔之位置，以減少自動導航所消耗之時間。
- (3) 增加且篩選 Cycle-GAN 之訓練資料與測試資料，使得 Cycle-GAN 轉化之風格無綠色框顯現。可利用傳統資料擴增之方式將真實腸道之顏色轉換成昏暗形式，使得模擬腸道轉換不會失真。
- (4) 將 MFN Platform 之系統結合 Cycle-GAN 形成即時轉換影片，使得操作者所控制之畫面轉換成真實腸道，來熟悉真實腸道之操作，以利後續人體試驗進行。

參考文獻



- [1] 衛生福利部. "108 年死因記者會簡報." [Online]. Available: <https://dep.mohw.gov.tw/DOS/cp-4927-54467-113.html>
- [2] 以色列 GivenImaging 官方網站. [Online]. Available: <http://www.givenimaging.com>
- [3] Yu-Dong Zhang, Zhengchao Dong, Xianqing Chen, Wenjuan Jia, Sidan Du, Khan Muhammad, Shui-Hua Wang. "Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation." *Multimed Tools Appl* 78, 3613–3632 (2019)
- [4] Agnieszka Mikołajczyk and Michał Grochowski. "Data augmentation for improving deep learning in image classification problem." *2018 International Interdisciplinary PhD Workshop (IIPHDW)*, Swinoujście, 2018, pp. 117-122, doi: 10.1109/IIPHDW.2018.8388338. (2018)
- [5] Jelmer Wolterink, Anna Dinkla, Mark Savenije, Peter Seevinck, Cornelis A.T. van den Berg, Ivana Išgum. "Deep MR to CT synthesis using unpaired data." In: SASHIMI. pp. 14–23 (2017)
- [6] Gi-Shih Lien, Chih-Wen Liu, Joe-Air Jiang, Cheng-Long Chuang and Ming-Tsung Teng. "Magnetic Control System Targeted for Capsule Endoscopic Operations in the Stomach—Design, Fabrication, and in vitro and ex vivo Evaluations." in *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 7, pp. 2068-2079, July (2012)
- [7] 高振翔. "磁控膠囊內視鏡在腸道內移動之研究." 臺灣大學碩士論文. (2017)
- [8] 京都科學官方網站. [Online]. Available: <https://www.kyotokagaku.com/products/detail01/m40.html>



- [9] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. “Generative Adversarial Nets.” In NIPS’2014. (2014)
- [10] Jemma. “GAN 入門理解及公式推導.” [Online]. Available: <https://zhuanlan.zhihu.com/p/28853704>
- [11] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. “Unpaired image-to-image translation using cycle-consistent adversarial networks.” In: CoRR, abs/1703.10593 (2017)
- [12] Cycle-GAN 2017. Cycle-GAN: Open Source GAN. [Online]. Available: <http://junyanz.github.io/CycleGAN/>
- [13] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. “Rich feature hierarchies for accurate object detection and semantic segmentation.” In: CoRR abs/1311.2524 (2013)
- [14] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. “You Only Look Once: Unified, Real-Time Object Detection.” In: arXiv: 1506.02640 (2015)
- [15] Joseph Redmon. Darknet: Open Source Neural Network. [Online]. Available: <https://pjreddie.com/darknet/yolo/>
- [16] Joseph Redmon, Ali Farhadi. “YOLOv3: An Incremental Improvement.” In: arXiv: 1804.02767 (2018)
- [17] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, Andrew Zisserman. “The PASCAL Visual Object Classes (VOC) Challenge.” *Int J Comput Vis* 88, 303–338 (2010).