南華大學科技學院永續綠色科技碩士學位學程

碩士論文

Master Program of Green Technology for Sustainability
College of Science and Technology
Nanhua University

Master Thesis

基於 U-Net 神經網路與熱像儀實現人流計數

Crowd Counting via Thermal Imaging and U-Net

張功爾

Kong-Erh Chang

指導教授：黃冠雄 博士

Advisor: Guan-Shyong Hwang, Ph.D.

中華民國 112 年 1 月

January 2023

南 華 大 學

永續綠色科技碩士學位學程

碩 士 學 位 論 文

基於 U-Net 神經網路與熱像儀實現人流計數

Crowd Counting Via Thermal Imaging and U-Net

研究生：張功爾

經考試合格特此證明

口試委員：賴信吉

茉文昇

黃冠雄

指導教授：黃冠雄

系主任(所長)：科技學院永續綠色科技碩士學位學程主任洪耀明(甲)

口試日期：中華民國 111 年 7 月 1 日

致謝

首先我要感謝我的指導教授－黃冠雄博士提供我具國際級水準的研究環境與能量，使我能在南華大學時期，能潛心研究與學習新知，更引領我挑戰未知的領域－深度學習，在教授的帶領下完成了實驗設計與 AI 的研究基礎，透過與老師的討論使學生的研究更加完整，同時也提拔學生往更高的學術殿堂發展，實屬感激。感謝共同指導教授賴信志老師在課外的幫助，給予我在研究以外的發展，十分感謝。感謝兩位指導老師與莊文河、賴信志百忙之中參與學生的碩士學位口試，在口試中給予寶貴的意見，使學生的論文更加完備。感謝楊恩麒的幫助，使我的作品集能更加豐富。感謝以上前輩、教授的幫助，使學生獲得滿滿的研究能量，更促使我往畢業的道路前進，萬分感謝。

碩士班修業期間，感謝兩位好戰友筵龍、璨爰，能夠互相討論彼此的研究上的問題並解決疑慮，讓研究視野更加開闊，除了學業外，也一同娛樂出遊。感謝恩麒、璨爰常常一起吃飯，讓我們在嘉義時期的生活更加豐富。

最後我要感謝我的父母親與弟弟們的支持，使我在求學的道路上能好好完成，讓我在就學期間資源不虞匱乏，不管是物質上或精神上的幫助與付出，也時常提醒我在研究忙碌時不忘要好好放鬆，保持生活與研究間的平衡過著快樂的心情去研究。感謝我的太太－凱東，陪伴我在南華學的期間，和我一起奮鬥努力，從研究聊到創業、旅遊，讓我在研究之外也能很好玩的事情能夠參與，也都順利完成了學業。
謹以此本文成果，獻給過去曾幫助我、關心我的人們，未來的我也會好好的努力。

張功爾 謹誌於

南華大學 永續綠色科技碩士學位學程

摘要

自動人群行為分析是智能交通系統的一項重要任務，可以為不同的道路參與者實現有效的流量控制和動態路線規劃。人群計數是自動人群行為分析的關鍵之一。近年來，使用深度卷積神經網絡 (CNN) 進行人群計數取得了令人鼓舞的進展。研究人員在變體 CNN 架構的設計上投入了大量精力，其中大部分都是基於預訓練的 VGG16 模型。由於表達能力不足，VGG16 的骨幹網絡後面通常是另一個笨重的網絡，專門為良好的計數性能而設計。儘管 VGG 模型在圖像分類任務中已經優於 Inception 模型，但現有的使用 U-Net 構建的人群計數網絡仍然只有少量具有 U-Net 模塊基本類型的層。為了填補這一空白，在本文中，我們首先在常用人群計數數據集上對基線 U-Net 模型進行了基準測試，並取得了與大多數現有人群計數模型相當或更好的驚人性能。隨後，我們通過提出以 U-Net 為骨幹的分割引導注意網絡和用於人群計數的新課程損失，進一步推動人群計數的極限。

關鍵詞：人群計數、課程式學習損失函數、分割與注意力神經網路框架

ABSTRACT

Automated crowd behavior is an important task of intelligent traffic systems, which can implement efficient flow control and dynamic route planning for different road participants. Crowd counting is one of the keys to automatic crowd behavior. In recent years, there has been encouraging progress in crowd counting using deep convolutional neural networks (CNNs). Researchers have devoted a lot of effort to designing variant CNN architectures, most of which are based on pre-trained VGG16 models. Due to lack of presentation, the backbone network of the VGG16 is usually behind another heavy network, designed specifically for good counting performance. Although the VGG model is already superior to the Inception model for image categorization tasks, traditional crowd counting networks built using U-Net modules still have only a small number of layers with a basic type of U-Net module. To fill this gap, we first tested the baseline U-Net model on a common population count dataset and achieved remarkable performance comparable to or better than most existing population count models. Subsequently, we further push the limits of crowd counting by proposing U-Net-based segmentation to guide attention to networks and new lesson losses for crowd counting.

Keywords: Crowd Counting, Curriculum loss function, Segmentation guided attention networks, U-Net

目錄

圖次

表次

第1章　緒論

1.1 研究背景與動機

在公共監控與智能交通系統中[1-5]，自動人群計數重要性在研究界中受到越來越多的關注。人群密集程度會對公共交通產生重大的影響，因此在智慧城市與智能交通系統中(如圖1)能夠從公共監控中實時獲取人群相關資料，並動態調整規劃以實現更有效率的交通指示，就顯得相當重要。人群和車輛計數可以在統一的計數框架中訂製，該框架旨在估計靜態圖片或影片人群數量，並已在許多實際場景中得到應用。例如，已經有一些專注在自動計數不同的對象，包括細胞[6]、車輛[7, 8]、樹葉[9, 10]和人[4]。在這樣的條件限制下，如何利用熱像儀提高人流計數的精準度，且通用各式不同的場域將會是一大挑戰。此外，智慧城市與智能交通系統的商業活動盛行且經濟利潤高，若加入注重隱私的熱像儀的技術則會產生外溢效果，應用於智慧城市最基礎的人群計數中，提供另一種兼顧隱私權的解決方案。

圖 1，智慧城市的廣泛，深度盤點中的資訊應用系統，顯得相當重要 (資料來源：逢甲大學資訊系統研究中心)

1.2 研究目的

早些年，人群計數是由邊緣偵測[11-13]或計數回歸[14, 15]來實現。過去隱含一種問題，邊緣偵測是假設完整的人形在圖片內，因此提取邊緣後若被判斷為人形就能容易的實現計數，然而，這樣的假設並不總是成立，尤其當人群非常密集時，就顯得難以執行。為了能夠解決假設完整的人形才能夠計數，本篇提出使用先使用深度學習邊緣特徵提取的方式，將各式各樣的人形先完整框出來，同時滿足人群非常密集的時候，人形很容易失真的問題。接下來計數的方式就比較單純一些，計數回歸目標在學習回歸模型(如：支持向量機[15]或神經網路[14])，將人工合成的圖片特徵直接映射到圖片中的人數，其缺點為缺乏可靠性與可解釋性。近幾年，自從密度圖的概念在[16]中首次被提出之後，多項人群計數研究皆會基於密度估計為神經網路提供可解釋的量測方式。使用深度卷積神經網路[17]來估計密度圖以及大規模數據集的可用性[18, 19]，進一步提高真實場景的準確性。最近在人群計數方面的研究一直專注於深度神經網路(如：CNN[18, 20]和注意力機制[21, 22])的架構設計，以實現精確的密度圖估計。這些設計的動機通常是改進尺度的變化以增強人群圖片的泛化能力，其中，VGG16、VGG19、ResNet101 和 Inception 已在[23]、[24]、[25]、[26] 中用作人群計數的骨幹網絡。實際舉例：假設智慧城市一秒鐘產生大量的圖片，是否準確的預測，是否能夠及時偵測，受其影響的個案中工程需求是多變的，因此，在本文中，我們嘗試研究 U-Net 模型對人群計數的有效性，提出一種解決方案以熱像儀做人流計數並含有動態之效果，進而發展出智慧城市人流計數演算法。

1.3 研究範圍

本研究主要探討範圍僅限單張人群計數、不同的神經網路精確度的差異、不同的公開資料集的消融實驗、注意力神經網路的有效性，以及領域的適應性。本研究所使用的神經網路為 U-Net 自動編碼器架構，結合課程式學習(Curriculum Learning)，分割與注意力神經網路，因此混合不同有效的方式為本論文的亮點，工程案件中的多變性與智慧城市的連結性，不在此次研究範圍內。

1.4 研究方法與步驟

首先進行本研究動機、目的以及範圍之說明。第二章，透過相關文獻探討方式，針對神經網路的基礎知識開始著手，包括卷積神經網路、FCN 模型、U-Net 模型、FCN 和 UNet 的差別，總共四小結加以討論。第三章，本研究方法以深度學習式演算法並配合本研究設立之情境進行修改，應用於熱像儀人流計數，並提出模型架構流程圖，再來展示本研究所使用之 ShanghaiTech 數據公開資料集，導入參數並實際執行演算法之步驟，以此來驗證模型與演算法之可行性與結果。第四章，依據本研究結果提出結論以及未來方向之建議。

第2章　文獻探討

2.1 人工神經網路簡介

人工神經網路(Artificial Neuron Network)，簡稱 ANN，受到生物學習系統的啟發，由類似大腦內處理複雜資訊的神經元所建構成的一種運算模型，如表 1 所示。人類腦中含有大小約 $10^{11} - 10^{12}$ 個神經元所構成的密集相連網路，每個相連的神經元，平均連接 $10^4 - 10^5$ 個神經元，而 ANN 捕捉這種分散性高度平行運算的特性，在電腦視覺、聲音辨識、機器人控制的決策等領域中取得成功。

表 1，人腦與電腦在處理資訊上的差異

人腦(生物神經網路)	電腦(人工神經網路)
非同步工作模式	同步工作模式
計算速度緩慢($\cong 10^{-3}$sec)	計算速度快($<10^{-9}$sec)
採分散式處理資訊，神經元不可靠且隨時可能死亡	處理資訊時每一個步驟需按照順序，否則程式無法順利運行
隨著時間推移，大腦會改變連結性，以應付新資訊的要求	電子元件間的連結性不會改變，除非更換零件
具有複雜的拓樸(Topologies)結構	通常採用樹(Tree)結構
科學家仍在探究大腦實際學習方式	使用梯度下降法(Gradient descent)學習

最初人工神經元的模型 (The McCulloch-Pitts model of neuron) 是 Warren McCulloch 和 Walter Pitts 於 1943 年提出，也被稱作線性閥值閘[19]。這個神經元是由 N 個輸入I_i和 1 個輸出 Y 所組成，負責將這 N 個輸入分成兩組不同的類別，這樣的函數可以被表示以下數學式:

$$Sum = \sum_{i=1}^{n} I_i w_i \tag{2.1}$$

$$Y = f(Sum) \tag{2.2}$$

神經網路從其他節點或外部資源接收輸入並計算輸出，每一項輸入 I_i 都有對應的權重(w_i)，根據重要性來分配大小，將函數 Y 定義如公式 2.2 所示，單一神經元[21]，如圖 2 所示：

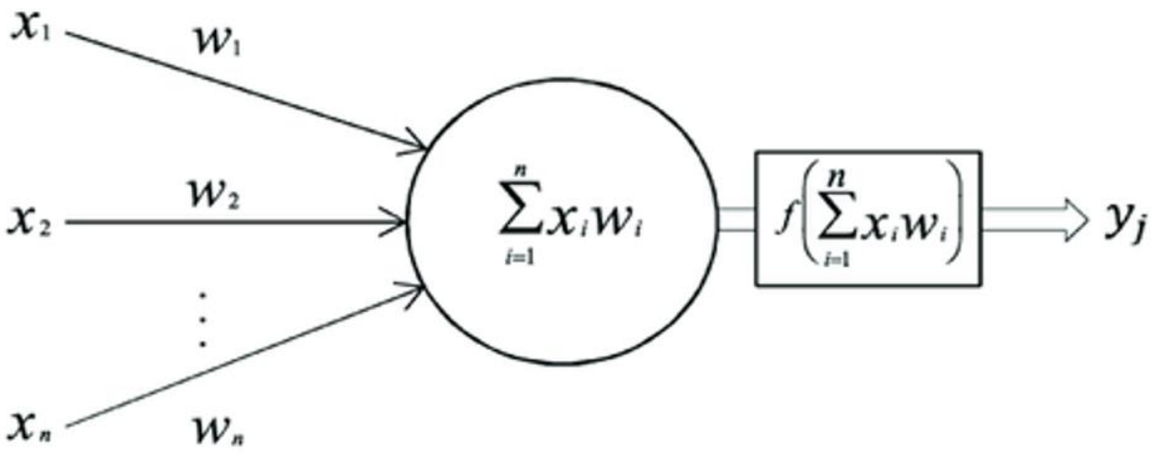


圖 2，單一神經元[21]

圖 2 的神經網路有 x_1、x_2 兩個輸入和相應的 w_1、w_2 的權重;此外還有一項 b 為 bias， 其主要功能為提供每個神經元一項可訓練的常數值，允許移動線性函數中的線來幫助數據預 測的擬合;函數 f 為激勵函數(Activation function)，目的是使輸出非線性化，因為在現實世界的問題絕大部分都是非線性的，關於激勵函數會在下面更詳盡解釋。在 ANN 中，神經元的幾何排列至關重要，所有神經網路皆具備輸入層、輸出層，另外多出的則是隱藏層(Hidden Layer)，可以增加系統運算處理能力，同時也使訓練過程更加複雜，對使用者們來說就如同黑盒子(Black Box)。前饋神經網路 (Feedforward Neural Network) 是第一種也是最簡單的 ANN，如圖 3 所示，資訊由輸入節點-隱藏節點-輸出節點單向移動， 不像循環神經網路(Recurrent Neural Network) 一樣有循環。

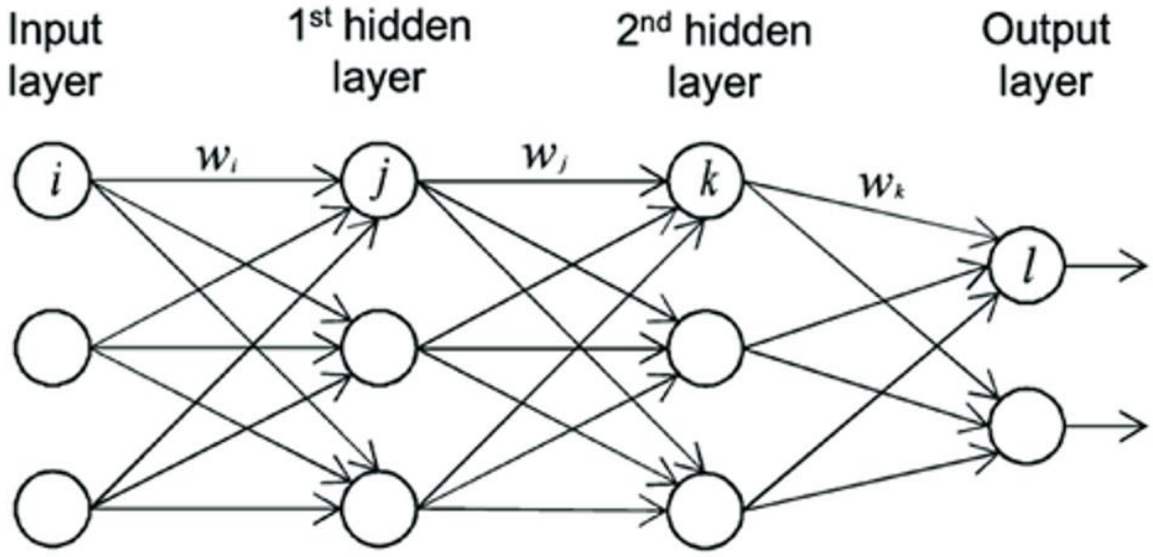


圖 3，前饋神經網路範例[21]

激勵函數(Activation Function)與損失函數(Loss Function) 在建立神經網路的過程中，必須要抉擇在隱藏層和輸出層間使用哪種激勵函數(Activation Function)，透過計算權重和添加偏差(bias)與否來決定是否應該激發該神經元，目的是為了將 使輸出非線性化。在實作中，會遇到以下幾種激勵函數:

Sigmoid $$\sigma(x) = \frac{1}{1+e^{-x}} \quad (2.3)$$

Tanh $$\tanh(x) = 2\sigma(2x) - 1 \quad (2.4)$$

ReLU $$f(x) = max(0, x) \quad (2.5)$$

Softmax $$\sigma(x)_j = \frac{e^{x_j}}{\sum_{k=1}^{K} e^{x_k}} \; for\; j = 1, \dots, K \quad (2.6)$$

損失函數(Loss Function)是一種評估算法，用以衡量根據資料集所搭建的模型，如果預測效果不佳，那損失函數將會是一個很高的數值，當我們不斷調整參數改進神經網路模型時， 損失函數會給予我們一些回饋。實作上，François Chollet 出版的書籍[22]也在 Tensorflow Keras 框架下，提供如何根據不同問題的類型去選擇激勵函數和損失函數(Loss Function)的參考表格，當然也可以根據資料集的特性去多嘗試，請見表 2:

表 2，不同問題類型相應之激勵與損失函數表

Problem type	Last-layer activation function	Loss function
Binary classification	Sigmoid	Binary_crossentropy
Multiclassand singled-label classification	Softmax	Catrgorical_crossentropy
Multiclassand singled-label classification	Sigmoid	Binary_crossentropy
Regression to arbitrary values	None	MSE
Regression to values between 0 and 1	Sigmoid	MSE or Binary_crossentropy

在計算機科學中，我們利用「矩陣」來模擬建造神經網路的過程，為了簡單說明神經網路，利用 Python 搭建兩層的網路來解決線性問題，見表 3 範例。訓練過程包含以下步驟：

- 前向傳播(Forward Propagation)：將輸入與權重相乘(此處以亂數作為權重)，使$Y = W_i X_i = W_1 X_1 + W_2 X_2 + W_3 X_3 Y$，將這結果送入可將數值歸一化在範圍 0-1 的 Sigmoid 函數，計算出神經元的輸出。
- 反向傳播(Back Propagation)：計算實際輸出和預期輸出間的誤差，利用誤差傳播回神經網路來計算梯度並調整權重。

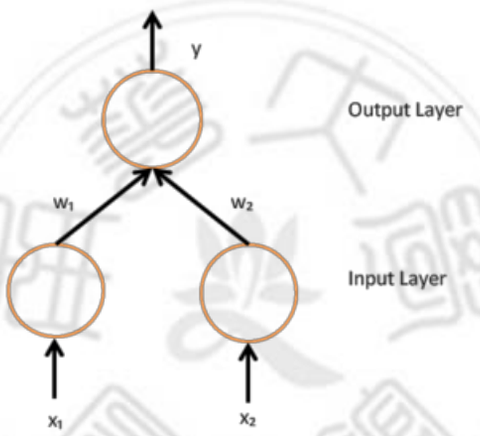


圖 4，模擬神經網路

表 3，模擬神經網路之輸入與輸出

Input 1	Input 2	Input 3	Output
0	1	1	1
1	0	0	0
1	0	1	1

2.2 卷積神經網路

卷積神經網路(Convolution Neural Network)，也被稱作 CNN 或 ConvNet，專門處理以網格狀方式排列的資料(例:影像)， 其中包含的像素值表示每個像素的亮度和顏色。CNN 如同人類大腦，具有處理豐富資訊的影 像之能力，其過程如圖 5，每個神經元像是負責在自己的工作區域觀察影像，並和其他神經 元協同合作，共同處理得到的數據後，得以辨識影像透露的訊息。卷積層會先檢測簡單的圖 案(例:線條、曲線)，接著再檢測複雜的圖案(例:臉、物件等)，透過神經元之間彼此共享參數， 可以幫助電腦擁有視覺的能力。

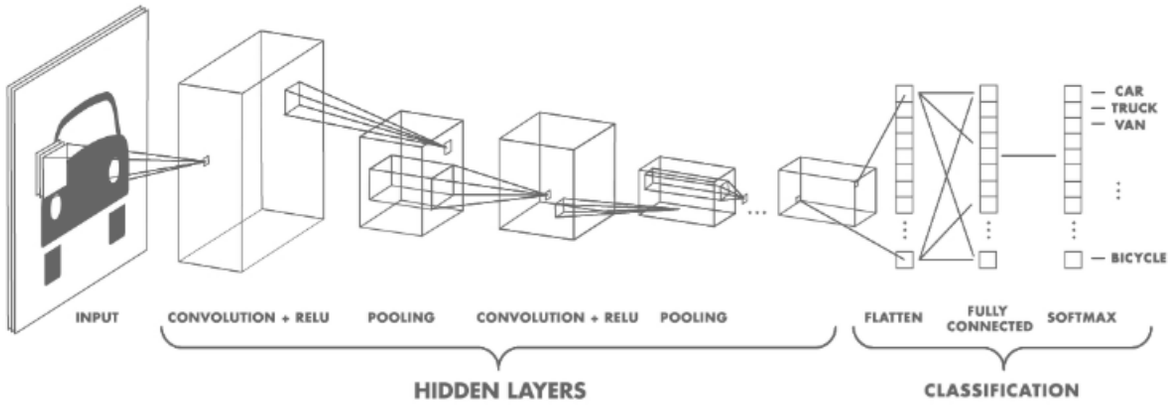


圖 5，CNN 架構[23]

神經網路內的影像，被表示為具有長度、寬度和深度的長方體，其中長度和寬度即是影像的尺寸大小，而深度則是影像色彩由三原色(紅、綠、藍)的 channel 所組成，如圖 6。

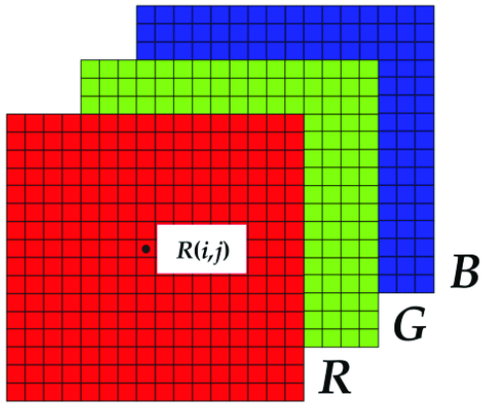


圖 6，神經網路內影像之表示

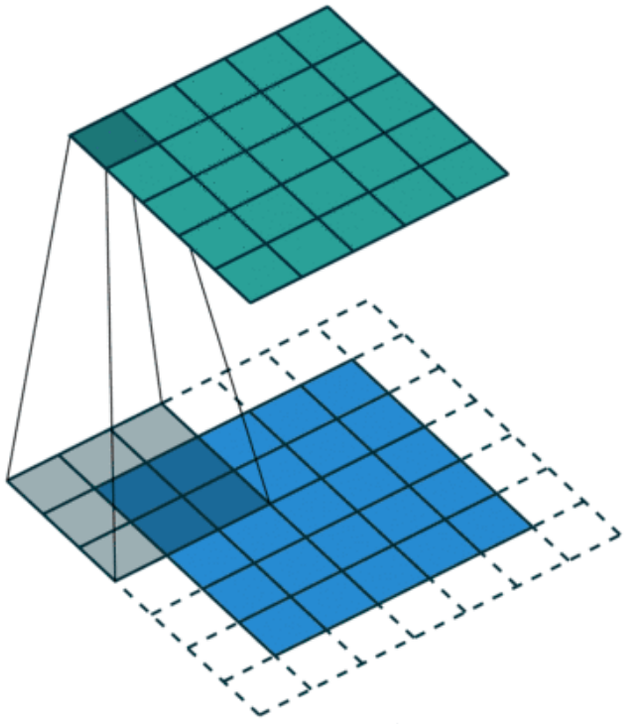


圖 7，神經網路卷積過程

接著從圖 7 可看見在此影像上會有一個 patch(圖 7 的 x)滑過整張影像，得到另一張有著不同長度、寬度、深度的影像(圖 7 的 y)，相較原本的 RGB 三通道，我們得到更多的 channel 數(K 個)，但長度寬度皆縮小，而此過程被稱作卷積(Convolution)，因為這個較原始影像小的 patch，我們得以用更少的參數量去完善神經網路。

卷積層由一組可學習的濾波器(filter，如圖 7 的 Patch)所組成，每個濾波器的長和寬都 很小，但深度和輸入影像相同(如果輸入層是影像，則為 3)。例如本論文必須在 256*256*3 的 圖像上進行卷積，則 filter 可能為 n*n*3，n 可以是 3、5、7 等數字，只要比影像尺寸小即可。在前向傳遞過程中，每個步驟稱為步幅(Stride)，會一步一步地在整張影像上滑動每個濾波器，計算濾波器和 patch 的點積，獲得每個濾波器的二維輸出後，將它們堆疊再一起即獲得深度等 於濾波器數量的輸出，而神經網路將學習所有濾波器所獲取的特徵。

CNN 其實是一系列的層所組成，每一層都透過可微分函數將輸入做體積轉換，輸入層以本論文 256*256*3 的原始影像輸入尺寸為例；卷積層負責濾波器和 patch 點積運算，假設濾波器數量為 12，會得到 256*256*12 的輸出體積;激勵函數層逐項地將激勵函數應用於卷積 層的輸出，輸出尺寸不變;池化層 (Pooling Layer)主要功能是減少輸入體積的大小，為了使運算量快速減少並防止過擬合現象 (Overfitting)，詳細的類型會在下面說明;最後是全連接層 (Fully-Connected Layer)，負責運算前一層輸出以獲得各類別的數值，輸出尺寸為 1 維的陣列 而大小即是類別的數量。

池化層(Pooling Layer)是在特徵映射圖(features map)上的每個 channel 上滑動一個二維的濾波器，用於整合濾波器掃描區域內的特徵。對於維度為 $n_h \cdot n_w \cdot n_c$ 的特徵映射圖來說，經過池化層後得到的輸出維度為

$\frac{(n_h-f+1)}{s}, \frac{(n_w-f+1)}{s \cdot n_c}$，其中$n_h$、$n_w$、$n_c$分別為特徵映射圖的長度、寬度、通道數，$f$為濾波器的大小，$s$則為步幅。池化操作包含以下類型：

- 最大池化(Max-Pooling): 為了積極對特徵圖進行縮小採樣，從輸入特徵圖中做採樣並輸出樣本的最大值，因此經過池化層後的輸出是包含前一個特徵圖中最顯著的特徵，實際操作 如圖 8：

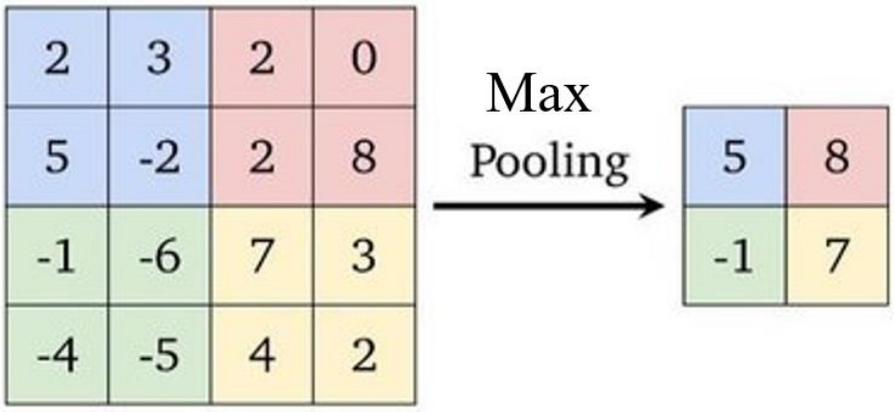


圖 8，最大池化範例

- 平均池化(Average-Pooling): 從輸入特徵圖中取每個 patch 的平均值來轉換，如圖 9：

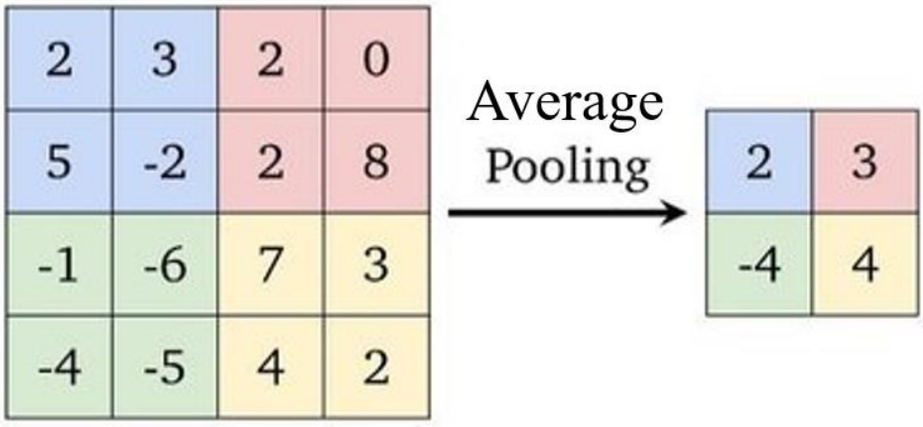


圖 9，平均池化範例

池化操作可以搭配步幅(Stride)去做調整，但就經驗法則來說，最大池化效果最佳，畢竟特徵即源自於影像中某些有特色的 pattern，為了避免意外流失寶貴的空間訊息，最合理的採樣策略是使用步幅為 1 的卷積層密集地掃描特徵圖。

n*n 的影像和 f*f 的濾波器經過卷積運算後產生的影像尺寸為(n-f+1)*(n-f+1)，例如 32*32 的影像和 3*3 的濾波器，產生的輸出為 30*30，每次經過卷積後就會縮小影像尺寸，因為影像減少到零之前的次數有限制，從而導致我們無法建構更深層的神經網路；此外，較接近邊緣的像素，使用率遠低於中間的像素，如圖 10 所示，可以看到 A 像素只被掃描到一次，B 像素為 3 次，而 C 像素則高達 9 次，這會導致影像邊緣的空間資訊被遺漏；綜上所述，為克服 這些問題，使用填補法(Padding)成為必要。

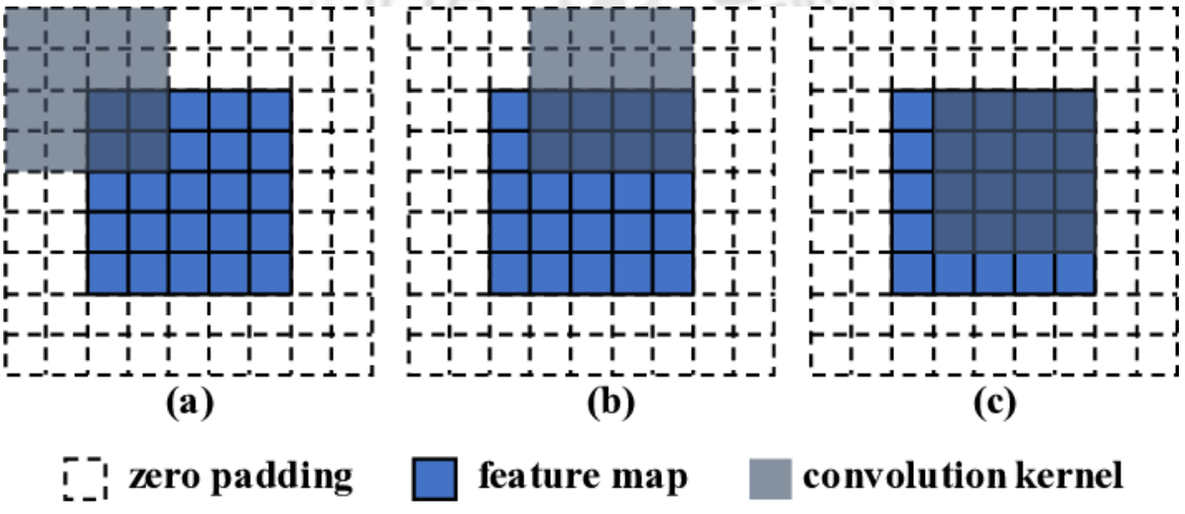


圖 10，遺失影像空間訊息示意圖

2.3 FCN 模型

在進入模型介紹之前，必須了解深度學習用於影像任務中的發展進程：

- 影像分類(Image Classification): 目標是能辨別出影像中的物體，在圖 11 的左上方可以看到，透過辨識為何種動物的機率，進而判斷該張影像為該類動物，CNN 為經典範例。
- 物件偵測(Object Detection/Localization): 當影像中有許多不同物體，我們需要使用 bounding boxes 的方框標示出每種物體所屬的類別，在圖 11 左下方可辨識出不同的三隻羊和落單的一隻狗，這同時意味著我們需要清楚知道每個物件的類別、位置、大小。
- 語意切割(Semantic Segmentation): 將影像中的每一個像素分類，意味著每個像素都有一個標籤(label)，同一類物件會被劃分在一起，如圖 11 中右上角。
- 實例切割(Instance Segmentation): 由語意切割的例子可以看見，三隻不同的羊被劃分為同一類別，不能很好地區分他們個別的形體。將 Semantic segmentation 和 Object detection/localization 融合得出 Instance segmentation，經典的例子為 Mask-RCNN。

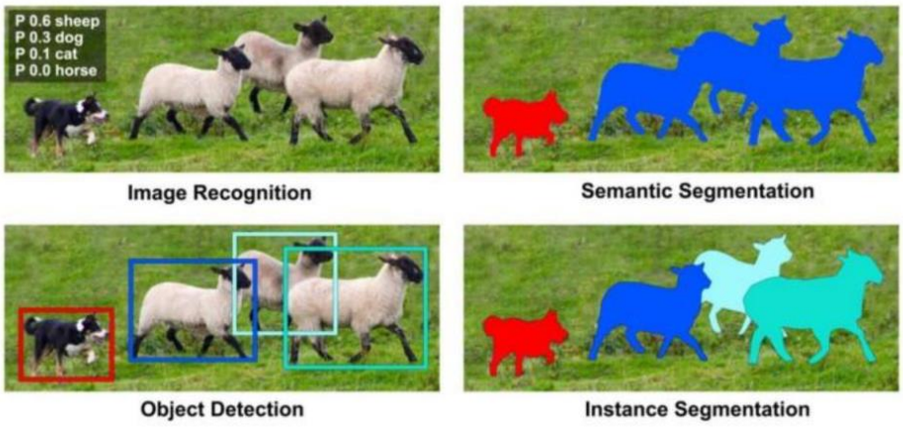


圖 11，影像任務演進之範例

2015 年，由 Long Jonathan 等人提出的 FCN (Fully Convolution Network)為影像語意切割 任務中的代表作，同時也是一種像素對像素的監督式學習方法，適用於任何尺寸的影像。圖 12 所示的 FCN 架構主要是由一個分類網路開發而來的，其中最後的全連接層(Fully- connected layer)被全卷積網路(FCN)所替代，使我們得以將影像中不同類別的物體切割出來。論文[25]提出的跳接(Skip)步驟，包含跳接層和雙線性插值，將分類網路的應用衍生到密集預測，從而在影像上實現精準的像素級切割。

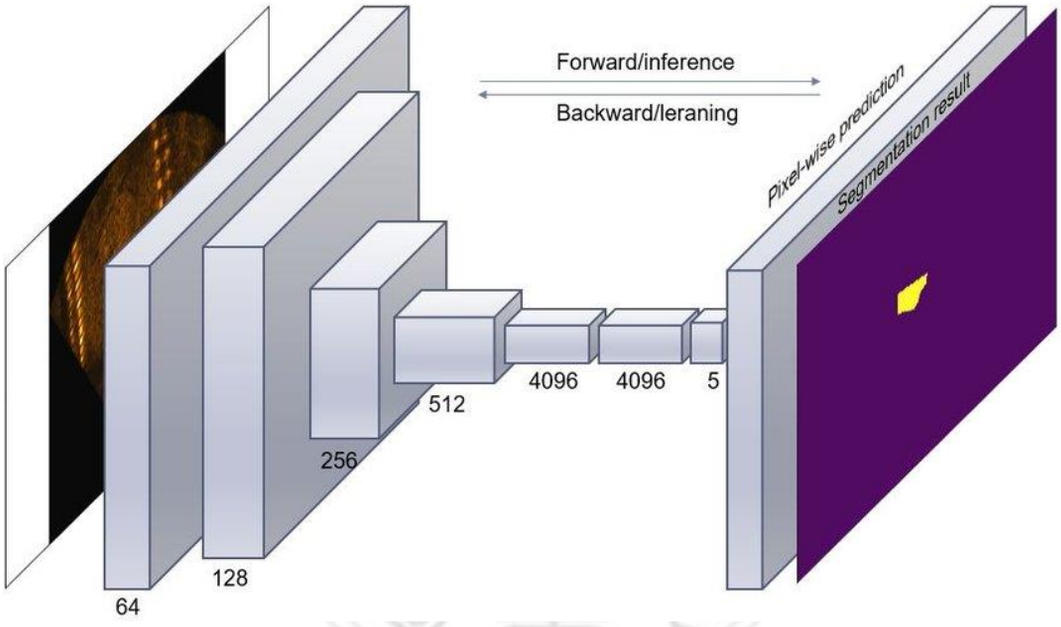


圖 12，FCN 架構

由圖 12 中，倒數第二層是經過 softmax 後有 5 層的深度(分為 4 類和背景 1 類)，每一層 就是每一類別中每個像素的機率。卷積是可使影像輸出尺寸縮小的過程，而反卷積(Deconvolution)，亦被稱作上卷積(Up-convolution)或轉置卷積(Transposed convolution)，則是應用於放大影像輸出尺寸。需切記反卷積並不是卷積的逆過程，舉例來說將 A 影像卷積後得到 B 影像，但 B 影像經反卷積後並不會重新回到 A 影像，因此其實反卷積被稱作轉置卷積更 為合適，Zeiler Matthew 等人也修正了其定義[26]，轉置卷積的過程和卷積一樣是透過參數學習，依據輸出特徵映射圖的梯度回推輸入特徵映射圖的梯度，下方利用 Dumoulin Vincent 等人的研究[27]幫助理解：

見圖 13，卷積是將 4*4 的特徵圖映射成 2*2，而轉置卷積則是將 2*2 的特徵圖映射成 4*4，濾波器的大小皆為 3*3，可以發現轉置卷積 zero padding 的數量為 $f-1=2$，其中 f 為濾 波器的大小，目的是為了恢復成輸入映射圖的尺寸。

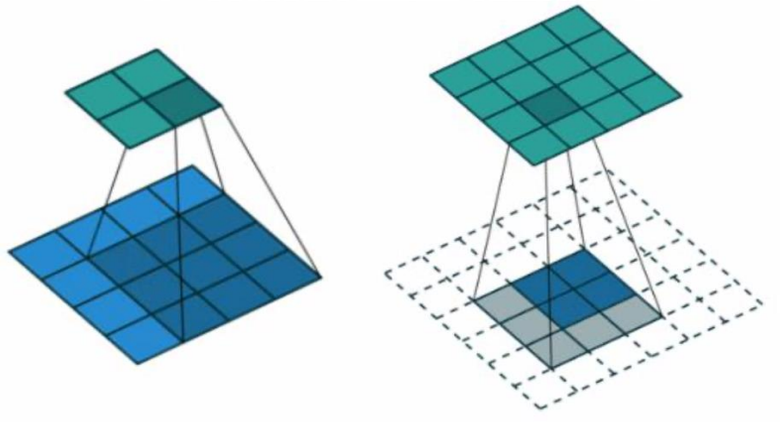


圖 13，卷積與反卷積示意圖[27]

接著見圖 14 的卷積，藍色部分為輸入，綠色部分為輸出，每 9 個輸入連接 1 個輸出。

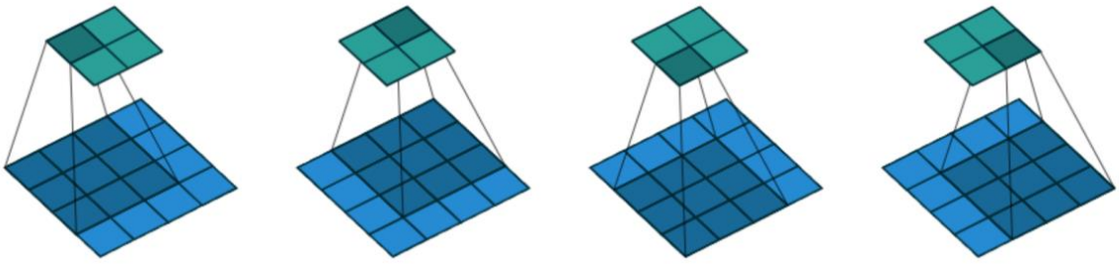


圖 14，卷積過程[27]

接著要思考如何將長度為 4 的向量 y 映射為長度為 16 的 x 向量，其實只要將 C 轉置，可表示 成$C^T y = x'$，見圖 2-15，每 1 個輸入連接 9 個輸出。

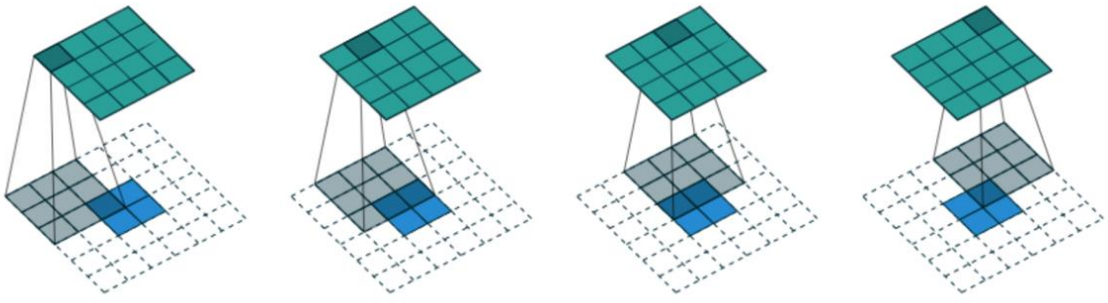


圖 15，反卷積過程[27]

FCN 還有一特別之處，會結合所有輸出，如圖 15 所示，經過 conv7 層後的影像輸出尺寸較小，經過 32 倍的上採樣(Up-sampling)後得到和輸入影像相同的大小，這樣相較粗糙的標籤圖(Label map)被稱作 FCN-32s。

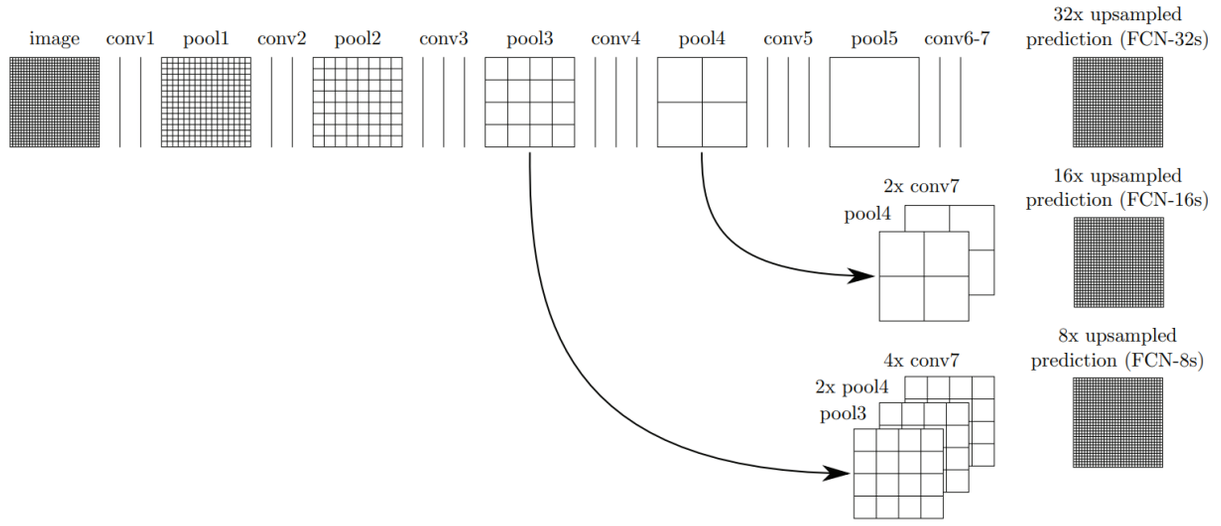


圖 16，FCN 結合所有輸出之架構[27]

因為當神經網路越深入時，雖然會得到更不易獲取的特徵，卻也會遺漏空間、位置的資訊，因此結合淺層輸出具有較多空間和位置資訊的特性，可以提升模型預測的表現，而論文[27]是透過逐項元素加法(Element-wise addition)結合輸出，如圖 二-17 所示，FCN-16s 是將 pool5 的輸出進行 2 倍上採樣，再和 pool4 結合進行 16 倍上採樣；FCN-8s 則是將 FCN-16s 進行上採樣前的特徵圖先進行 2 倍上採樣，再和 pool3 結合進行 8 倍上採樣，而根據經驗法則，FCN-8s 通常效能最佳。

DenseNet 由密集模塊(Dense blocks)和池化操作(Pooling operation)所建立[28]，其中每個密集模塊是先前特徵映射圖的迭代連接(iterative concatenation)，如圖 17 所示，同時也被視為是 ResNets 的衍伸版本[29]。

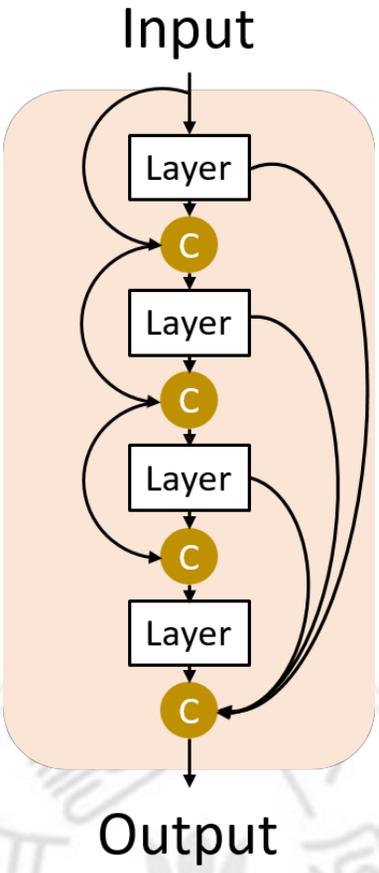


圖 17，密集模塊(Dense block)

起初 DenseNet (如圖 18)普遍被使用於影像分類的任務中，其具有三項優點：模型參數使用率更高、可透過捷徑(Short paths)實現隱性的深層監督、所有的層(Layer)可獲取先前特徵映射圖的資訊，增進特徵重複使用率，也因此被修改使用於語意切割的任務中。

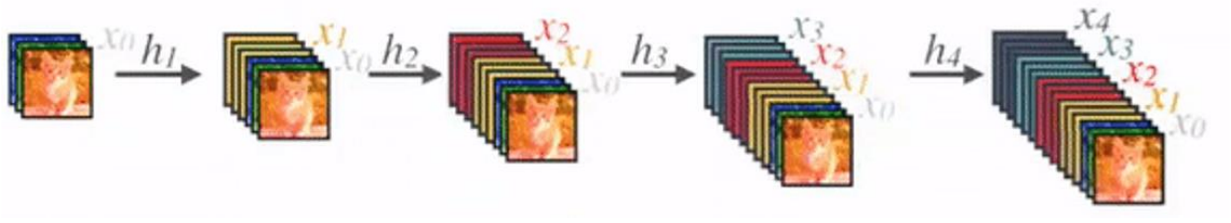


圖 18，DenseNet 的 Concatenation 示意圖[30]

由於每一層都接收來自前幾層的特徵映射圖，因此通道數可以更少，讓神經網路更輕薄、緊湊，如圖 19 所示。運用這種特性組成新的網路架構，如圖 20 所示，TD 表示下採樣 (Down-sampling)，而 TU 為上採樣(Up-sampling)，只有對先前密集模塊生成的特徵映射圖進行上採樣，而不是對每層 Layer 做上採樣，否則會耗費大量內存資源，具體 TD 和 TU 內容如圖 20。

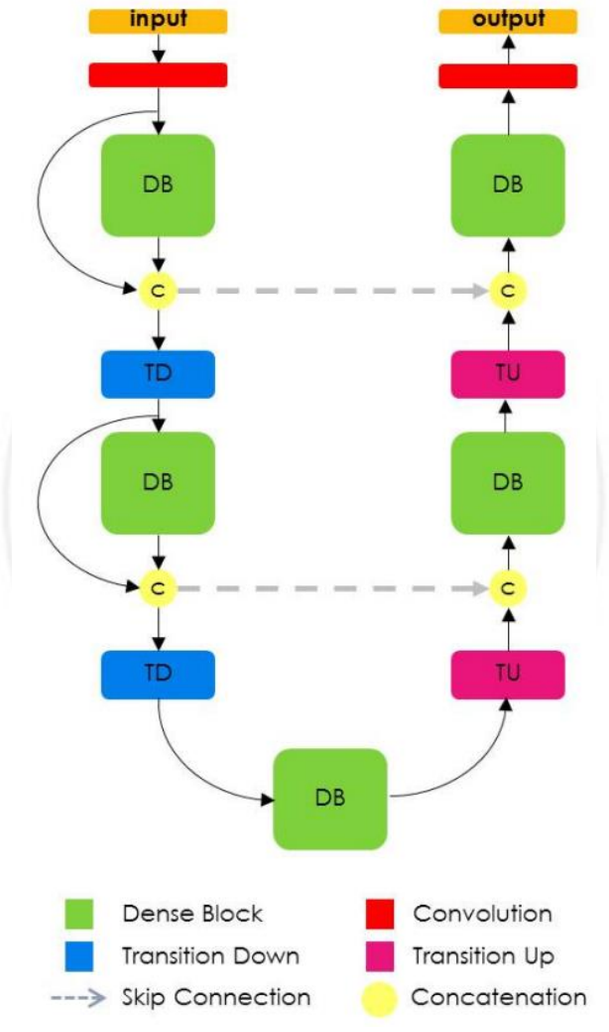


圖 19，FC-DenseNet 架構圖[31]

Transition Down (TD)
Batch Normalization
ReLU
1×1 Convolution
Dropout $p = 0.2$
2×2 Max Pooling

Transition Up (TU)
3×3 Transposed Convolution $stride = 2$

圖 20，TD 和 TU 內容[31]

使用神經網路訓練時，在輸入模型前會先將所有像素進行正規化(Normalization)，目的是提升模型收斂的速度，而若使用 Batch Normalization 則表示在一層輸入前都會執行一次正規化，每個批次會變成平均為 0 而標準差為 1 的常態分佈，在使用 Sigmoid 函數時才能有效地被傳遞到下一層，解決梯度消失的疑慮。

2.4 UNet 模型

UNet 於 2015 年由 Ronneberger Olaf 等人所提出，基本架構由兩條路徑所組成：第一條是收縮路徑(Contracting Path)，也被稱為編碼器(Encoder)，跟正常的卷積網路類似；第二條是擴展路徑(Expansion Path)，也被稱為解碼器(Decoder)，由上卷積和串聯(Concatenation)來自收縮路徑的特徵所組成，可以提高輸出的解析度，傳送到最終的卷積層以創造出完全切割的影像。UNet 網路架構幾乎是對稱的，呈現 U 形狀也是名稱之由來，大多數的卷積網路主要是將整張影像分類為某一類別，無法提供生醫影像分析中必要的像素等級資訊，UNet 和 FCN[27]的橫空出世推進了語意切割領域的進展。

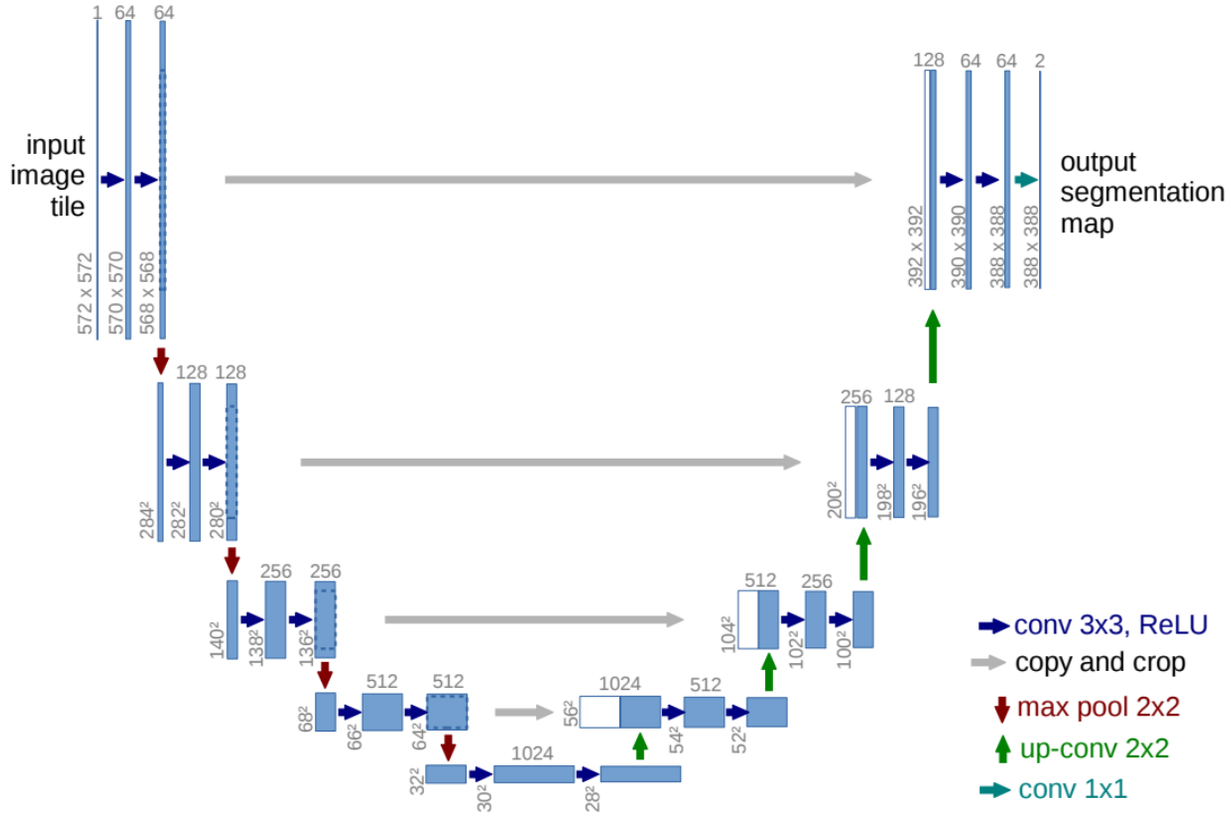


圖 21，UNet 架構圖[32]

UNet 特別之處在於能使用相當少量和有限的訓練樣本得到高度清晰的切割圖，這項特點在生醫影像領域中十分重要，因為正確標註的影像通常是有限的，且容易受限於專業人士的經驗，外加標註時間成本過高。

整理架構如圖 21 所示，左側為收縮路徑，負責做 Down-sampling，壓縮影像尺寸並執 行特徵萃取，每個模塊由兩個連續的 3*3 卷積、ReLU 組成，再經過最大池化層(Max Pooling Layer)，這樣的設計會重複多次；右側為擴展路徑，負責 Up-sampling，增加解析度並執行插值 (Interpolation) 到原影像的尺寸，以利提高定位精準度，使用 2*2 上卷積對特徵映射圖進行上採樣，以及將收縮路徑中相應層的特徵映射圖透過 Skip connection 串聯(concatenate)到擴展路徑上採樣的特徵映射圖上到最後階段，應用 1*1 卷積將特徵映射圖減少到所需的通道數目並輸出切割影像。

UNet 中間的四條灰線為 Skip connection，可以幫助神經網路獲得空間資訊，其中裁剪(Crop) 被認為是必要的，因為邊緣的像素特徵所含有的語意資訊較少，丟棄後可增加訓練速度。

ResUNet 是基於 ResNet[29]的架構所衍伸出來的 UNet 變體。最初的動機是為了克服訓練深度神經網路的困難，當層數越多時，神經網路往往能夠更快速地收斂，然而實驗結果發現單純增加層數會導致飽和，進而造成效果不彰。這種衰減是因為權重向量中的梯度減少，導致深層神經網路丟失學習到的特徵，ResUNet 透過 skip connection 來減緩梯度消失的問題，將先前所獲取的特徵映射圖添加到更深的層，這種方式能更完善地保留更深層神經網路中的特徵映射圖，也推進 UNet 模型得以設計更深層的網路，每個殘差單元(Residual Unit)可以被表示為：

$$y_l = h(x_l) + F(x_l, w_l) \tag{2.7}$$

$$x_{l+1} = f(y_l) \tag{2.8}$$

此處 x_l 和 x_{l+1} 為殘差單元的輸入和輸出，$F(\cdot)$ 為殘差函數(Residual function)，$f(\cdot)$ 為激勵函數 (Activation function)，$h(\cdot)$ 則為恒等映射函數 (Identity mapping function)，如圖 22 所示：

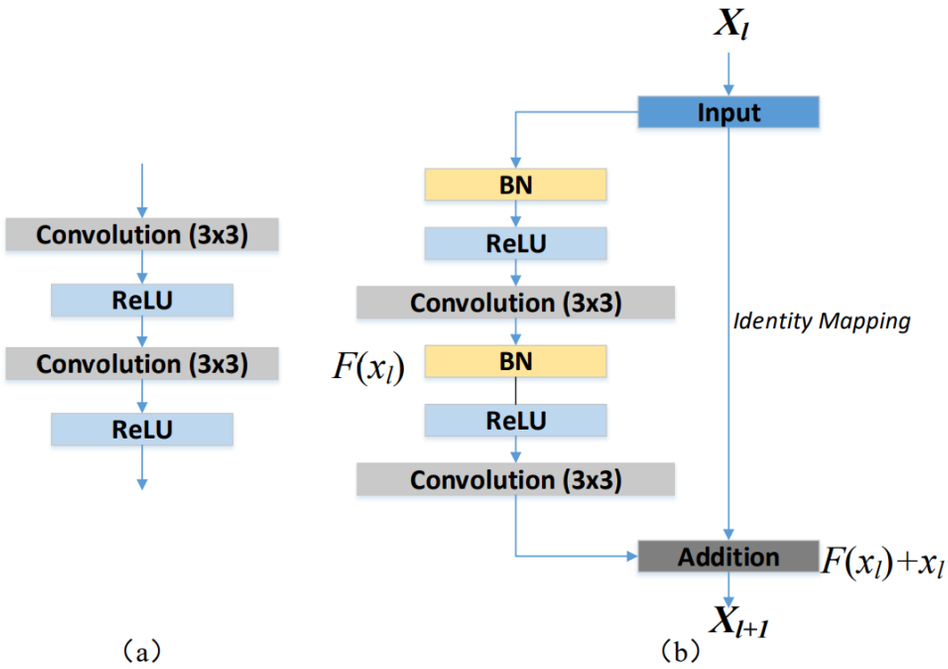


圖 22，(a) UNet 使用之神經元 (b)ResUNet 使用之神經元[33]

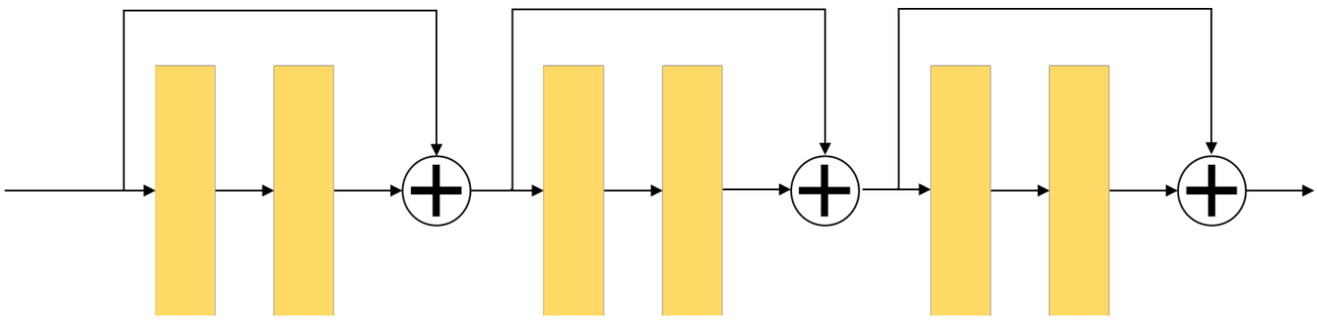


圖 23，有 skip connection 的 ResNet blocks [33]

較為常見的 ResNet 是跳過兩層(如圖 23)或三層，這種架構也被廣泛應用於許多生醫影像的論文並獲得奇效，被認為是複雜影像分析的理想選擇，ResUNet 完整架構如圖 24：

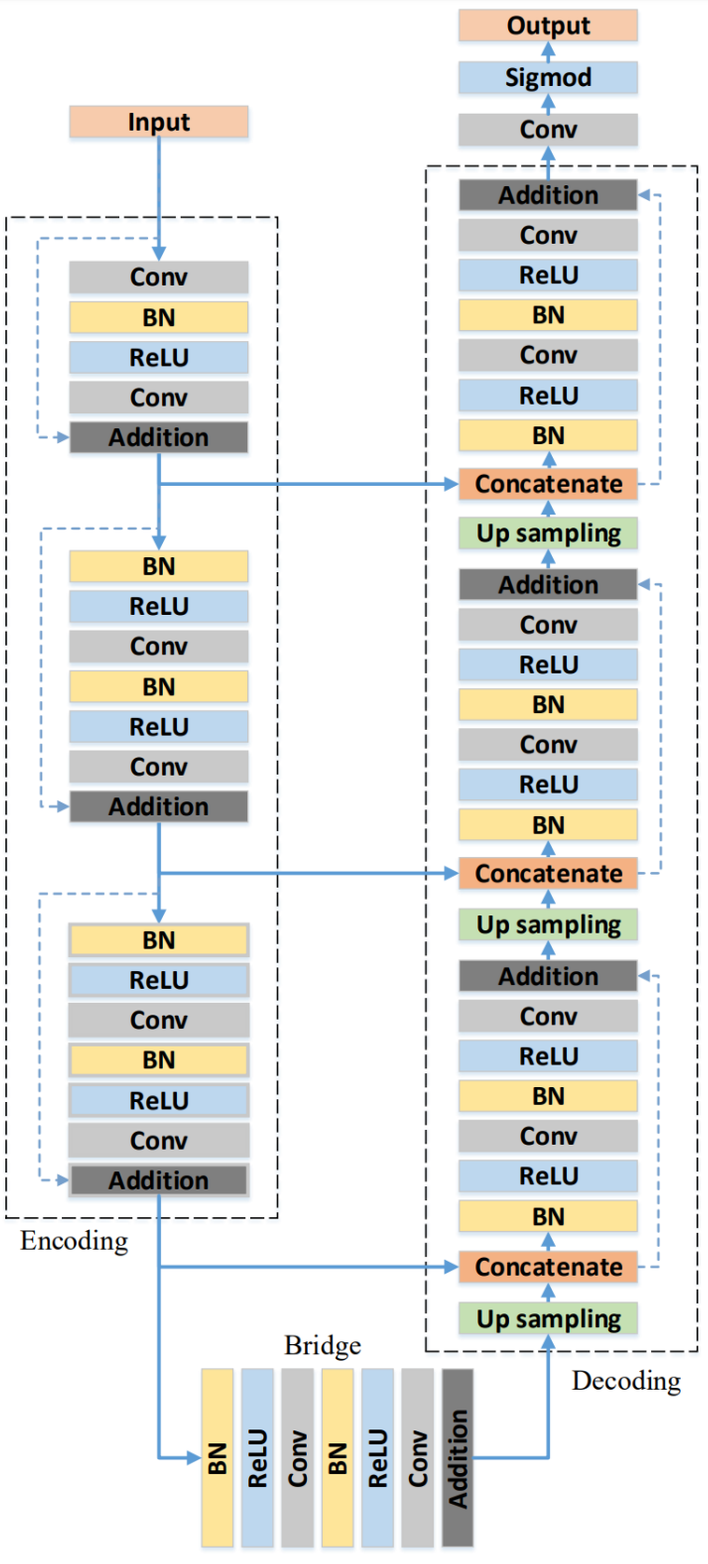


圖 24，ResUNet 架構圖 [33]

Attention UNet[34]利用深度神經網路對影像分析通常需要能夠專注於重要的對象而忽略不必要的區域，而 Attention UNet 透過注意力閘(Attention Gate)來實現這目標。注意力閘可以將與切割任務無關的特徵都修剪掉，在 UNet 架構右邊的擴展路徑都有一個注意閘，中間 Skip connection 之處也會將收縮路徑的特徵映射圖過濾一遍，再和擴展路徑的部分串聯起來，不僅提高了切割效果也不會造成模型太多的計算負擔。

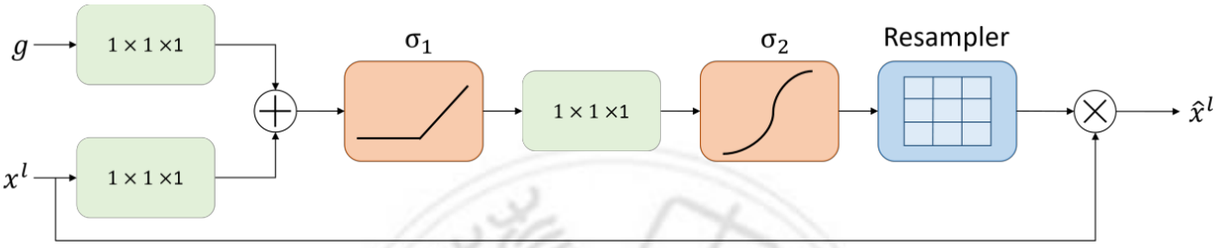


圖 25，注意閘(Attention Gate)示意圖[34]

圖 25 中，輸入訊號 (x^l)和門控訊號 (g_l) 都經過 $1\times1\times1$ 卷積，接著將訊號相加並進行一系列的 線性變換: ReLU(σ_1)、$1\times1\times1$ 卷積、Sigmoid(σ_2)和重新採樣器 (Resampler)，最後連接到原始輸入後得到輸出。

注意單元(Attention Unit)在端對端訓練的模型(如：UNet)很有幫助，因為可以提供局部的分類，允許網路的不同部分各自專注於切割不同的對象;此外，透過正確標註的訓練資料，網路可以調整影像中鎖定的對象，注意閘應用一個注意係數 (Attention Coefficient)，可以根據每個類別對特徵映射圖進行加權，可以將神經網路調整為關注特定的類別[35]。雖然有不同類型的注意閘，但加性注意閘在影像處理中更受到推崇，因為可以提高準確率，公式表示如下：

$$q_{att}^{l} = \varphi^{T}\left(\sigma_1\left(w_x^T x_i^l + w_g^T g_l + b_g\right)\right) + b_\varphi \tag{2.9}$$

$$\alpha_i^l = \sigma_2\left(q_{att}^l\left(x_i^l, g_l\right); \vartheta_{att}\right) \tag{2.10}$$

其中x_i^l是來自從收縮路徑學習到的特徵，g_l為由 skip connection 傳的輸入，b_g和b_φ為 bias 的修正項，ϑ_{att}為一系列線性變換的表示，$\sigma_2(x_{i,c})$為 sigmoid 函數，公式如下:

$$\sigma_2(x_{i,c}) = \frac{1}{1+\exp(-x_{i,c})'} \tag{2.11}$$

圖 26 的F_n為通道數，N_c為類別數。Attention UNet 和 UNet 的區別只在於解碼器 (Decoder)，在進行解碼前將從編碼器(Encoder)提取的特徵進行注意閘，重新調整解碼器的輸 出特徵，引此能用極少的計算量帶出模型準確率、敏感度的提升。

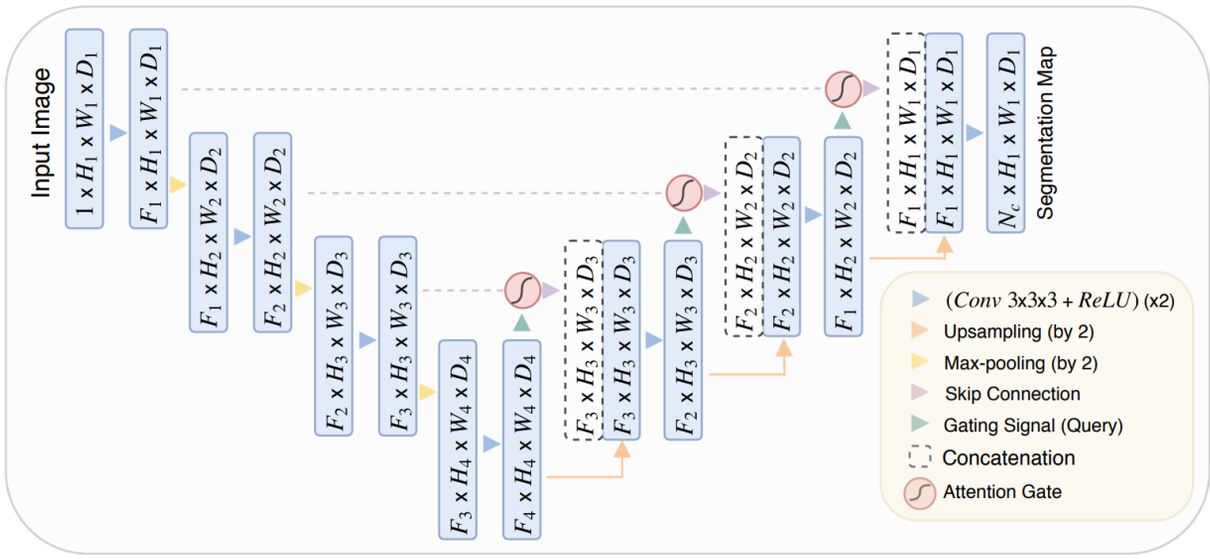


圖 26，Attention UNet 架構圖[34]

第3章　模型發展

3.1 實驗流程圖

本論文提出之 U-Net 與其他六種比較的模型之實驗流程如圖 27，將訓練好的模型所預測的人數與 Ground truth 比較，在經過錯誤評估指標即可得知模型用於此熱像儀人流計數影像資料集的優劣。

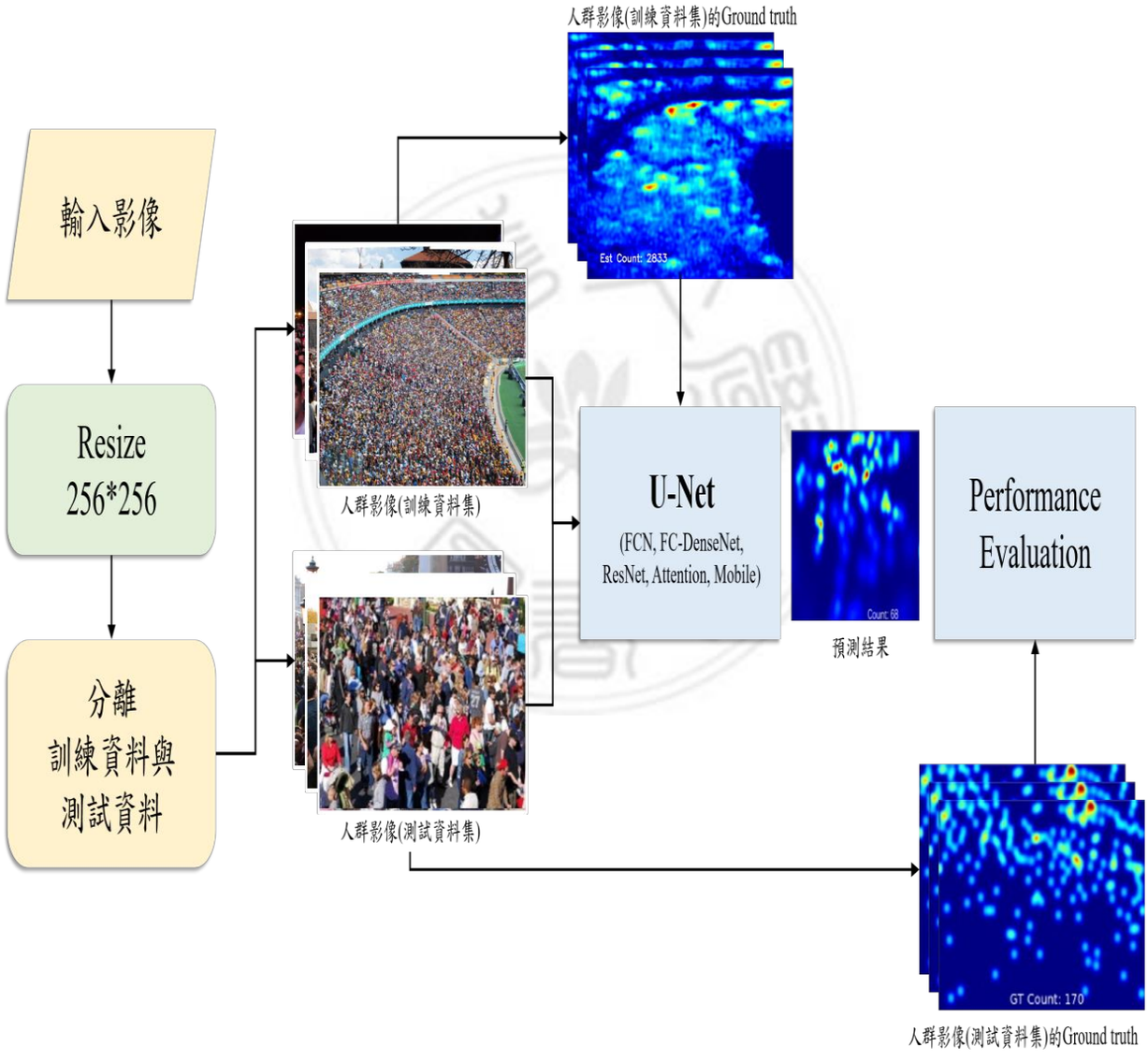


圖 27，本論文之實驗流程圖

3.2 公開資料集

本論文所使用之人群影像 ShanghaiTech 數據集由[18]收集和發布，由四部分組成，資料總數如表 4 整理。A 部分分別由 300 張和 182 張不同分辨率的圖像組成，分別用於訓練和測試。最小和最大計數分別為 33 和 3139，平均計數為 501.4。B 部分分別由 400 張和 316 張具有獨特分辨率 (768×1024) 的圖像組成，用於訓練和測試。與 A 部分相比，這些圖像中的人數要少得多，最小和最大計數分別為 9 和 578，平均計數為 123.6。UCF_QNRF 數據集 [19] 包含 1,535 張高質量圖片，其中 1201 張用於訓練，334 張用於測試。 最小和最大計數分別為 49 和 12,865，平均計數為 815。UCF_CC_50 數據集 [35] 包含 50 張圖像，最小和最大計數分別為 94 和 4,534。由於圖像數量有限，這是一個具有挑戰性的數據集。遵循 [35] 和許多其他工作中的建議，我們在實驗中使用交叉驗證，程式碼如圖 28，資料細節如圖 29、30、31 與 32。

表 4，影像資料庫之數量及統計

資料集	ShanghaiTech (A)	ShanghaiTech (B)	UCF_QNRF	UCF_CC_50	Total
訓練資料 (張)	300	400	1201	-	1901
測試資料 (張)	182	316	334	50	882

```
### Train/val split ###
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.1, random_state=3, shuffle = True)
x_train, x_val, y_train, y_val = train_test_split(x_train, y_train, test_size=0.2, random_state=3, shuffle = True)
```

圖 28，train_test_spilt 函式程式範例

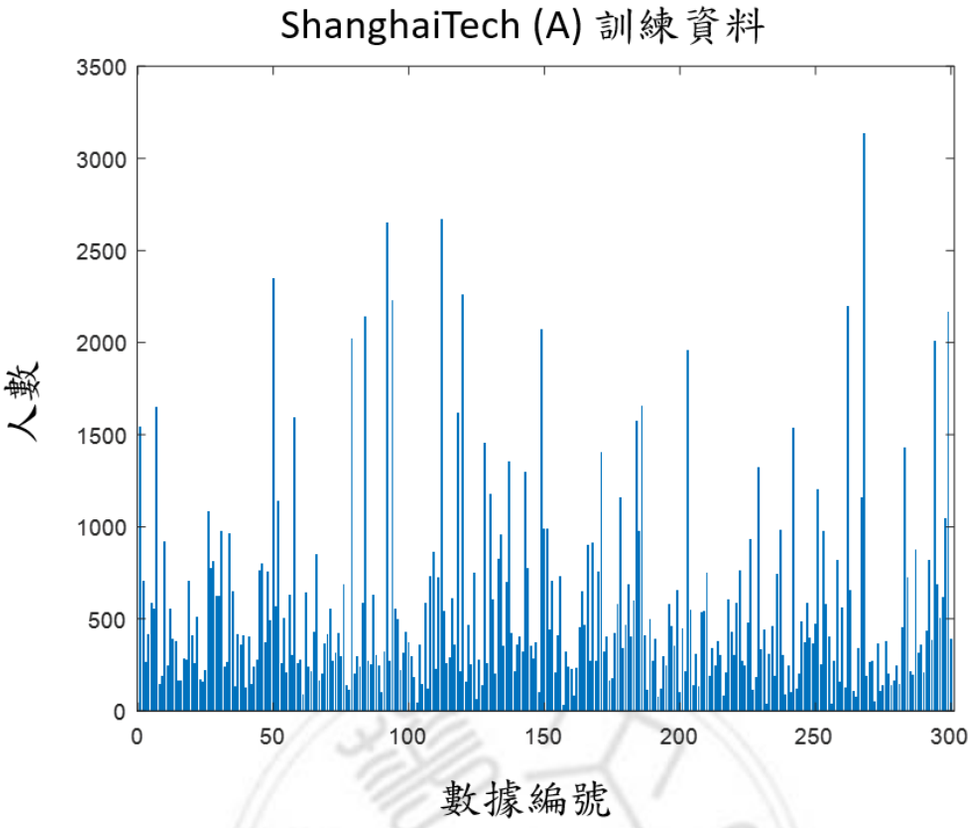


圖 29，ShanghaiTech (A) 訓練資料

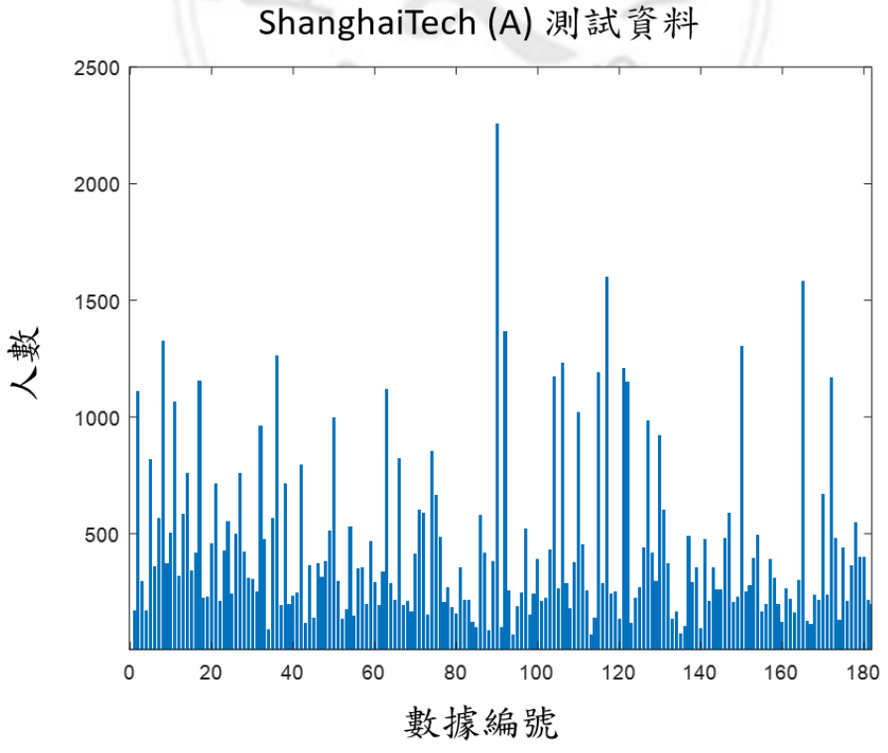


圖 30，ShanghaiTech (A) 測試資料

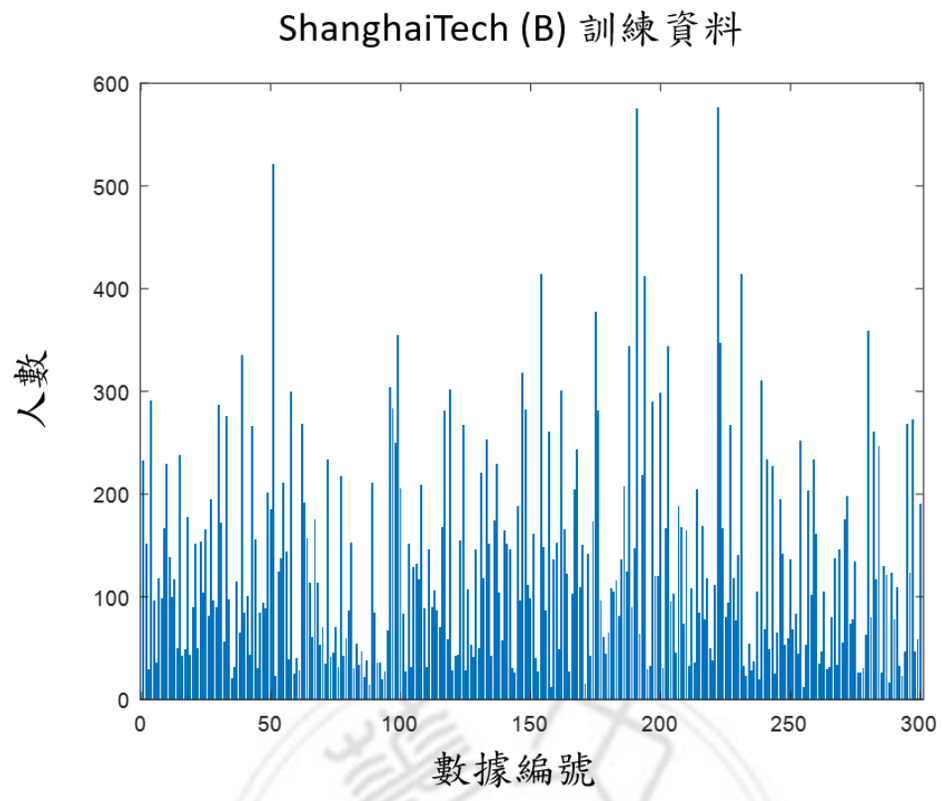


圖 31，ShanghaiTech (B) 訓練資料

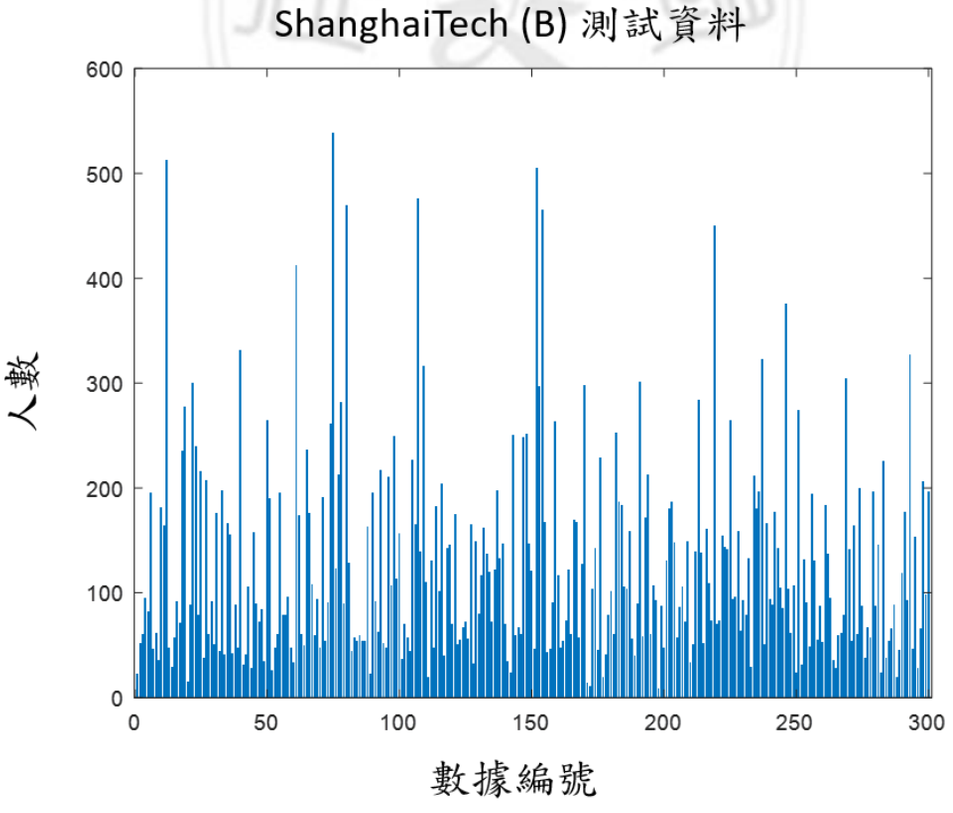
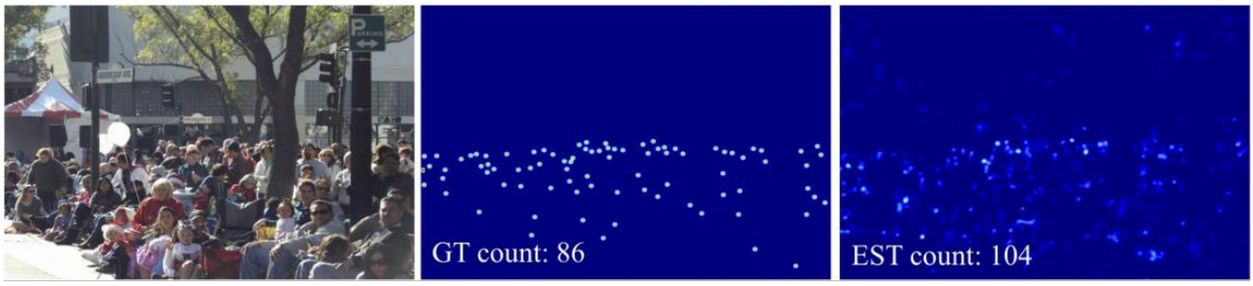
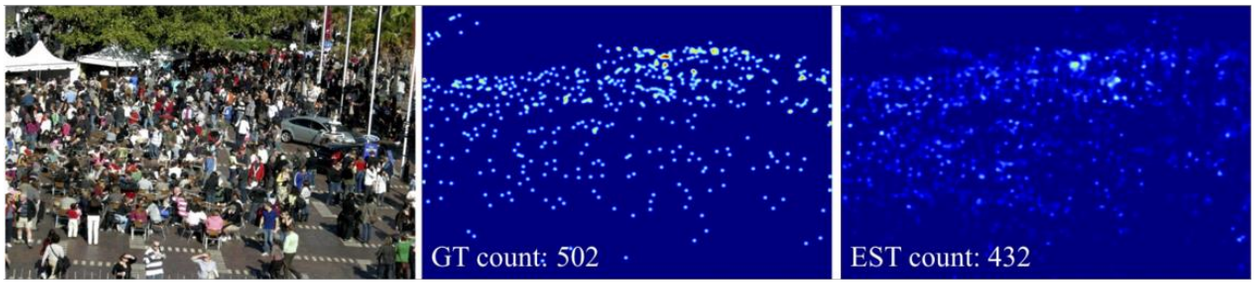


圖 32，ShanghaiTech (B) 測試資料

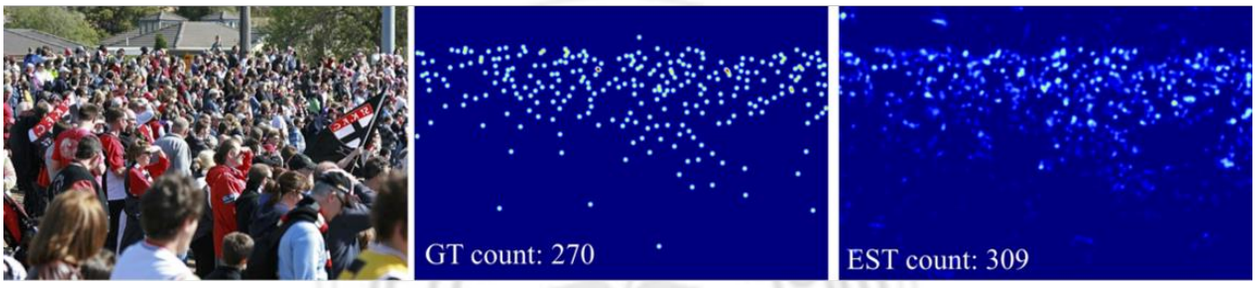
ShanghaiTech 影像資料庫大致上可分為五種難易度：難、偏難、中、偏易、易，如圖 3-7 所示，左邊欄位各提供一張範例影像說明：(a) 難度為易，人形明顯，可用肉眼判斷輪廓明確，且人旁周圍沒有陰影；(d) 難度為偏易，人形明顯，可用肉眼判斷輪廓明確，但人數偏多，需解析度高；(g) 難度為中，可大致觀察到輪廓，但人群擁擠，用肉眼判斷稍顯困難；(j) 難度為偏難，必須由現場確切計數，且由經驗豐富的專家協助判斷；(m) 難度為難，人群擁擠，且人旁周圍有陰影，影響模型對於人形的判斷。



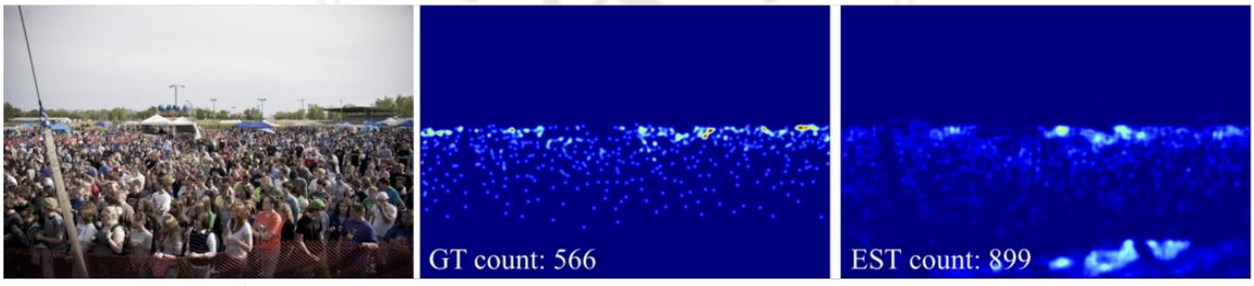
(a) 易　(b) (a)之Ground Truth　(c) (a)之預測值



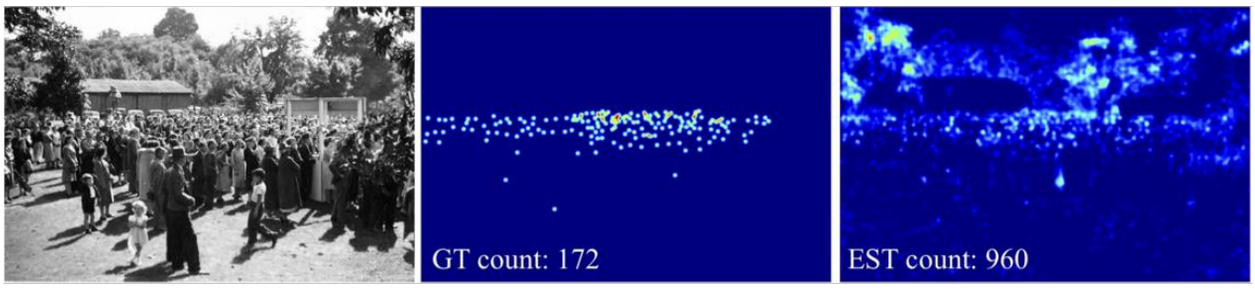
(d) 偏易　(e) (d)之Ground Truth　(f) (d)之預測值



(g) 中　(h) (g)之Ground Truth　(i) (g)之預測值



(j) 偏難　(k) (j)之Ground Truth　(l) (j)之預測值



(m) 難　(n) (m)之Ground Truth　(o) (m)之預測值

圖 33，各難易度人群影像之 Ground Truth 範例

3.3 神經網路模型

在本節中，我們首先回顧基於 CNN 的人群計數的相關文獻，並專注於本研究提出的人群計數模型並比較不同網路架構。緊接著，我們介紹相關先備知識，以及如何應用於人群計數模型。人群計數的常用網絡設計原則包括多列網絡 (multi-column networks)、豐富的特徵融合 (feature fusion) 和注意力機制 (attention mechanism)。

多列神經網絡被用來解決人群計數中的尺度變化問題[18, 36, 37]。作為最早的基於 CNN 的人群計數模型之一， MCNN [18] 由三個分支組成，目標為處理不同密度的人群。繼承這個想法，[24]提出了 SwithCNN，發展出一種分類器根據不同人群的密度，選擇三個分支中的其中之一。雖然這些方法目標在不同的分支中使用不同的卷積大小來捕獲尺度變化訊息，但[38]提出了一個模型，該模型由 VGG16 網絡的多個分支組成，具有共享權重，分別處理縮放的輸入圖片。同樣[39]設計一個兩列網絡，它通過 CNN 的兩個分支迭代學習低分辨率和高分辨率密度圖。這些專門設計的網絡架構的成功驗證了多列 CNN 模型能夠捕獲用於人群計數的尺度變化特徵。

網絡設計的第二項目標是針對豐富特徵值的有效融合[40, 41]。不同的特徵融合策略包括直接融合[40]、自上而下融合[42]和雙向融合[43]已被用於人群計數。利用 Inception 模組[26]可以達成兩種特徵值萃取，該模組在[44]中首次提出，已演變成各種更有效的方式。之前在 SANet[45]和 TEDNet[41]中，Inception 模組已用於人群計數模型。在本研究中，目標在探索 U-Net 框架。

在設計用於人群計數的網絡架構時，注意力機制是另一種有用的技術[21, 23, 40, 46]。注意力機制層通常與多列結構相結合，以便不同理解訊

息(例如背景、稀疏、密集等)的區域可以分別由不同的分支卷積和處理。這些模型的注意力特徵圖已被證明可以捕捉人群密度訊息[23]，但此特徵圖無法提供圖片中更微小刻度的訊息。為了明確解決此問題，透視圖被應用於準確估計密度圖[47-49]。然而很多情況下透視圖是不能被使用的，所以需要專門設計和訓練網路從圖片中模擬這些透視圖[50]，或通過多任務學習從二元分割特徵圖訓練人群計數。

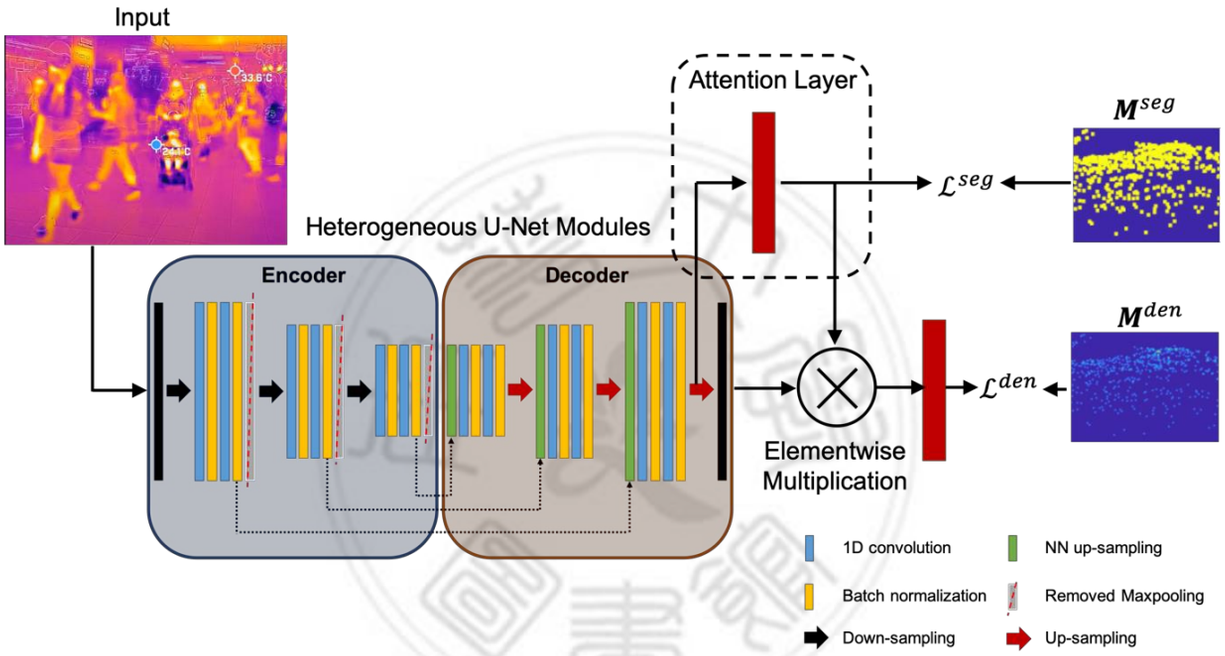


圖 34，分割與注意力神經網路框架，改編自 U-Net (1) 去除 maxpooling 層，(2) 最後加上採樣層 ，(3) 添加注意力層 ，(4) 添加卷積層用於密度估計圖。

在本研究中，二元分割特徵圖被使用為顯著的視覺特徵。從不同角度的總結，本研究與[51]與[52]更相關，其中二元分割特徵圖也用作注意力機制層，但方式完全不同，如圖 34 所示。

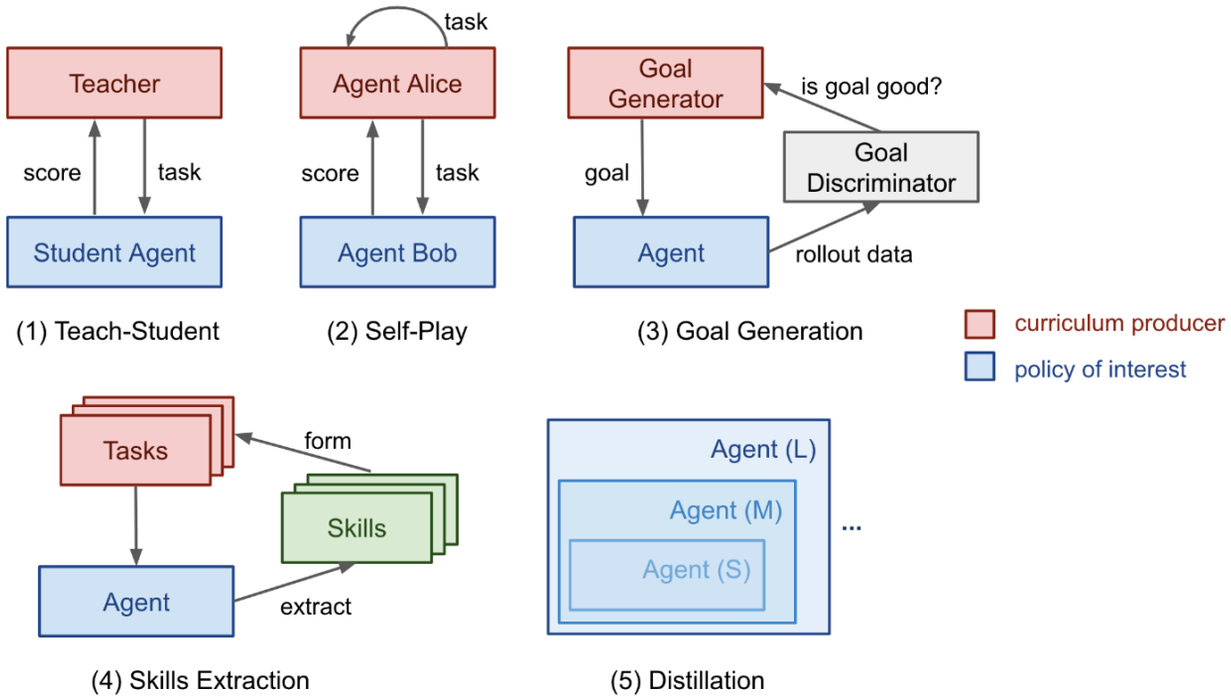


圖 35，五種課程式學習[53]

課程式學習是機器學習中模型訓練的一種策略 [54]。課程式學習(如圖 35)的想法可以追溯到 1993 年，當時 [53] 證明訓練神經網絡通過“從小處開始”來學習簡單語法是有效的。課程式學習策略的靈感來自於人類如何從簡單的概念逐漸學習到難以抽象的知識。在訓練機器學習模型的特定情況下，課程式學習在訓練開始時選擇簡單的示例，並允許將較難的示例逐漸添加到訓練集中。課程通常定義為通過一些先驗知識對訓練示例進行排序，以確定給定示例的難度級別。[55] 通過將原始課程學習和自定進度學習 [56] 的思想整合在一個統一的框架中，將課程學習擴展為所謂的自定進度課程學習。

在這項工作中，我們在人群計數中應用課程學習策略來解決圖像中人群密度差異較大的問題。課程式學習已在 [57] 中用於人群計數，其中課程是在圖像級別設計的，即為每個訓練圖像計算難度分數。訓練圖像根據難度分數分為多個子集，最簡單的子集首先添加到訓練集中。相比之下，

我們的課程學習策略的特點是在像素級別定義了一種新的課程損失，如第 3.4 節所述。我們定義高於閾值的密度圖像素具有更高的難度分數，因為這些像素位於更密集的人群區域內。我們在整個訓練過程中使用所有訓練圖像，但在開始時將閾值設置為較低的值並逐漸增加它，使困難的像素變得容易並且對訓練有更大的貢獻。因此，我們的課程學習策略易於實施，零額外成本，並且已被證明是有效的，尤其是當圖像中存在極其密集的人群區域時。

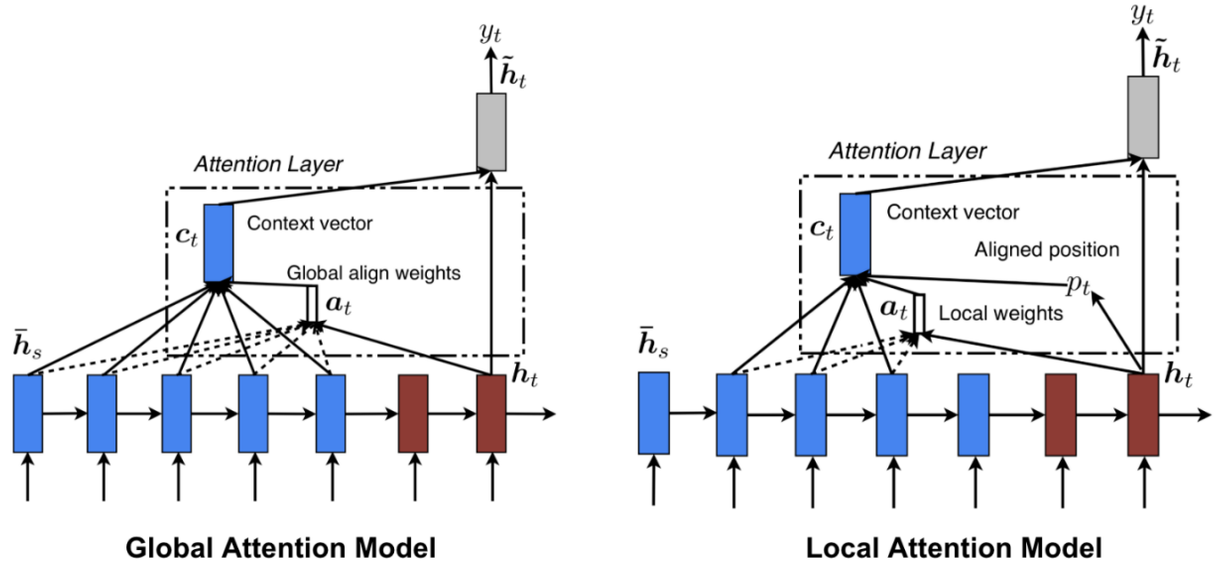


圖 36，注意力機制圖解[58]

在本研究中，密集的人群計數被表述為密度圖回歸問題。給定人群圖像 I，我們的目標是學習表示為 $\mathcal{F}$ 的全卷積網絡 (FCN)，便可以透過以下方程式估計相應的密度圖 $\boldsymbol{M}^{\text{den}}$：

$$\widehat{\boldsymbol{M}}^{\text{den}} = \mathcal{F}(I; \boldsymbol{\Theta}) \tag{3.1}$$

其中$\boldsymbol{\Theta}$全卷積網絡的集和參數。

如圖34所示，我們提出的網絡改編自著名的 U-Net ，最初由 Google Research [26] 設計用於圖像分類。我們首先將 U-Net 修改為 FCN，以便它可以處理任意大小的圖片並生成估計的密度圖M^{den}作為輸出。網絡中添加

了一個注意力層[58] (如圖36) 以過濾掉背景區域內的特徵，並專注於前景特徵以進行精確的密度圖估計。由於此注意力層生成的注意力圖目的在區分特徵圖的背景和前景區域，因此我們使用可以從點標籤中分別出的地面真實分割圖作為注意力層訓練的額外指導。最後在訓練過程中，函數將注意力圖近似於分割圖。

本研究另外運用課程式學習中的損失函數在人群計數的模型中使用。具體來說，我們定義一個基於像素難度級別的課程，以便網絡通過優先加強注密度圖中的簡單區域（稀疏）並降低複雜區域（密集）的權重來開始訓練。在訓練過程中，逐漸將複雜區域給模型認識，最後模型將能在所有情況都能表現良好。

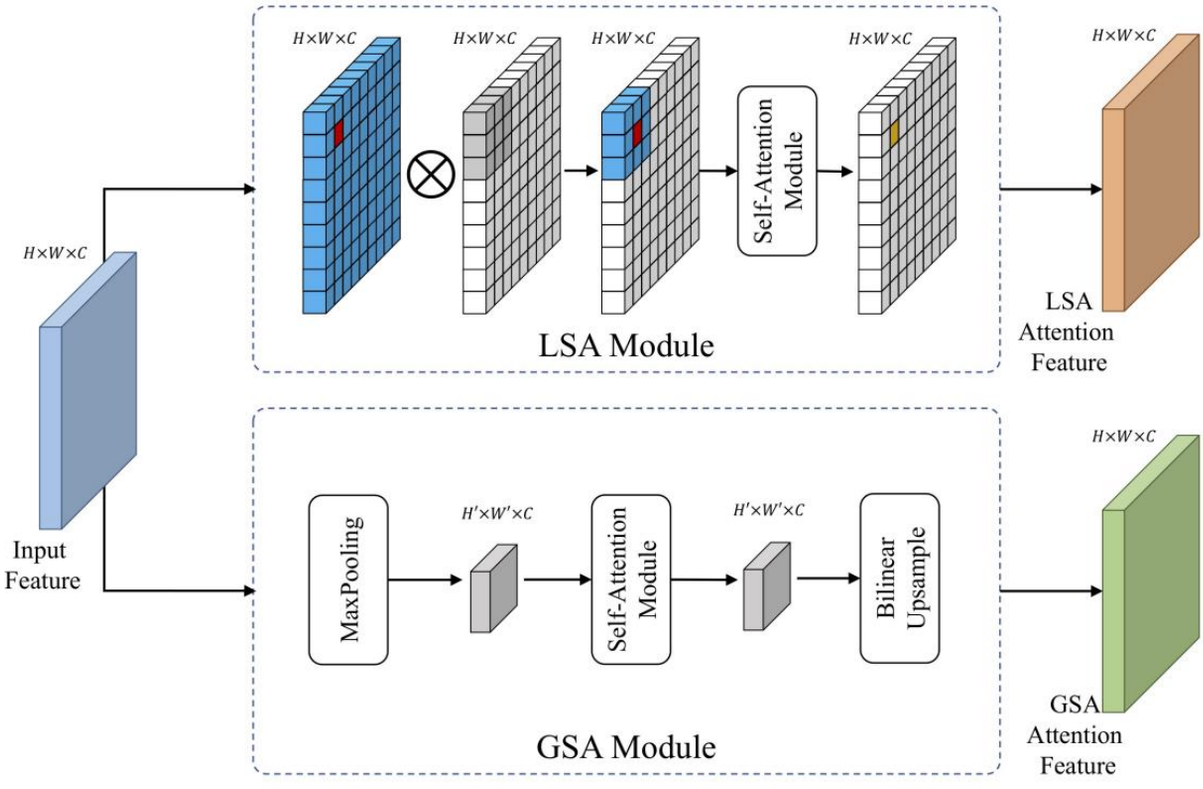


圖 37，密度與分割圖

在這項研究中，儘管更複雜的方法 [52] 可能有利於性能，但我們使用較簡單的方法生成密度和分割圖(如圖37)。密度圖$\boldsymbol{M}^{den} \in \mathbb{R}^{H\times W}$ 其中H和W是圖片的高度和寬度，我們遵循 [18] 使用高斯核心 $G_\sigma \in \mathbb{R}^{15\times 15}$ 並固定$\sigma = 4$：

$$\boldsymbol{M}^{den}(x) = \sum_{i=1}^{N} \delta(x - x_i) \cdot G_\sigma(x) \tag{3.2}$$

分割圖 $\boldsymbol{M}^{seg} \in \{0,1\}^{H\times W}$，我們運用相似的方法：

$$\boldsymbol{M}^{seg}(x) = \sum_{i=1}^{N} \delta(x - x_i) \cdot J_n(x) \tag{3.3}$$

其中$J_n(x)$ 是大小為 $n \times n$ 在位置為 x 中心的矩陣，數值皆為 1 。因此，矩陣 M^{seg} 中的 1 和 0 分別表示注意區和背景區的像素。我們在所有實驗中憑經驗設置 $n = 25$，以確保圖像中特定頭部的特徵在於分割圖中的像素多於密度圖中的像素，以避免缺失其他有用的數據。

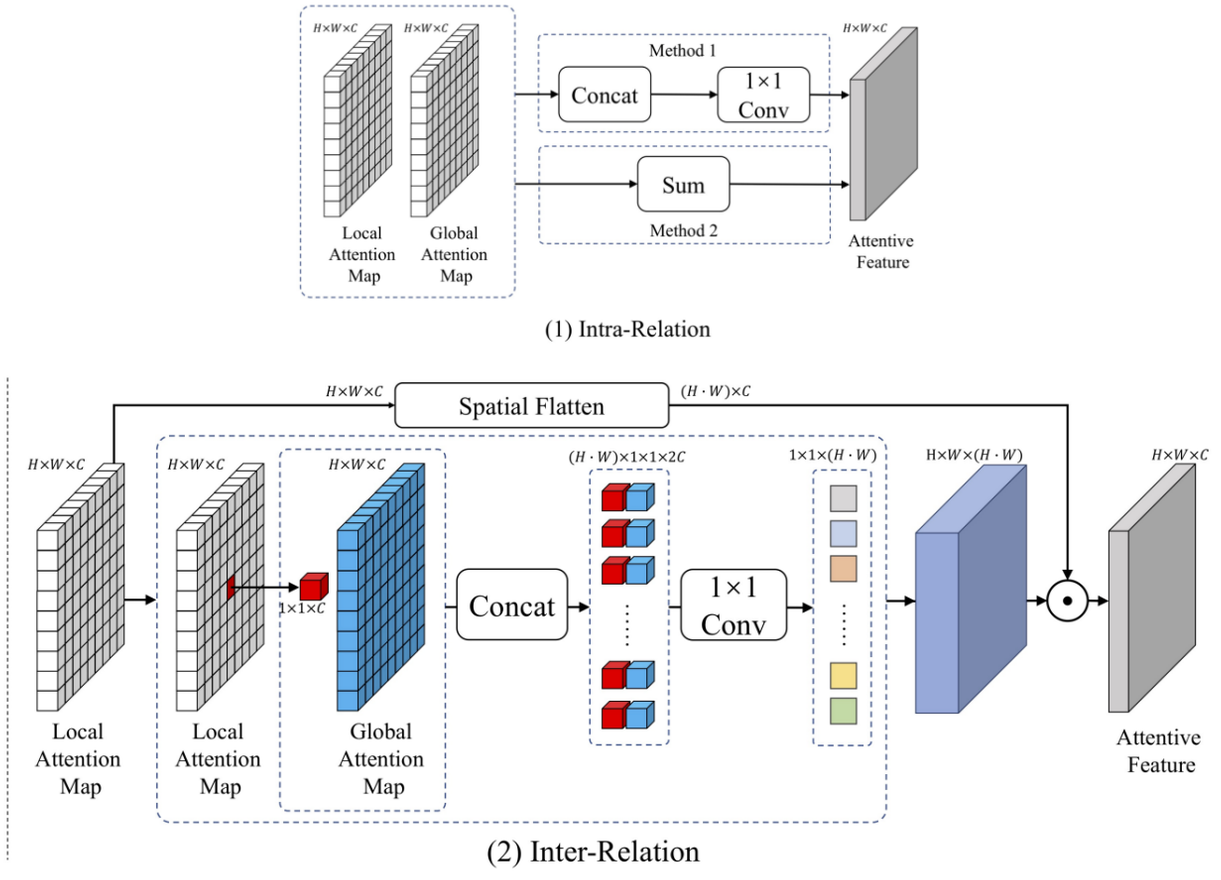


圖 38，神經網路超參數配置

在此研究中，我們不是從頭開始設計新的網絡，而是利用最經典的 CNN 模型進行圖像分類 U-Net。為了在人群計數中應用原始的 U-Net 網絡，已經進行了一些有利的修改。首先，我們移除最後的全連接層並保留所有卷積層。原始 U-Net 網絡的輸入大小為 299×299，最終卷積層的輸出大小為 8×8。也就是說，最後一個卷積層生成的特徵圖具有輸入圖像的大約 1 個空間分辨率。這 25 個是通過第一個卷積層（步幅為 2）、兩個最大池化層（步幅為 2）和兩個使用最大池化（步幅為 2）的 U-Net模塊實現的。為了確保在人群計數中很重要的輸出密度圖的空間分辨率，我們從原始網絡中刪除了前兩個最大池化層，並在之前添加了一個上採樣層最終的 U-Net 模塊。結果，當輸入大小為 2n（例如，在我們的例子中為 128 × 128）時，修改網絡的輸出恰好具有輸入4 圖像的 1 空間分辨率。這種修改不會改變網絡參數的數量，因此可以直接加載和使用預訓練的權重，由於中間特徵圖的空間分辨率提高了，操作次數也增加，這個修改後的模型也將表示為 U-Net，並在我們的實驗中用作基準方法。

在多任務學習框架中使用分割圖提取更顯著特徵以進行密度圖估計與現有的方法不同。分割圖可以用作為理想的注意力圖，以強調內部特徵的貢獻，另外將前景區域轉換為密度估計圖，同時壓縮背景區域內的特徵效果。為此，我們添加了一個注意力網路層來估計注意力圖。注意力網路層是卷積層，後面是一個 sigmoid 層，它將輸出值限制在 0-1 的範圍內。 注意層將倒數第二個 U-Net 模塊生成的特徵圖作為輸入，並輸出與輸入具有相同空間分辨率的單通道注意力圖。 隨後，通過與特徵圖的每個通道的元素級乘積，將注意力層估計的注意力圖應用於最後一個 Inception 模塊生成的特徵圖。

$$F^l = F^{l-1} \odot M_{att} \tag{3.4}$$

在此模型中注意力網路層的設計與[51, 52]中的類似。然而，在 [51]中估計一個所謂的逆注意力圖，而我們的注意力層生成一個直接應用於特徵圖的注意力圖。此外，[51]中的地面實況分割圖中的前景區域是通過對密度圖進行閾值處理得到，因此兩個圖的每個頭部都有相同的正數，而本模型的則不同（參見方程式 (3.1)）。在 [52]中，注意力層將特徵圖作為輸入來估計注意力圖，該注意力圖再次應用於相同的特徵圖。這可能會限制模型的容量，因為它被迫通過兩個參數有限的捲積層從同一個特徵圖中學習兩個不同的圖。相比之下，如上所述，我們的注意力層的輸入是來自前一層的特徵圖，它具有更高的空間分辨率，並且不同於生成的注意力圖將應用於的特徵圖。這些有利的區別共同有益於密度圖的估計，並將在我們的實驗中進行經驗評估。

3.4 錯誤評估指標

為了能和過往的文獻做比較，也能和未來的研究有參考的依據，人流計數系統必須使用有效且眾所皆知的標準來衡量。選擇合適的評估指標取決於多種因素，目前包含：執行時間、內存容量、使用場域、準確性...等。本論文提供六個錯誤評估指標：特異度(Specificity)、敏感度(Sensitivity)、精密度(Precision)、骰子相似系數(Dice Jaccard similarity coefficient)、亞卡爾相似系數(Jaccard similarity)、準確率(Accuracy, ACC)，可用於將本論文預測之影像和 Ground Truth 實行數值化分析，這些數值指標月趨近於 1 表示越精準，TP、FP、FN、TN，在表 5 中定義：

表 5，混淆矩陣 (Confusion Matrix)

		模型預測	
		Positive	Negative
真實情況	Positive	TP	FN
	Negative	FP	TN

特異度 (Specificity)，亦稱做真陰性率 (True negative, TNR)，代表在所有真實情況為發人形的區域裡，模型正確判斷為非人形的比例。

$$\text{Specificity} = \frac{TN}{FP+TN} \tag{3.5}$$

靈敏度和特異度處於平衡狀態，若一項實驗靈敏度很高，其代價就是低特異度(假陽性的案例較多)；同樣，若一項實驗特異度高，靈敏度就會較低(假陰性的案例較多)。

靈敏度 (Sensitivity)，亦稱作真陽性率(True positive rate, TPR)，和 Recall 的定義也相同，代表在所有真實情況為人形的區域裡，模型亦正確判斷為人形的比例。

$$\text{Sensitivity} = \frac{TP}{TP+FN} \tag{3.6}$$

精密度(Precision)，亦稱為陽性預測值(Positive predictive value, PPV)，如式 3.7 所定義，代表模型預測為腫瘤中真實情況亦為腫瘤的比例，因為對過度切割十分敏感，所以被認為是很有用的評估指標。

$$\text{Percision} = \frac{TP}{TP+FP} \tag{3.7}$$

骰子相似係數 (Dice Jaccard similarity coefficient)，如式 3.8 所定義，亦被稱為 F1-score，A 表示切割的面積，從式 3.9, 3.10 看出 DSC 與 JSI 呈正相關。

$$\text{DICE} = \frac{2|A_{Ground\ truth} \cap A_{predict}|}{|A_{Ground\ truth}|+|A_{predict}|} = \frac{2*TP}{2*TP+FP+FN} \tag{3.8}$$

$$\text{JSI} = \frac{DSC}{2-DSC} \tag{3.9}$$

$$\text{DSC} = \frac{2JSI}{1+JSI} \tag{3.10}$$

話雖如此，兩者還是有細微區別，由式 3.11 可看到，JSI會傾向對單一實例判斷錯誤的結果進行更多的懲罰，判斷錯誤的情況會有類似「平方」的效應，因此 DSC通常更傾向是評估平均的性能，而 JSI則更傾向是評估最壞情況的性能。

雅卡爾相似指數 (Jaccard similarity)，亦稱為 Intersection of Union (IoU)，如式 3.11 所定義，A 表示切割的面積，分母為真實切割面積和預測切割面積的聯集，分子為真實切割面積和預測切割面積的交集，這種評估指標經常被使用於物件偵測(Object detection)的任務中。

$$\text{JSI} = \frac{A_{Ground\ truth} \cap A_{predict}}{A_{Ground\ truth} \cup A_{predict}} = \frac{TP}{TP+FP+FN} \tag{3.11}$$

在機器學習領域中，通常習慣使用預測類別標籤來判定模型輸出正確與否的單一標籤，然而對於物件偵測任務，並非如此簡單，因為預測邊界框的(x, y)座標幾乎不可能與真實邊界框的(x, y)座標完全吻合，因此設定 JSI 指標獎勵與真實邊界框重疊較多的預測邊界框。

準確率 (Accuracy, ACC)，如式 3.12 所定義，分母為所有情形之加總，分子為判斷正確的情況，表示為正確分類的影像像素之百分比，是最基本的評估指標，但在類別不平衡的情況下，主導類別的較高準確率將掩蓋其他類別的較低準確率，出現偏差的結果。

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.12}$$

以本論文的資料集舉例，若人流區域極小，95%的影像都是背景，也被正確判斷為 TN，如此以來有 95%的像素被準確分類，但卻是毫無意義的預測，因此需要其他評估指標輔助我們如何正確評估模型的好壞。

ROC(Receiver Operating Characteristics)曲線圖可表示分類模型對所有分類閥值的效能，由真陽性率(TPR)和假陽性率(FPR)兩個參數所組成，其中 FPR為 $1 - \text{TNR}$，定義如式 3.13:

$$\text{FPR} = \frac{FP}{TN + FP} \tag{3.13}$$

以本論文提出之 U-Net 的 ROC 曲線圖為例，如圖 39，曲線愈趨近左上方表示效能愈好，愈貼近對角線表示效能愈差，AUC(Area Under The Curve)為 ROC 曲線下的面積，為評估分類模型預測能力中常用的統計值。

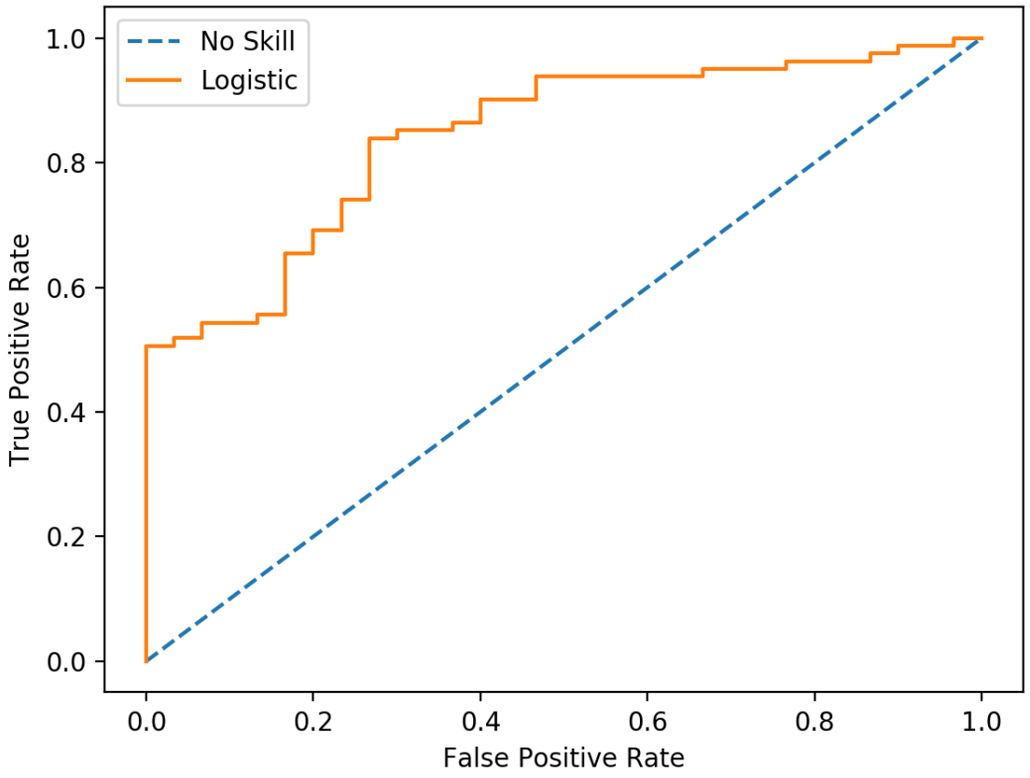


圖 39，ROC 曲線

PR 曲線圖(Presion-Recall Curve)由 Precision 和 Recall 兩個參數所組成，如圖 39，曲線愈趨近右上方、數值愈趨近 1 意味著具有愈良好的分類效能，當數值為 0.5 時則表示沒有任何分類能力，數值小於 0.5 表示分類效能比隨機猜測要差，直接反轉結果反而比較好。

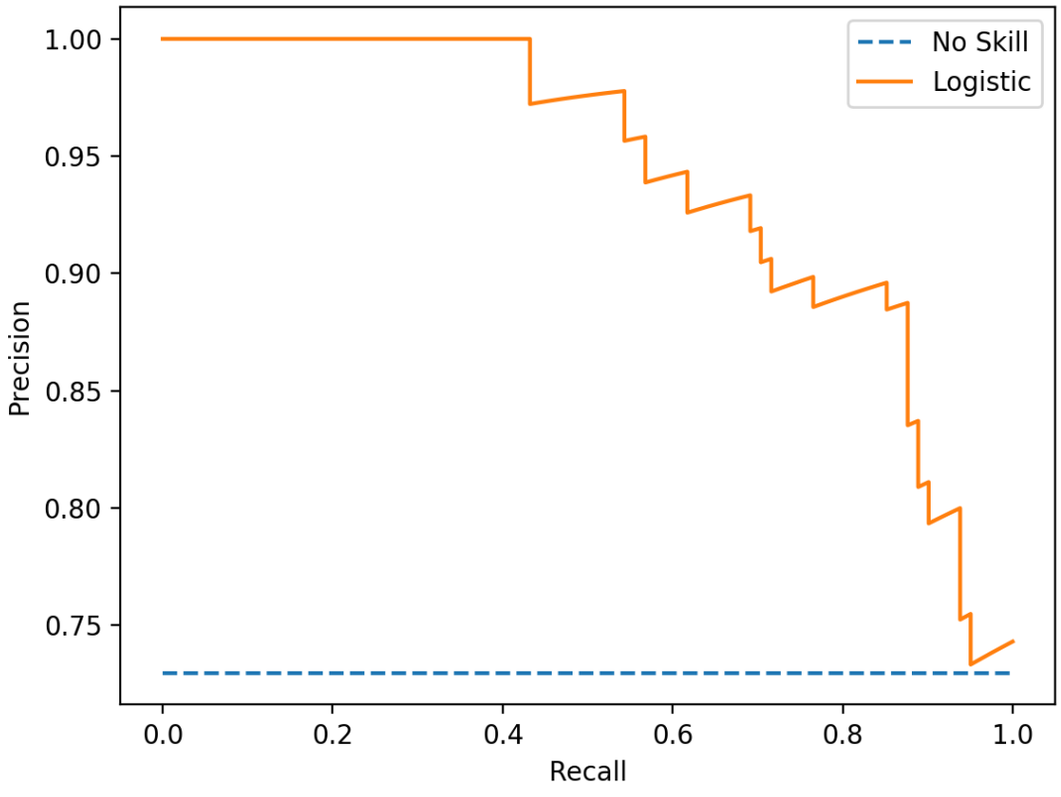


圖 40，PR 曲線

在沒有課程式學習中的損失函數情況下訓練本模型，並在下一節中描述課程式學習中的損失函數。損失函數由兩部分組成。第一個是應用於密度估計圖的 L2 損失，表示為$\mathcal{L}^{den}$。密度圖的損失可以計算如下：

$$\mathcal{L}^{den}(\boldsymbol{\Theta}) = \frac{1}{2N}\sum_{i=1}^{N}\left\|\widehat{\boldsymbol{M}}_i^{\,den} - \boldsymbol{M}_i^{den}\right\|_F^2 \tag{3.14}$$

損失函數的第二個組成成分是分割圖$\mathcal{L}^{seg}$，它被定義為交叉熵損失：

$$\mathcal{L}^{seg}(\boldsymbol{\Theta}) = -\frac{1}{N}\sum_{i=1}^{N}\left\|\begin{matrix}\boldsymbol{M}_i^{seg}\odot log\left(\widehat{\boldsymbol{M}}_i^{\,seg}\right) \\ +(1-M_i^{den})\odot log\left(1-\widehat{\boldsymbol{M}}_i^{\,seg}\right)\end{matrix}\right\|_1 \tag{3.15}$$

$\|\cdot\|_1$表示元素矩陣範數，即矩陣中所有元素的總和，⊙ 表示兩個相同大小的矩陣的元素相乘。 這兩個組件在網絡訓練期間組合在一起，組合損失函數為：

$$\mathcal{L}(\boldsymbol{\Theta}) = \mathcal{L}^{den}(\boldsymbol{\Theta}) + \lambda \mathcal{L}^{seg}(\boldsymbol{\Theta}) \tag{3.16}$$

其中 λ 是一個超參數，可確保兩個組件具有可比較的值，並在實驗中設置為 20。

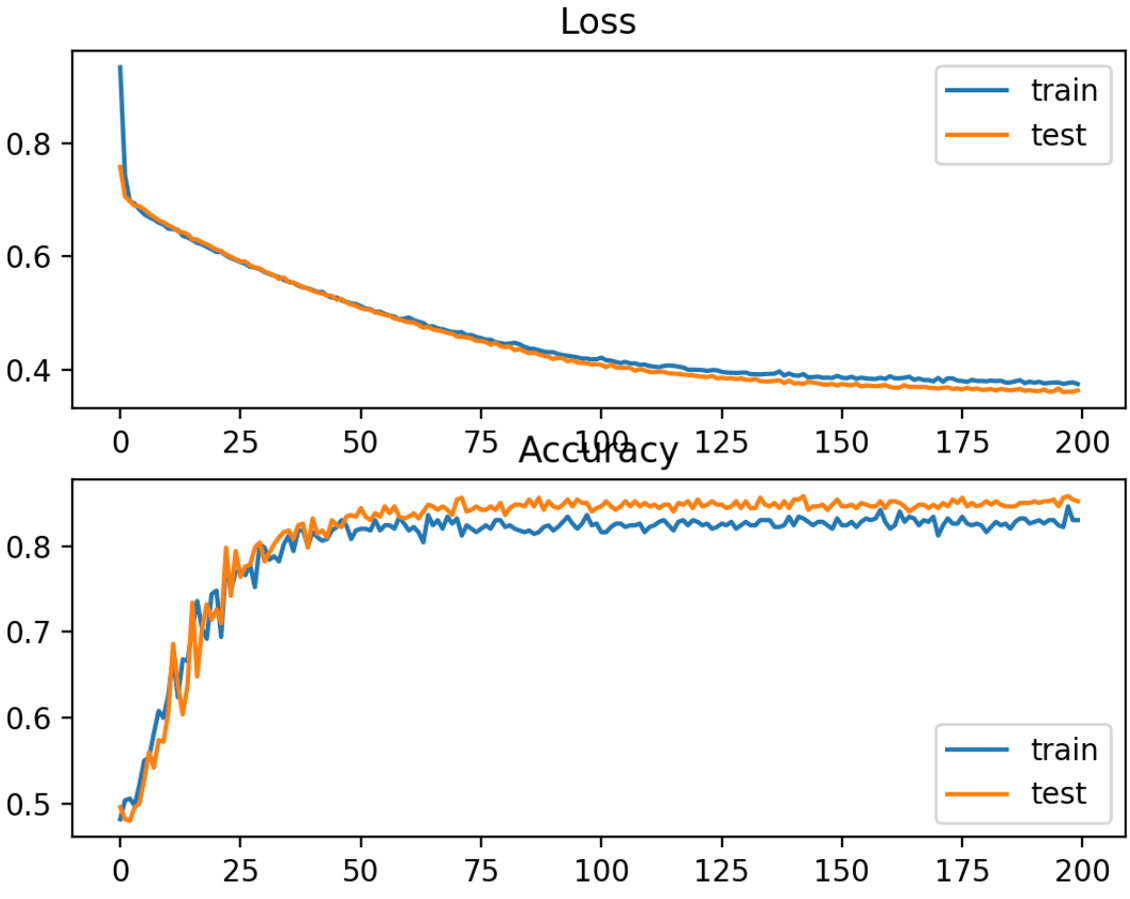


圖 41，損失函數曲線

從課程式學習的策略中受益，我們在本節中提出了一種新的課程式學習中的損失函數來代替等式中定義的傳統密度圖損失函數，如方程式 3.14 所示。課程式學習中的損失函數目的在計算密度圖損失時，了解像素級的難度級別。基於密集人群通常比稀疏人群更難計數的事實，我們設計一個課程，其中密度圖中高於動態閾值的像素被定義為困難像素。我們在計算密度圖損失時設置動態閾值並為密度圖的不同像素分配變量權重。具體來說，我們定義一個權重矩陣 $\boldsymbol{W}$ 如下：

$$\boldsymbol{W} = \frac{T(e)}{max\{\boldsymbol{M}^{den} - T(e), 0\} + T(e)} \tag{3.17}$$

權重矩陣 $\boldsymbol{W}$ 與方程式 3.14 中使用的密度圖矩陣 $\boldsymbol{M}^{den}$ 具有相同的大小，像素權重由動態閾值 $T(e)$ 和密度圖中的像素值確定。如果密度圖的像素值高於閾值，則將該像素視為困難像素，相應的權重設置為小於1，否則權重為1，像素值越高，權重越小。因此，訓練將更多地關注密度值低於 $T(e)$ 的像素。

動態閾值 $T(e)$ 被定義為訓練時期指數的函數，形式為：

$$T(e) = ke + b \tag{3.18}$$

其中 k 和 b 可以根據地面實況密度圖中像素值的人工標注確定。b 的值是初始閾值，它應該等於表徵單個頭部的區域中的最大密度值。 k 的值控制著增加難度的速度，當不使用課程學習策略時，這也可以很容易地從學習曲線中推導出來。

最後，密度圖中的程式學習中的損失函數可以通過將方程式 3.14 修改為：

$$\mathcal{L}^{den}(\boldsymbol{\Theta}) = \frac{1}{2N}\sum_{i=1}^{N}\left\|\boldsymbol{W}(e)\odot(\widehat{\boldsymbol{M}}_i^{\,den} - \boldsymbol{M}_i^{den})\right\|_F^2 \tag{3.19}$$

其中 $\boldsymbol{W}(e)$ 是一個關於訓練回合數 e 的函數。

3.5 資料處理與增強

交叉驗證(Cross Validation)是一種對樣本抽樣的機制，用於評估機器學習模型性能，並取得模型對獨立測試資料集的執行狀況。隨著資料分配之隨機狀態的變化，模型的精確度也會產生變化，因此我們無法讓模型保持固定的精確度。將所有資料分成用於模型開發的訓練資料、用於驗證同一模型性能的驗證資料以及最後用於評估模型性能的測試資料，如圖 42 所示:

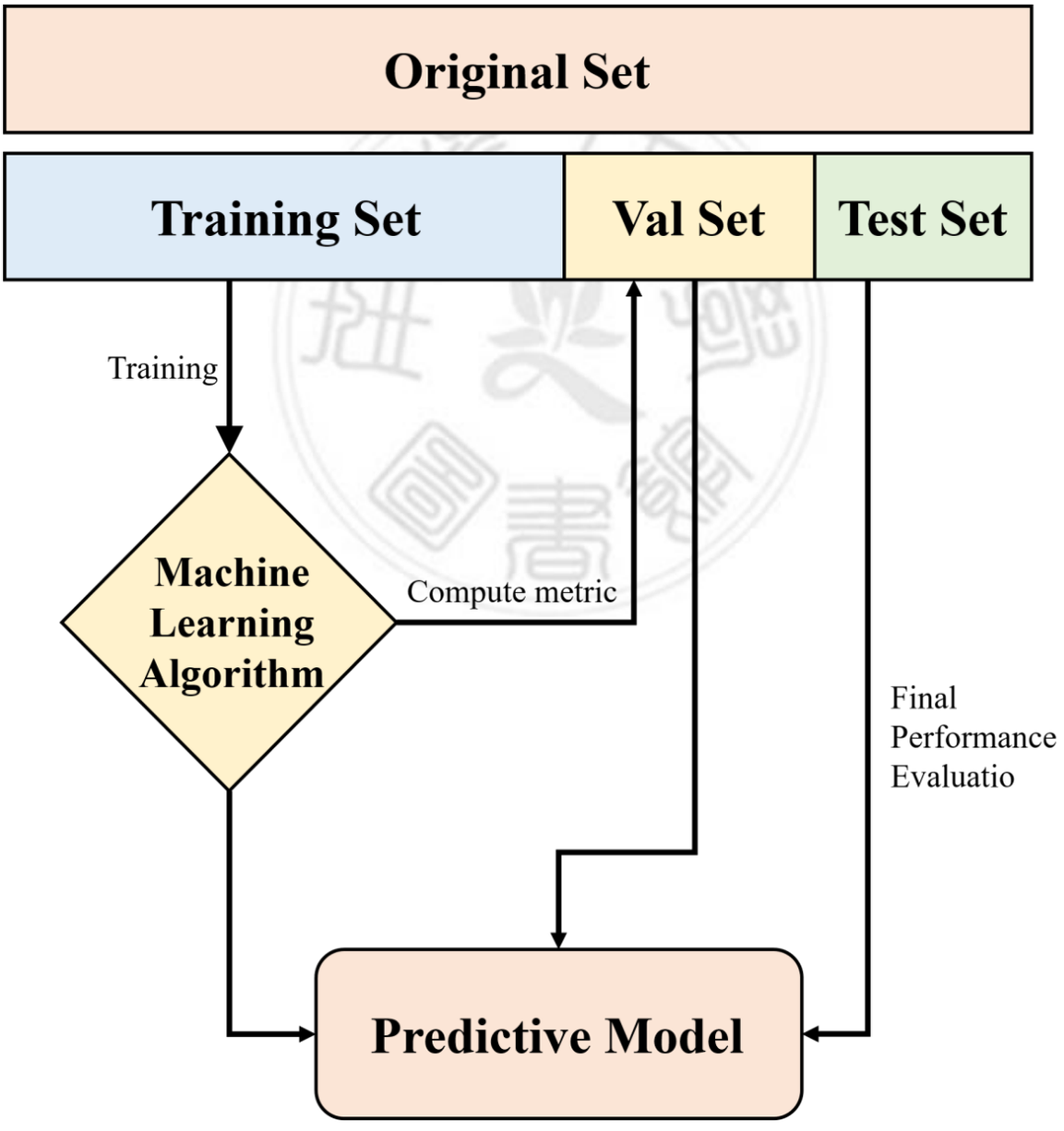


圖 42，Holdout CV 架構圖

本論文採用的是 Holdout CV 的方式，具體如圖 42，具體實施是從資料集中隨機取得 72%資料作為訓練資料以及 18%作為驗證資料，而剩餘的 10%測試資料是完全獨立的，並不會參與模型的訓練，在程式中的引數 random_state 設定之數字意義為該組隨機樣本的編號，注意若填 0 或預設不填，這每次的隨機樣本都會不同，這種機制可以幫助我們在需要重複讀取檔案時，取得一模一樣的隨機樣本，例如填上 3，之後若要使用同樣這組隨機樣本則應設為同樣為 3 的數字，進而實現 Holdout CV。優點為計算成本較低且易於實行，缺點則 43 是不適用於不平衡的資料集。

另外一種方式為 k-fold CV，將訓練資料分成 k 個子集，在每次迭代中，選擇其中一組作為驗證資料，其餘(k-1)組作為訓練資料，總共訓練模型 k 次，最後將所有評估指標在 k 次訓練中取平均值。優點為資料利用率高，因為整個資料集都有被訓練到，偏差(bias)也因此較低，缺點則為需要更高的計算成本，對內存需求更高，也因此不被本論文所採用。

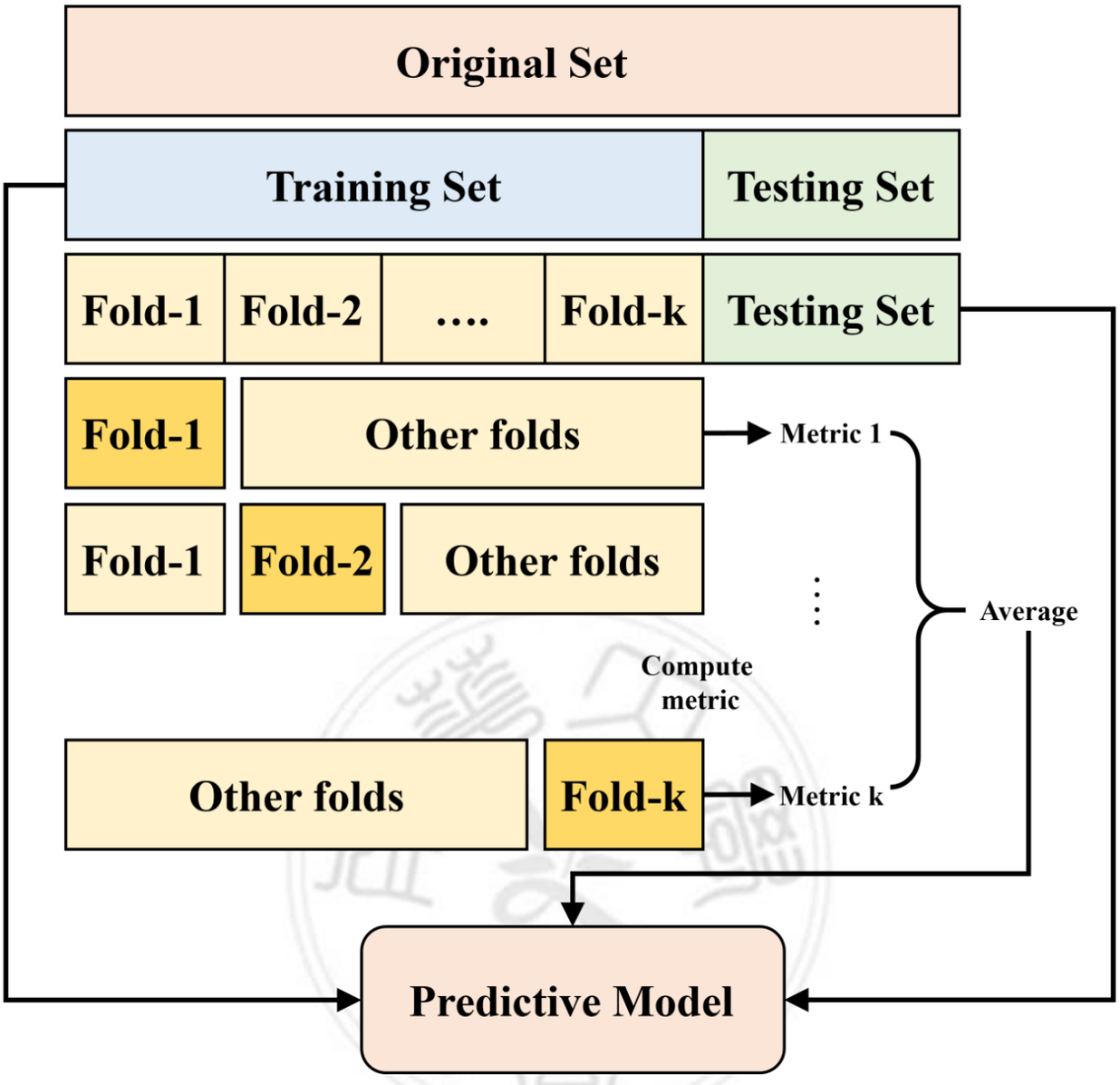


圖 43，k-fold CV 架構圖

使用如 Holdout CV 或是 k-fold CV (圖 43)來執行機器學習相關研究時，理論上不會單單執行一次即作為最後的結果，因為資料集在隨機取樣時可能發生不公平的情況，因此本論文透過調整不同的 random_state 引數，重複實驗 50 次，並將 50 次的評估指標數值(如：ACC、Jaccard、Dice Coefficient…等)取平均和標準差，作為最後模型的效能評估，標準差可以用以呈現模型的穩定性，假使數值過高可能表示其穩定性不佳。

本論文所使用資料總數僅有 2783 筆，由於人群影像不容易取得、標註成本過高，因此資料量通常偏少，導致無法訓練出具有強健性的模型並發生過擬合(Overfitting)。資料擴增(Data Augmentation)即為從現有資料創造出更多訓練資料的方法，可以透過如旋轉、平移、縮放、亮度調整、水平或垂直翻轉等操作來生成相似的影像，能大幅降低過擬合的可能性。

在 Pytorch 1.7 版本以上的深度學習庫中，藉由 torch.utils.data.Dataset 繼承函式建立自己的 Dataloader，如圖 44 所示，進而實現資料擴增的功能，其優點是不必事先生成一大批影像，造成電腦記憶體的負擔，而是在送入 batch 前透過隨機變換來達成目的。

```
class MyDataset_CarBrandsImages(torch.utils.data.Dataset):
    '''
    Class to load the dataset
    '''
    def __init__(self,transforms):

        with open('./dataset/kaggle/CarBrandsImages/carbrand.json') as jsonfile:
            data_load = json.load(jsonfile)
        self.imList = data_load['imagepaths']
        self.labelList = data_load['labels']
        self.transforms=transforms
        print('number of total data:{}'.format(len(self.imList)))
    def __len__(self):
        return len(self.imList)

    def __getitem__(self, idx):
        '''
        :param idx: Index of the image file
        :return: returns the image and corresponding label file.
        '''
        image_name = self.imList[idx]
        label = self.labelList[idx]

        # read image with PIL module
        image = Image.open(image_name, mode='r')
        image = image.convert('RGB')

        # Convert PIL label image to torch.Tensor
        image = self.transforms(image)
        label = torch.tensor(label)
        return image, label
```

圖 44，載入自有的 dataset 程式碼

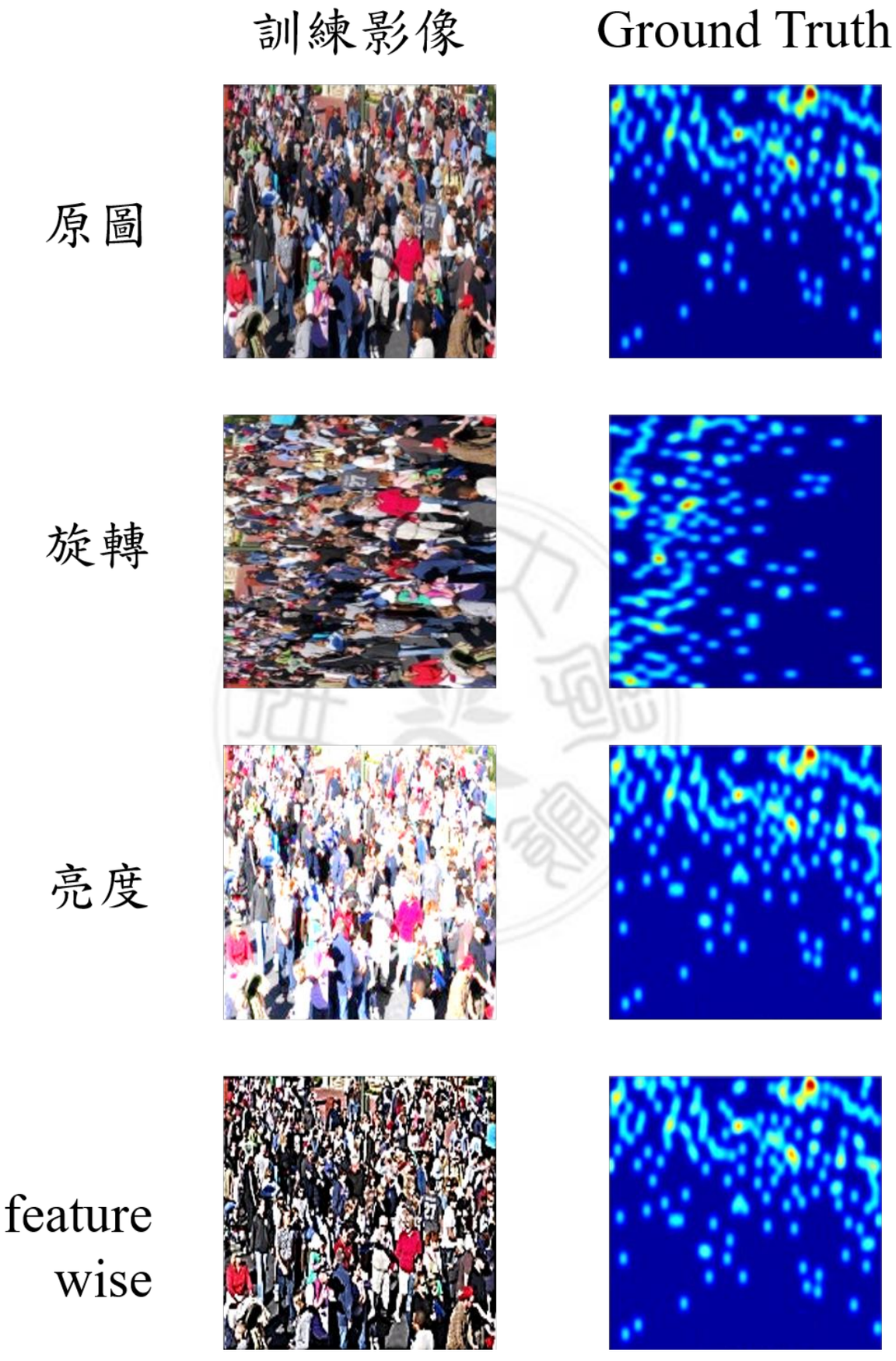


圖 45，資料擴增範例(旋轉、亮度、featurewise)

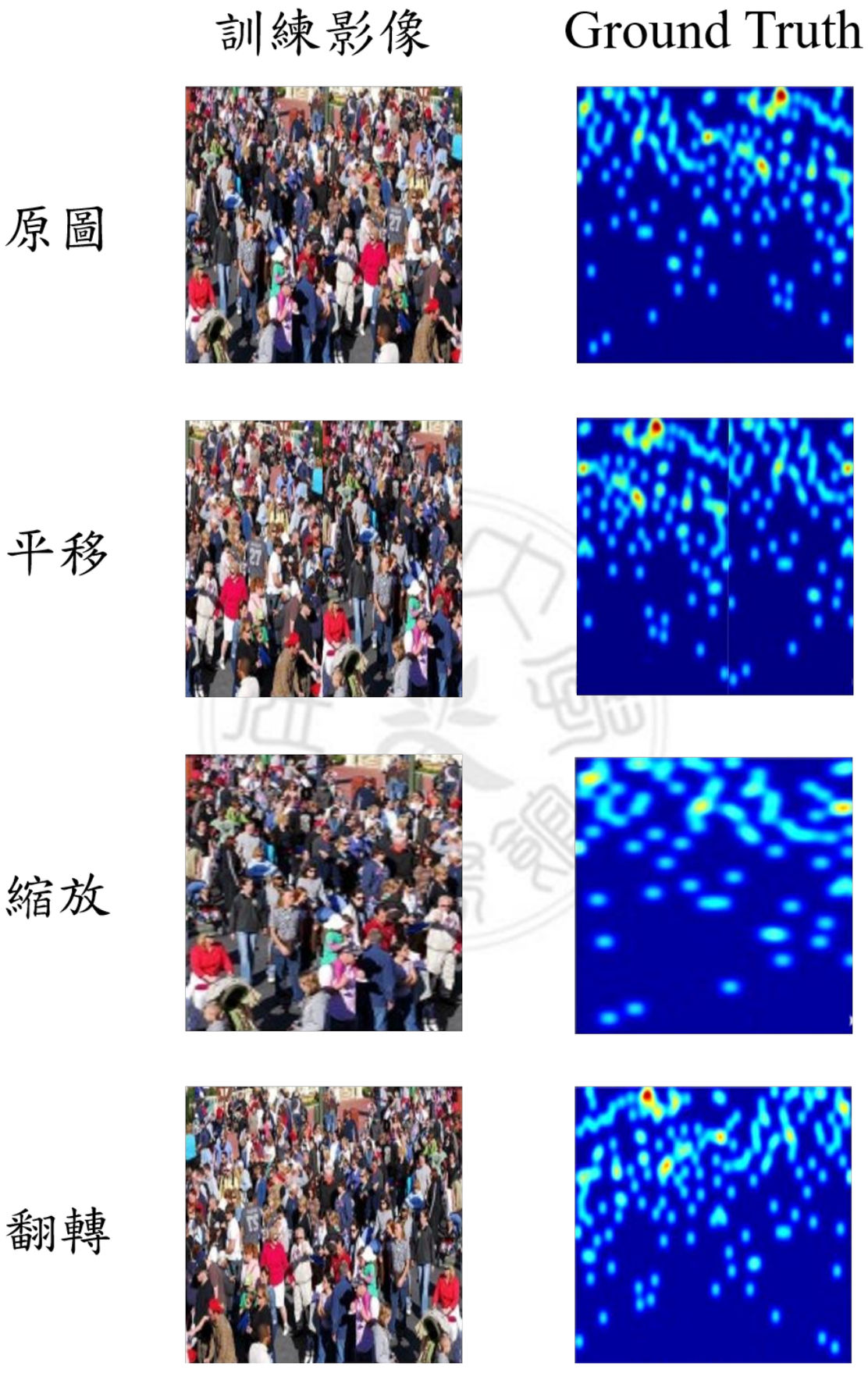


圖 46，資料擴增範例(平移、縮放、翻轉)

即便資料擴增方便我們擴大資料集，但必須了解原先資料集的特性，才能真正有效地利用資料擴增的優勢，例如圖 45 featurewise 是將影像 array 去中心化，將每個訓練樣本除以自身的標準差，但可以發現這會增加影像殘影的影響；圖 46 平移的程度不宜過大，若 fill_mode 填補不應該出現的值，則會發現影像不連貫的現象；圖 46 縮放的比例也應控制，若本身人形區域已經不小，再經過放大可能會如中間欄位的影像一樣超出範圍，而這些特性都是需要考量進實驗當中的。

3.6 實驗結果與討論

本論文所使用程式 Python 版本為 3.6.5，Pyotrch 版本為 1.7.0，透過 NVIDIA 推行的 CUDA 計算框架和 cuDNN 加速庫來訓練模型，版本依序為 9.0、7.6.5，作業系統為 Ubuntu 16.04，CPU 為 i7-7820X @ 3.6 GHz，電腦記憶體大小為 32GB，GPU 則是使用 NVIDIA GV 100 [TITAN V]。另外，使用 Matlab R2021a 協助繪圖與資料視覺化。

在表 6 呈現出各個模型方法的參數量大小和每 epoch 所需要的訓練時間，可以發現雖然使用 Dense block 可以減少參數量，但其中卻有一個隱性的缺點，就是較占用 GPU 的內存，訓練起來相較耗時。

表 6，各方法參數量和訓練時間表

Methods	#Parameters	Time per epoch (s)
MCNN	50,379,619	12
CSRNet	10,252,318	20
SANet	2,331,585	29
DADNet	2,037,697	4
CANNet	1,993,173	5
U-Net	1,941,105	3

在基準數據集上進行了大量實驗，以評估本模型的性能和人群計數中課程式學習中的損失函數的有效性。我們將簡要描述實驗中使用的數據集和評估指標、實驗協議和網絡訓練的細節。將實驗結果與最先進的方法進行比較和分析。

U-Net 最好的訓練結果之參數: batch_size = 8, steps_per_epoch = 41

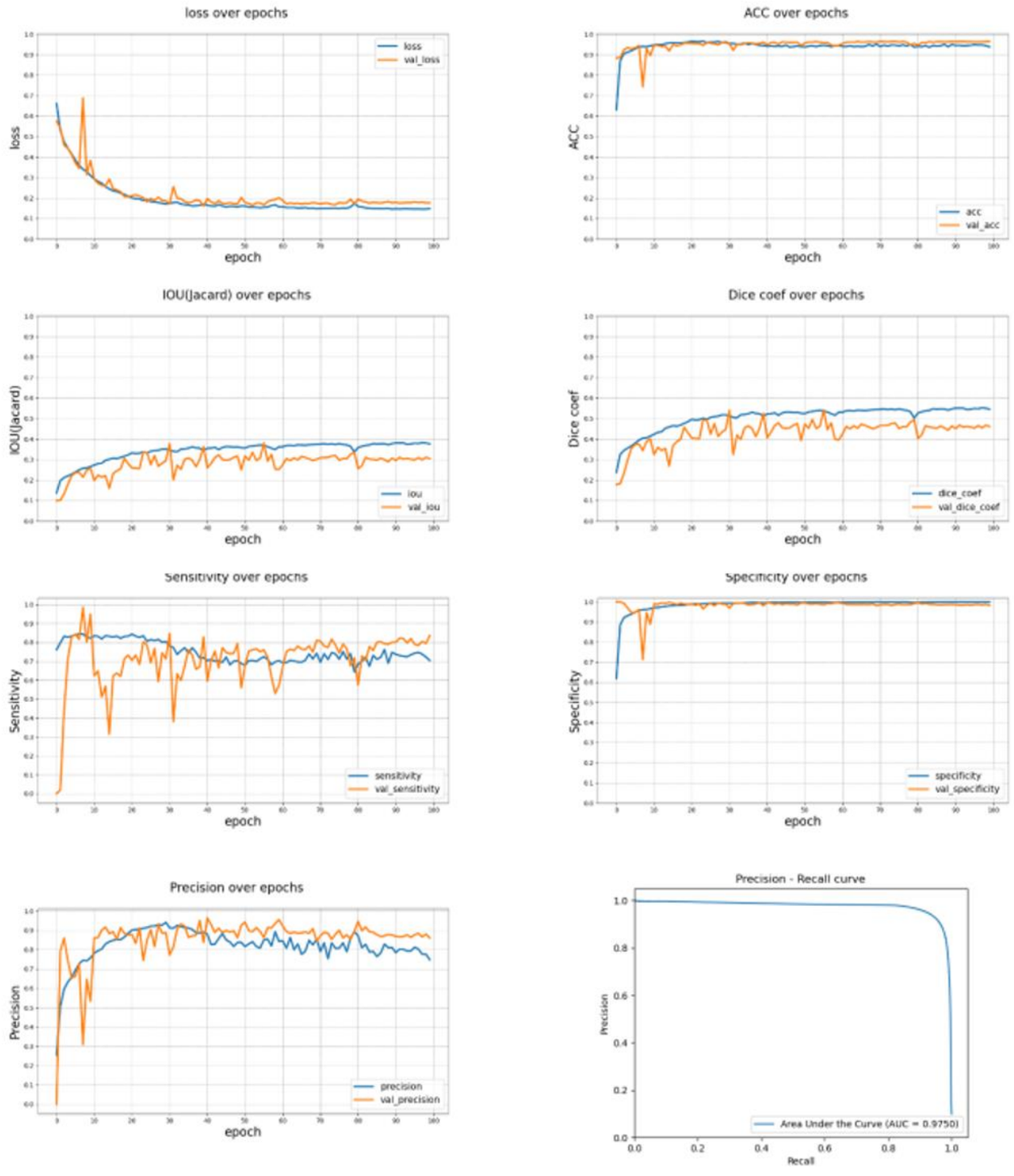


圖 47，U-Net 各項指標訓練圖

CANNet 最好的訓練結果之參數: batch_size = 12, steps_per_epoch = 27

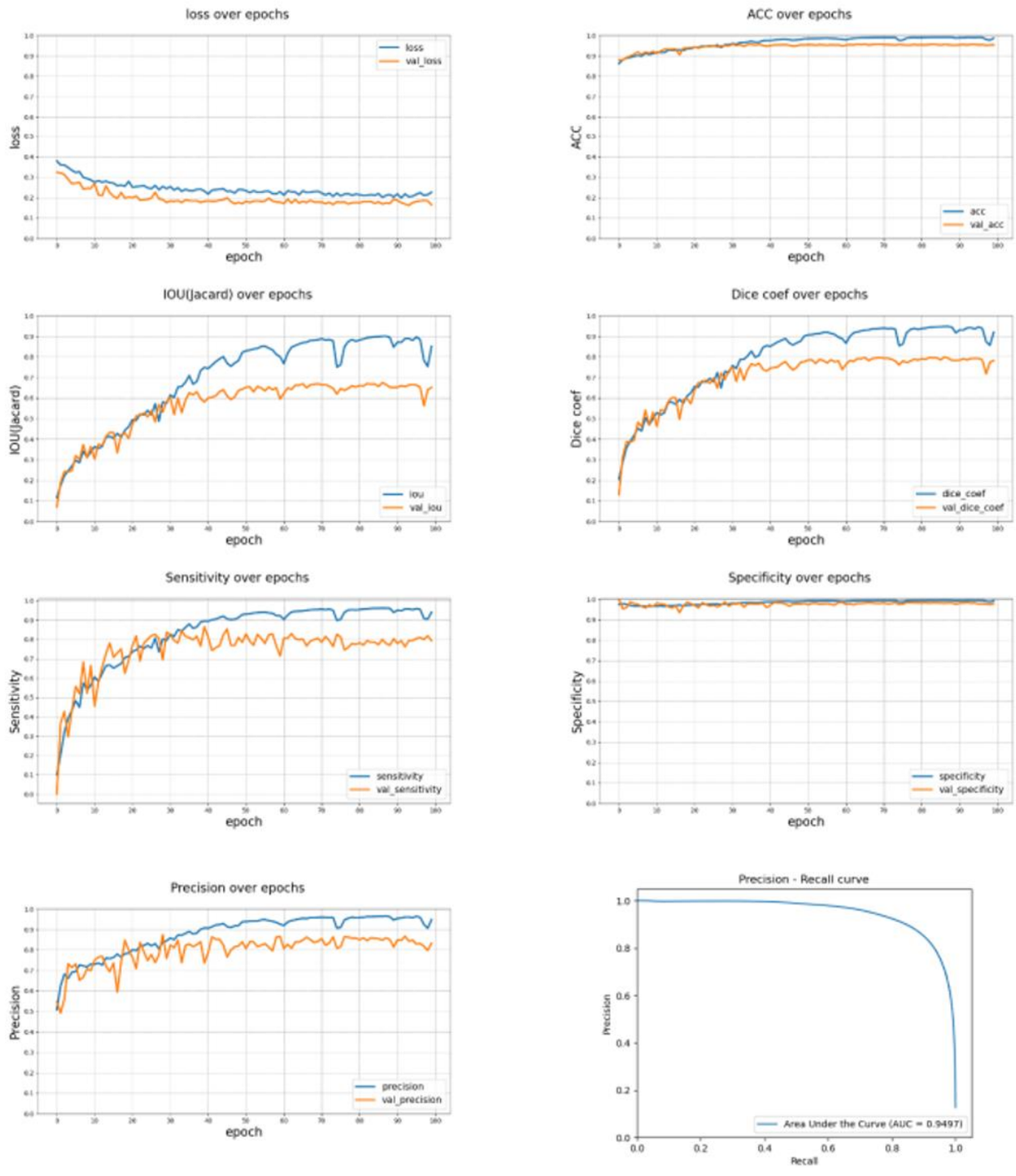


圖 48，CANNet 各項指標訓練圖

DADNet 最好的訓練結果之參數: batch_size = 12, steps_per_epoch = 27

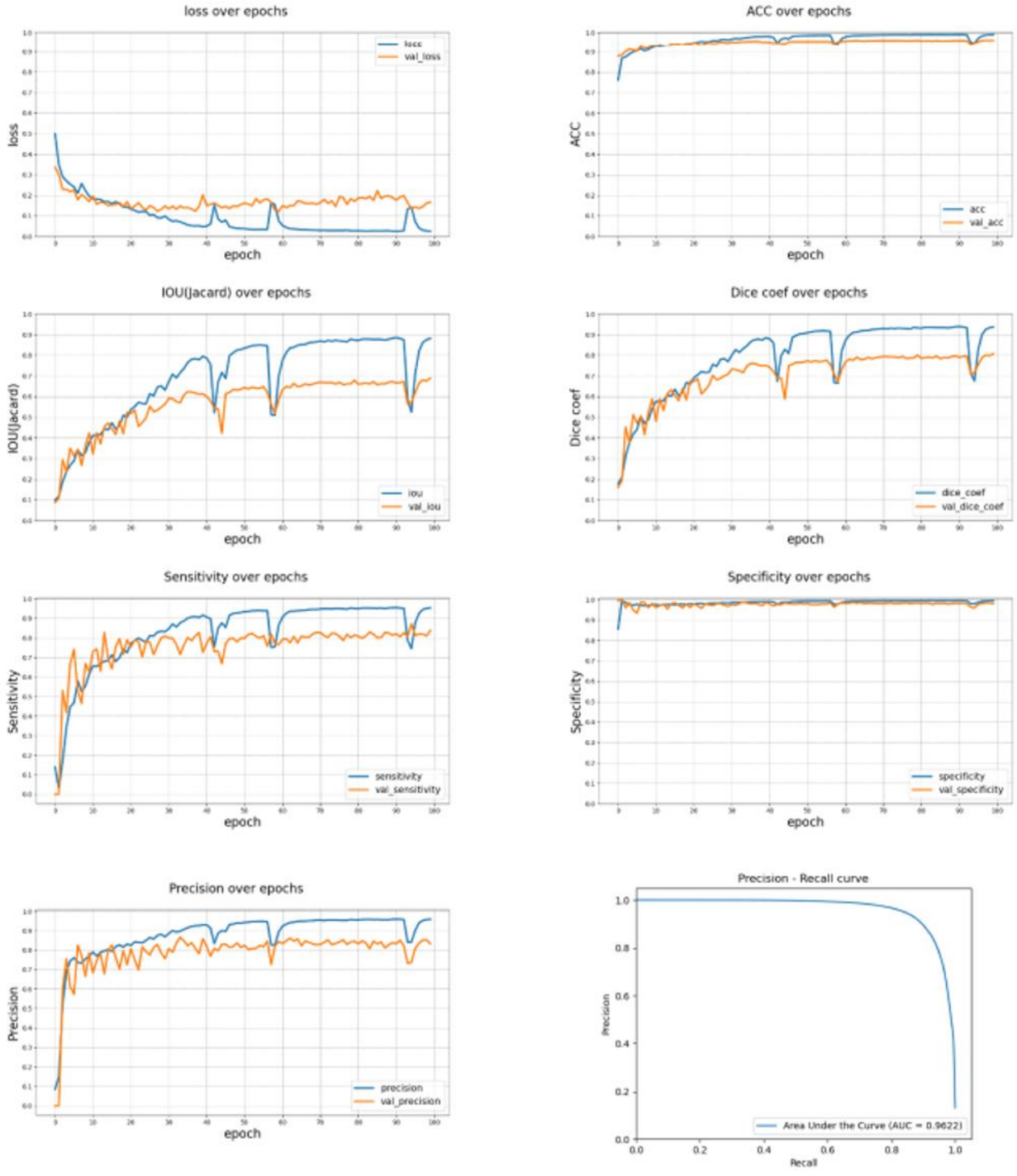


圖 49，DADNet 各項指標訓練圖

SANet 最好的訓練結果之參數: batch_size = 4, steps_per_epoch = 82

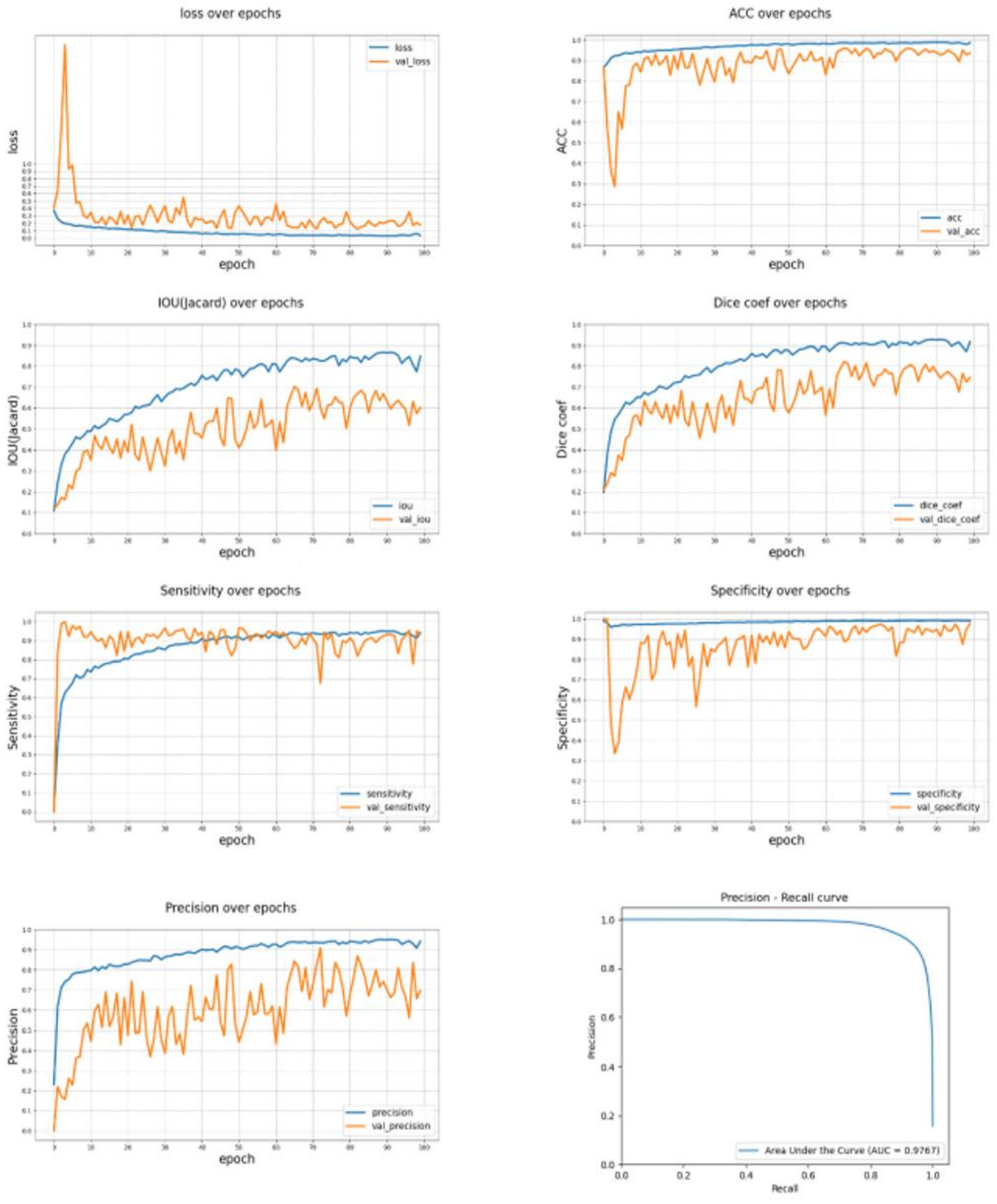


圖 50，SANet 各項指標訓練圖

CSRNet 最好的訓練結果之參數: batch_size = 8, steps_per_epoch = 41

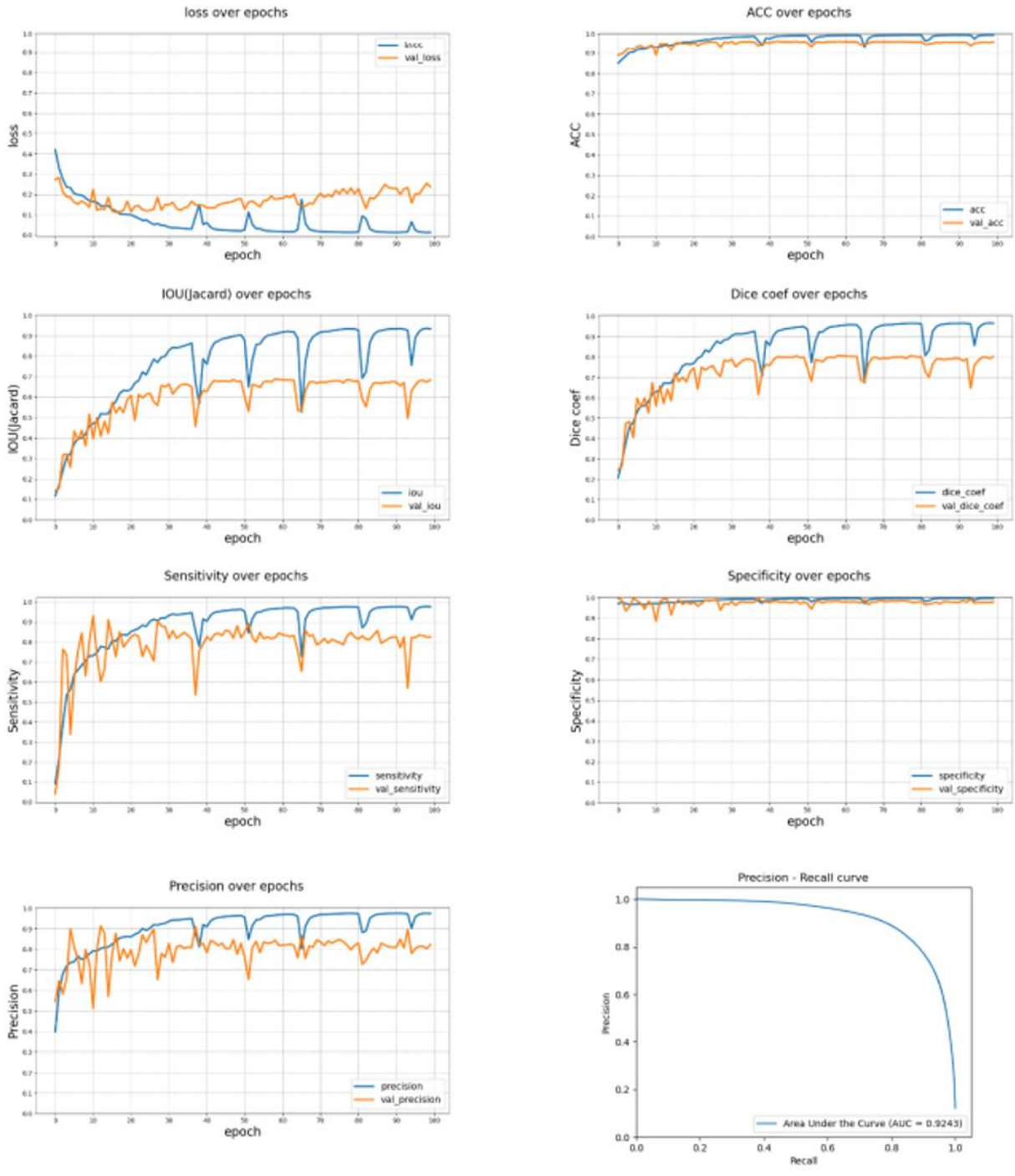


圖 51，CSRNet 各項指標訓練圖

MCNN 最好的訓練結果之參數: batch_size = 6, steps_per_epoch = 54

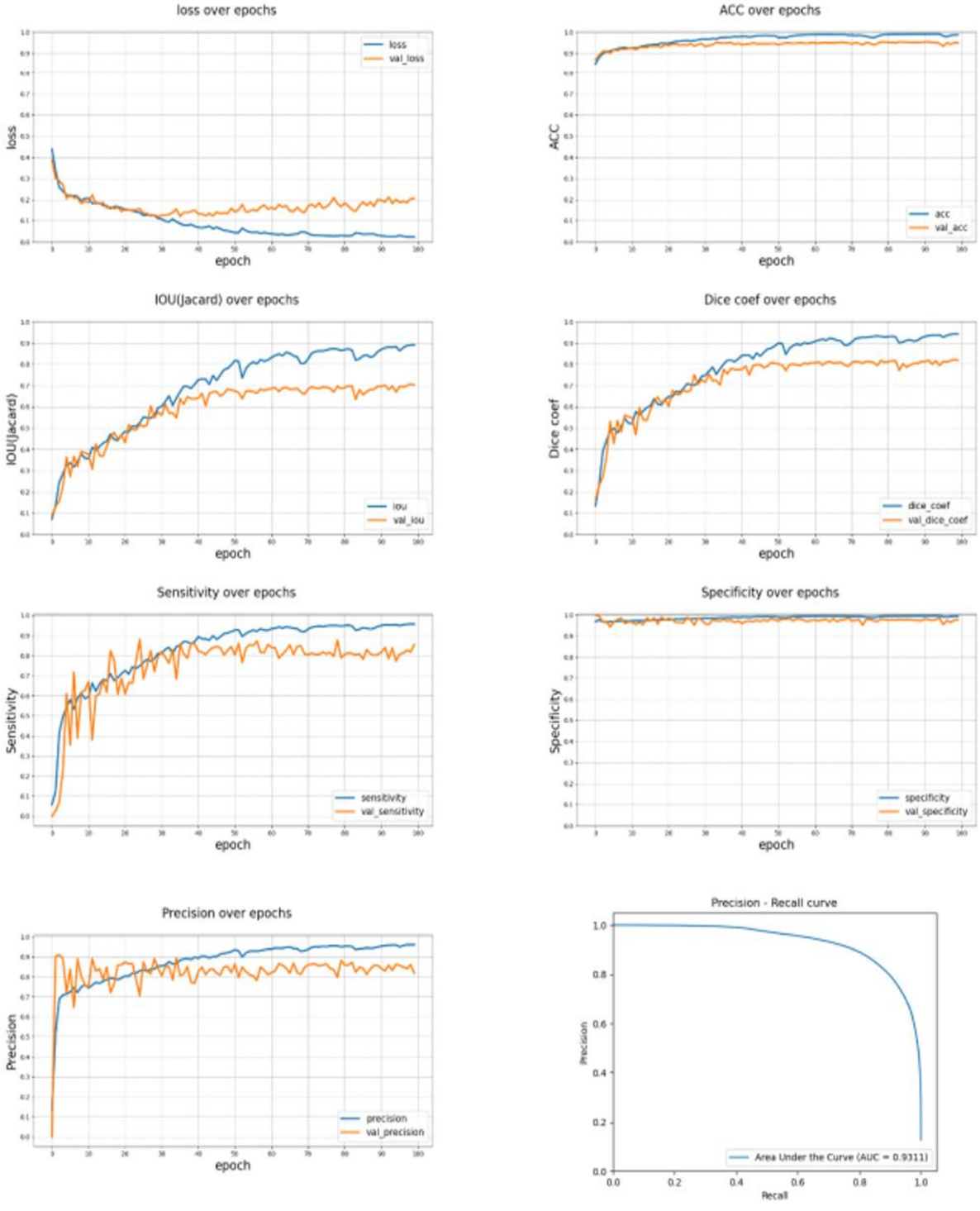


圖 52，MCNN 各項指標訓練圖

ShanghaiTech 數據集由[18]收集和發布，由兩部分組成。A部分分別由300張和182張不同分辨率的圖像組成，分別用於訓練和測試。 最小和最大計數分別為 33 和 3139，平均計數為 501.4。B 部分分別由 400 張和 316 張具有獨特分辨率 (768×1024) 的圖像組成，用於訓練和測試。 與 A 部分相比，這些圖像中的人數要少得多，最小和最大計數分別為 9 和 578，平均計數為 123.6。UCF_QNRF 數據集 [19] 包含 1,535 張高質量圖片，其中 1201 張用於訓練，334 張用於測試。 最小和最大計數分別為 49 和 12,865，平均計數為 815。UCF_CC_50 數據集 [35] 包含 50 張圖像，最小和最大計數分別為 94 和 4,534。由於圖像數量有限，這是一個具有挑戰性的數據集。遵循 [35] 和許多其他工作中的建議，我們在實驗中使用交叉驗證。

表7，與最先進的人群計數模型的比較結果（- 表示結果不可用）

Model	ShTechA		ShTechA		UCF-QNRF		UCF-CC-50	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MCNN	110.2	173.2	26.4	41.3	-	-	377.6	509.1
CSRNet	68.2	115.0	10.6	16.0	-	-	266.1	397.5
SANet	67.0	104.5	8.4	13.6	-	-	258.4	334.9
DADNet	64.2	99.9	8.8	13.5	113.2	189.4	285.5	389.7
CANNet	62.3	100.0	7.8	12.2	107	183	212.2	243.7
TEDNet	64.2	109.1	8.2	12.8	113	188	249.4	354.5
RANet	59.4	102.0	7.9	12.9	111	190	239.8	319.4
ANF	63.9	99.4	8.3	13.2	110	174	250.2	340.0
SPANet	59.4	92.5	6.5	9.9	-	-	232.6	311.7
Inception-v3	60.1	105.0	6.4	9.8	95.6	165.4	236.0	304.9
SGANet	58.0	100.4	6.3	10.6	89.1	150.6	224.6	314.6
SGANet+CL	57.6	101.1	6.6	10.2	87.6	152.5	221.9	289.8

我們選擇經典和最先進的模型進行比較，包括 MCNN，它是一個三列CNN，CSRNet，它使用 VGG16 作為前端，擴張的卷積層作為後端，SANet採用基本的 Inception 模塊但深度相對較淺，DADNet在框架中採用了擴張卷積、注意力圖和可變形卷積的思想，CANNet捕獲上下文感知特徵 通過多個分支，TEDNet [25] 也使用 Inception 風格的模塊，RANet [53] 使用迭代蒸餾算法，ANF使用條件隨機場 (CRF) 來聚合多尺度特徵，以及 SPANet。

實驗結果列在表 7 中，其中每列中的最佳結果以粗體突出顯示，第二好的結果以下劃線斜體突出顯示。從表 7 中，我們可以看到我們修改後的 U-Net 可以在所有四個數據集上實現非常有競爭力的性能。特別是在上海科技 B 部分，它實現了第二好的 MAE 6.4 和最佳 RMSE 9.8。在 UCF_QNRF 數據集上，U-Net 的結果也明顯優於大多數現有模型，包括 TEDNet（MAE：95.6 vs 113 和 MSE：165.4 vs 188），後者也使用了 U-Net 模塊。這些結果表明，異構 U-Net 模塊在分類問題中的優越性可以轉移到人群計數任務中，因此在設計用於人群計數的新型 CNN 架構以及其他受規模問題困擾的任務時，不同的 U-Net 模塊值得更多關注方差。另一方面，U-Net 在人群計數中的破壞性性能為研究社區在設計用於人群計數的新型網絡架構時選擇骨幹模型提供了更多見解。

從表7中，我們可以通過比較 U-Net 和 SGANet 之間的性能，看到分割引導注意層帶來的性能增強。為了驗證我們的分割引導注意層相對於其他類似設計的優越性，我們對 ShanghaiTech A 部分和 UCF_QNRF 進行了實驗。 在這個實驗中，我們遵循並通過將最後一個 Inception 模塊的特徵圖輸入註意力層並保持其餘部分不變來修改 SGANet。 實驗結果如表8所示，我們從中得出結論，我們的 SGANet 中使用分割圖的方式優於 [44] 中

的方式。

表8，不同的分割地圖監督方法的結果

Model	ShTechA		UCF_QNRF	
	MAE	RMSE	MAE	RMSE
W/o Seg. map	60.1	105.0	95.6	165.4
W/ Seg. map as	59.5	102.2	92.3	155.3
W/ Seg. map as SGANet	58.0	100.4	89.1	150.6

為了給出注意力層如何幫助密度圖估計的直觀證據，我們對來自 ShanghaiTech 部分 A 的五個示例測試圖像的估計注意力圖和密度圖進行了可視化。在圖34中，我們顯示了原始圖像、地面實況密度圖、分別在四列中預測密度圖和預測分割圖。真實計數和預測計數也顯示在密度圖上以進行直接比較。我們可以看到，鑑於準確預測的分割圖，前三個示例的預測誤差相對較低。但是，由於模型無法預測準確的前景區域，因此底部的兩個圖像具有更高的誤差。例如，第四行的圖像包含人在空中舉起手，並且手很容易被計數，因為它們的顏色與人臉相似。在底部圖像中，背景中的樹木被錯誤地識別為前景並導致高估計數。

第4章　結論與未來展望

4.1 結論

在智能城市資訊應用系統領域中，利用深度學習檢測人群計數的方法蓬勃發展，為了減少人力以及銜接智慧物連網的應用(AIoT)的發展，使用端對端的神經網路，用輸入影像和其 Ground truth 即可產生預測人形的影像，讓人流計算的任務不像從前那般嚴峻。

由於實驗室取得人群影像不易、資料量偏少，精確標記成本太過於昂貴，加上人力下降向智慧物連網發展的趨勢，深度學習的模型也應與時俱進，因此本論文在初期階段從在語意切割領域中廣為人知的 MCNN，轉而深入研究標榜能以更少量資料即可得到高切割準確率的 U-Net，模型參數量相較 MCNN 小二十倍。在解析 U-Net 和提出了一個使用 U-Net 作為主軸的分割引導注意網絡的過程中，注意到在這簡單卻有效的架構中，其實有許多部份還尚未被利用地淋漓盡致。有鑑於此，開始廣泛閱讀將現行深度學習模型結合 U-Net 發表的論文，發現這些精妙的模型架構大部分是使用於較普遍的公開資料集，尤以 MNIST、CIFAR 影像為大宗，因此本論文嘗試將用於其他影像的模型遷移到人群影像上，也提供給未來相關研究者一個有根據的參考。

深入理解多種模型的演進與變革後，開始探討其箇中精妙並予以整併，例如在收縮路徑與傳播路徑上存在語意程度上的差距，因此參考使用 SGANet 的引導注意層；而後發現像素級難度級別來解決人群圖像中的尺度差異問題，轉而使用分割引導注意網絡來取代，增加些許的訓練時間來換取正確率的上升，且在特徵學習部分得以提高重複利用率；最後透過課

程式學習，對於影像分類難度循序漸進的學習，可以克服模型難以收斂，並將其優化解碼器。

在交叉驗證與結果呈現的部分，本論文嘗試過 Holdout 和 k-fold 兩種交叉驗證的方式，礙於本論文使用之 GPU 硬體設備沒有足夠的 VRAM，使用 k-fold 時導致訓練時間倍增，且在中途程式經常被終止並出現資源耗盡的錯誤，因此最後決定採用 Holdout 的方式並根據 train_test_split 函式調整不同的訓練、測試資料比例，將每種方法實驗 50 次後取平均和標準差得到結果。

在錯誤評估指標的部分，在參閱多篇語意切割領域中較偏向智慧城市的應用後，決定新增 JSI、DSC 指標，因為此兩項指標不僅是對逐一像素評估，且更著重於真實人形與預測人形的交集，不會發生因為資料本身的特性導致評估效果失真的情況，例如人群範圍極小，導致 Accuracy 容易很高之現象。本論文提出之 U-Net 平均具有 78.10%的 JSI、85.79%的 DSC、97.81%的 ACC 以及 89.72%的 Precision 皆為所有方法最高，88.31%的 TPR 落後於 89.47%的 SGANet、98.73%的 TNR 落後於 98.79%的 TEDNet，總共有四項指標上平均數值較高，其餘兩項指標為第二名，且標準差也較小，反應出 U-Net 的穩定性，在醫學領域中尤為重要的一項特質，同時亦證實該方法可用於人形影像應用中，基於現存的方法做出改善。

綜上所述，本論文提出之方法具備以下優點: 其一，訓練過後的模型不需人工調適參數，運用全自動人群計數系統來填補重要資訊，可以節省寶貴的人力與時間；其二，使用較少的參數量、稀少的訓練資料也能有優良的像素級別切割能力；其三，在人群極小、殘影陰影嚴重等難易度較高的人群影像，依然能保持穩定的水準。

4.2 未來展望

在本文中，我們解決了一個對智能城市資訊應用系統具有重要價值的人群計數的重要問題。我們研究了 U-Net 在人群計數中的有效性，並提出了一個使用 U-Net 作為主軸的分割引導注意網絡。我們還通過定義像素級難度級別來解決人群圖像中的尺度差異問題，提出了一種新的人群計數課程損失函數。在四個常用數據集上的實驗結果表明，由於 U-Net 和分割引導注意層的結合，所提出的 SGANet 可以實現卓越的性能。 所提出的課程學習策略也被證明有助於各種現有的人群計數模型。

大多數現有的人群計數方法包括我們在本文中的方法都依賴於足夠的訓練數據，這需要大量的數據收集和註釋。在現實世界的應用中，獲得足夠的各種場景（例如，不同的相機分辨率、照明條件、天氣條件和視角）的訓練數據是具有挑戰性的。為了解決這個現實問題，我們未來的工作將集中在弱監督學習上，例如領域適應 [59] 和轉移學習 [28]。

第5章 參考文獻

[1] Q. Zhou, J. Zhang, L. Che, H. Shan, and J. Z. Wang, "Crowd Counting With Limited Labeling Through Submodular Frame Selection," *IEEE Transactions on Intelligent Transportation Systems,* vol. 20, no. 5, pp. 1728-1738, 2019, doi: 10.1109/TITS.2018.2829987.

[2] H. Y. S, G. Shivakumar, and H. S. Mohana, "Crowd Behavior Analysis: A Survey," in *2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, 16-17 March 2017 2017, pp. 169-178, doi: 10.1109/ICRAECT.2017.66.

[3] D. Ryan, S. Denman, S. Sridharan, and C. Fookes, "An Evaluation of Crowd Counting Methods, Features and Regression Models," *Computer Vision and Image Understanding,* vol. 130, 08/01 2014, doi: 10.1016/j.cviu.2014.07.008.

[4] V. A. Sindagi and V. M. Patel, "A survey of recent advances in CNN-based single image crowd counting and density estimation," *Pattern Recognition Letters,* vol. 107, pp. 3-16, 2018/05/01/ 2018, doi: https://doi.org/10.1016/j.patrec.2017.07.007.

[5] X. Ding, F. He, Z. Lin, Y. Wang, H. Guo, and Y. Huang, "Crowd Density Estimation Using Fusion of Multi-Layer Features," *IEEE Transactions on Intelligent Transportation Systems,* vol. 22, no. 8, pp. 4776-4787, 2021, doi: 10.1109/TITS.2020.2983475.

[6] W. Xie, J. A. Noble, and A. Zisserman, "Microscopy cell counting and detection with fully convolutional regression networks," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization,* vol. 6, no. 3, pp. 283-292, 2018/05/04 2018, doi: 10.1080/21681163.2016.1149104.

[7] M. Liang, X. Huang, C. Chen, X. Chen, and A. Tokuta, "Counting and Classification of Highway Vehicles by Regression Analysis," *IEEE Transactions on Intelligent Transportation Systems,* vol. 16, no. 5, pp. 2878-2888, 2015, doi: 10.1109/TITS.2015.2424917.

[8] T. Moranduzzo and F. Melgani, "Automatic Car Counting Method for Unmanned Aerial Vehicle Images," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 52, no. 3, pp. 1635-1647, 2014, doi: 10.1109/TGRS.2013.2253108.

[9] J. Praveen Kumar and S. Domnic, "Image based leaf segmentation and counting in rosette plants," *Information Processing in Agriculture,* vol. 6, no. 2, pp. 233-246, 2019/06/01/ 2019, doi: https://doi.org/10.1016/j.inpa.2018.09.005.

[10] B. Jiang, P. Wang, S. Zhuang, M. Li, Z. Li, and Z. Gong, "Leaf Counting with Multi-Scale Convolutional Neural Network Features and Fisher Vector Coding," *Symmetry,* vol. 11, no. 4, p. 516, 2019. [Online]. Available: https://www.mdpi.com/2073-8994/11/4/516.

[11] Z. Tao and R. Nevatia, "Bayesian human segmentation in crowded situations," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, 18-20 June 2003 2003, vol. 2, pp. II-459, doi: 10.1109/CVPR.2003.1211503.

[12] L. Dong, V. Parameswaran, V. Ramesh, and I. Zoghlami, "Fast Crowd Segmentation Using Shape Indexing," in *2007 IEEE 11th International Conference on Computer Vision*, 14-21 Oct. 2007 2007, pp. 1-8, doi: 10.1109/ICCV.2007.4409075.

[13] V. B. Subburaman, A. Descamps, and C. Carincotte, "Counting People in the Crowd Using a Generic Head Detector," in *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, 18-21 Sept. 2012 2012, pp. 470-475, doi: 10.1109/AVSS.2012.87.

[14] D. Kong, D. Gray, and T. Hai, "A Viewpoint Invariant Approach for Crowd Counting," in *18th International Conference on Pattern Recognition (ICPR'06)*, 20-24 Aug. 2006 2006, vol. 3, pp. 1187-1190, doi: 10.1109/ICPR.2006.197.

[15] P. Siva, M. J. Shafiee, M. Jamieson, and A. Wong, "Real-Time, Embedded Scene Invariant Crowd Counting Using Scale-Normalized Histogram of Moving Gradients

(HoMG)," in *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 26 June-1 July 2016 2016, pp. 885-892, doi: 10.1109/CVPRW.2016.115.

[16] V. Lempitsky and A. Zisserman, "Learning To count objects in images," presented at the Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 1, Vancouver, British Columbia, Canada, 2010.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM,* vol. 60, no. 6, pp. 84–90, 2017, doi: 10.1145/3065386.

[18] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-Image Crowd Counting via Multi-Column Convolutional Neural Network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 27-30 June 2016 2016, pp. 589-597, doi: 10.1109/CVPR.2016.70.

[19] H. Idrees *et al.*, "Composition Loss for Counting, Density Map Estimation and Localization in Dense Crowds," *ArXiv,* vol. abs/1808.01050, 2018.

[20] V. A. Sindagi and V. M. Patel, "Generating High-Quality Crowd Density Maps Using Contextual Pyramid CNNs," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 22-29 Oct. 2017 2017, pp. 1879-1888, doi: 10.1109/ICCV.2017.206.

[21] J. Liu, C. Gao, D. Meng, and A. Hauptmann, *DecideNet: Counting Varying Density Crowds Through Attention Guided Detection and Density Estimation*. 2018, pp. 5197-5206.

[22] Y. Zhang, C. Zhou, F. Chang, A. Kot, and W. Zhang, "Attention to Head Locations for Crowd Counting," 2019, pp. 727-737.

[23] D. Guo, K. Li, Z.-J. Zha, and M. Wang, "DADNet: Dilated-Attention-Deformable ConvNet for Crowd Counting," presented at the Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 2019. [Online]. Available: https://doi.org/10.1145/3343031.3350881.

[24] Z. Ma, X. Wei, X. Hong, and Y. Gong, "Bayesian Loss for Crowd Count Estimation With Point Supervision," *2019 IEEE/CVF International Conference on Computer Vision (ICCV),* pp. 6141-6150, 2019.

[25] Q. Wang, J. Gao, W. Lin, and Y. Yuan, "Learning From Synthetic Data for Crowd Counting in the Wild," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 8190-8199, 2019.

[26] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 2818-2826, 2016.

[27] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7-12 June 2015 2015, pp. 3431-3440, doi: 10.1109/CVPR.2015.7298965.

[28] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 21-26 July 2017 2017, pp. 2261-2269, doi: 10.1109/CVPR.2017.243.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 27-30 June 2016 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[30] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700-4708.

[31] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 11-19.

[32] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015: Springer, pp. 234-241.
[33] Z. Zhang, Q. Liu, and Y. Wang, "Road extraction by deep residual u-net," *IEEE Geoscience and Remote Sensing Letters,* vol. 15, no. 5, pp. 749-753, 2018.
[34] O. Oktay *et al.*, "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999,* 2018.
[35] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source Multi-scale Counting in Extremely Dense Crowd Images," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 23-28 June 2013 2013, pp. 2547-2554, doi: 10.1109/CVPR.2013.329.
[36] D. Sam, S. Surya, and R. Babu, *Switching Convolutional Neural Network for Crowd Counting*. 2017, pp. 4031-4039.
[37] Z.-Q. Cheng, J.-X. Li, Q. Dai, X. Wu, J.-Y. He, and A. Hauptmann, *Improving the Learning of Multi-column Convolutional Neural Network for Crowd Counting*. 2019.
[38] L. Liu, Z. Qiu, G. Li, S. Liu, W. Ouyang, and L. Lin, "Crowd counting with deep structured scale integration network," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1774-1783.
[39] V. Ranjan, H. Le, and M. Hoai, "Iterative crowd counting," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 270-285.
[40] V. A. Sindagi and V. M. Patel, "Ha-ccn: Hierarchical attention-based crowd counting network," *IEEE Transactions on Image Processing,* vol. 29, pp. 323-335, 2019.
[41] X. Jiang *et al.*, "Crowd counting and density estimation by trellis encoder-decoder networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6133-6142.
[42] D. B. Sam and R. V. Babu, "Top-down feedback for crowd counting convolutional neural network," in *Thirty-second AAAI conference on artificial intelligence*, 2018.
[43] V. A. Sindagi and V. M. Patel, "Multi-level bottom-top and top-bottom feature fusion for crowd counting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1002-1012.
[44] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1-9.
[45] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale Aggregation Network for Accurate and Efficient Crowd Counting," Cham, 2018: Springer International Publishing, in Computer Vision – ECCV 2018, pp. 757-773.
[46] N. Liu, Y. Long, C. Zou, Q. Niu, L. Pan, and H. Wu, "Adcrowdnet: An attention-injective deformable convolutional network for crowd understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3225-3234.
[47] Z. Cong, L. Hongsheng, X. Wang, and Y. Xiaokang, "Cross-scene crowd counting via deep convolutional neural networks," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7-12 June 2015 2015, pp. 833-841, doi: 10.1109/CVPR.2015.7298684.
[48] D. Oñoro-Rubio and R. J. López-Sastre, "Towards Perspective-Free Object Counting with Deep Learning," in *Computer Vision – ECCV 2016*, Cham, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., 2016// 2016: Springer International Publishing, pp. 615-629.
[49] M. Shi, Z. Yang, C. Xu, and Q. Chen, "Revisiting perspective information for efficient crowd counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7279-7288.
[50] M. Zhao, J. Zhang, C. Zhang, and W. Zhang, "Leveraging Heterogeneous Auxiliary Tasks to Assist Crowd Counting," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 15-20 June 2019 2019, pp. 12728-12737, doi: 10.1109/CVPR.2019.01302.

[51] V. A. Sindagi and V. M. Patel, "Inverse attention guided deep crowd counting network," in *2019 16th IEEE international conference on advanced video and signal based surveillance (AVSS)*, 2019: IEEE, pp. 1-8.

[52] Z. Shi, P. Mettes, and C. G. Snoek, "Counting with focus for free," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4200-4209.

[53] J. L. Elman, "Learning and development in neural networks: the importance of starting small," *Cognition,* vol. 48, no. 1, pp. 71-99, 1993/07/01/ 1993, doi: https://doi.org/10.1016/0010-0277(93)90058-4.

[54] P. Soviany, R. T. Ionescu, P. Rota, and N. Sebe, "Curriculum learning: A survey," *arXiv preprint arXiv:2101.10382,* 2021.

[55] L. Jiang, D. Meng, Q. Zhao, S. Shan, and A. G. Hauptmann, "Self-paced curriculum learning," presented at the Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, Texas, 2015.

[56] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," presented at the Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 1, Vancouver, British Columbia, Canada, 2010.

[57] Y. Liu, M. Shi, Q. Zhao, and X. Wang, "Point in, box out: Beyond counting persons in crowds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6469-6478.

[58] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025,* 2015.