# 第六章 瑕疵分割原理與實作

## 前言

本章將從捲積網路用於圖像分割的歷史，介紹各種網路架構以及其內含的技術的內容。首先以下會介紹何謂分割以及分割常見的類型。接著開始進入主要章節，有各類型的分割網路，以及筆者認為其重要的技術或貢獻。最後提到如何將分割技術用於工業上瑕疵分類的應用以及實作方法。

### 何謂分割?

分割是將一幅圖像分割爲若干個部分的過程，這種圖像處理過程可以得到圖像中的目標或者紋理，常常被用於遙感影像或者腫瘤的檢測應用中。

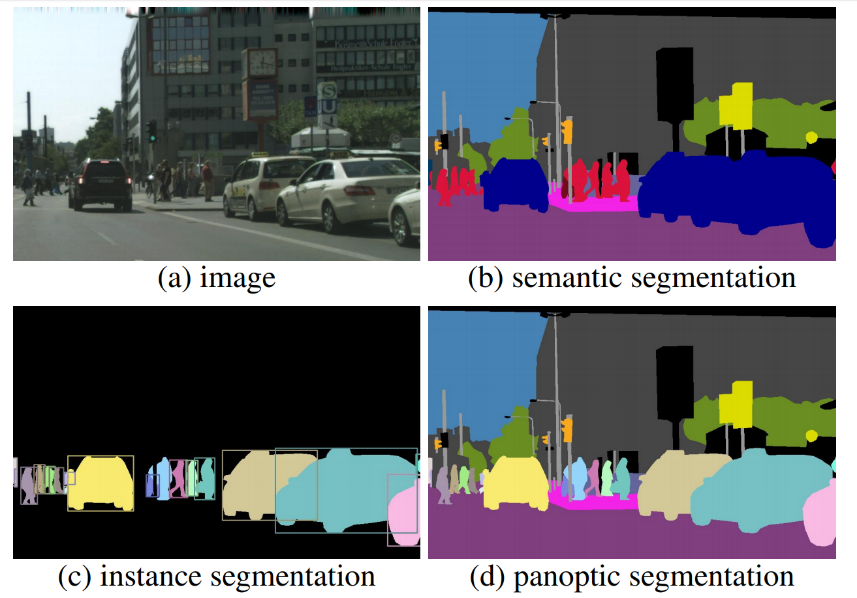
目前已經有了很多傳統方法可以實現這一過程，例如：邊緣檢測方法、閾值分割法、基於區域的聚類、基於像素的聚類、形態學的方法等等。同時，目前也出現了很多利用卷積神經網絡進行分割的方法，這些方法已成爲解決圖像分割中更高級任務中不可或缺的方法。

目前使用卷積神經網絡的影像分割技術，包含：語意分割(semantic segmentation)、實例分割(instance segmentation)、全景分割(panoptic segmentation)，其分割結果如下圖●所示，以下再對這三者詳細說明：

一、語意分割(見下圖b)：對每個像素給予分類，呈現出來的結果會將同個“類別”的景物以相同顏色表示，包含背景及物體，常見例子如FCN。

二、實例分割(見下圖c)：先以物件偵測(object detection)的方法找出物體，再對每個物體給予分類，呈現出來的結果會將每個“物體”以不同顏色表示，而不包含背景，常見例子如Mask RCNN。

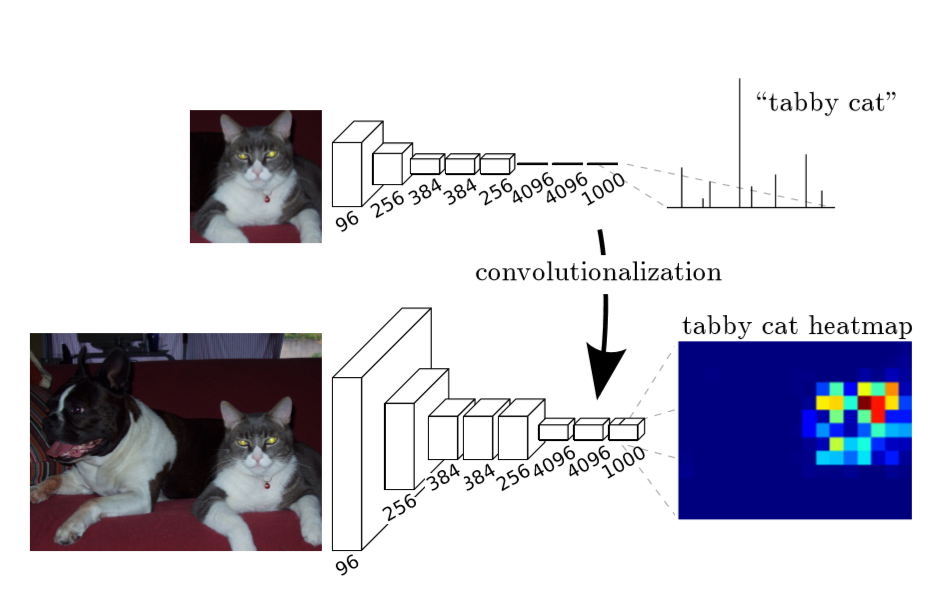
三、全景分割(見下圖d)：結合語意分割與實例分割，即在網路內同時執行兩個子任務(語意分割與實例分割)，於最後融合子任務結果，呈現出包含實例分割將每個“物體”以不同顏色表示，並同時包含語意分割的背景及物體，常見例子如Panoptic FPN。



圖●、使用AI的不同影像分割技術之呈現結果

## 6.1全卷積網路(Fully Convolutional Networks)

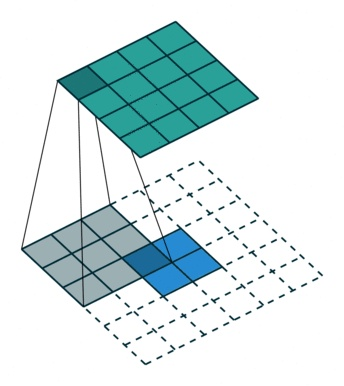
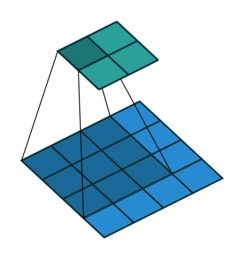
Jonathan Long等人提出用於圖像的全卷積網絡（FCN）語義分割(sematic segmentation)。由於內容闡述了如何對CNN進行端到端(end-to-end)訓練以達到像素級影像分類的功能，此篇論文被視為是CNN用於像素級分割的里程碑。FCN架構是根據傳統的CNN分類器開發的，其主要使用了三種技術：（1）卷積層、（2）上採樣層、（3）捷徑連結層。如下圖●所示，上方的AlexNet網路為一種常見的CNN分類器結構，輸入227x227x3的圖片並經過捲積層(convolution layer)、池化層(pooling layer)、全連接(fully connected layer)對圖片進行分類，最終輸出所屬類別的機率。而下方的全卷積網絡將最後的全連接層替換成捲積層，藉此可以輸出分類的熱點圖(heat map)。



圖●、AlexNet結構網路的CNN分類器與FCN分割

### 上採樣 upsampling

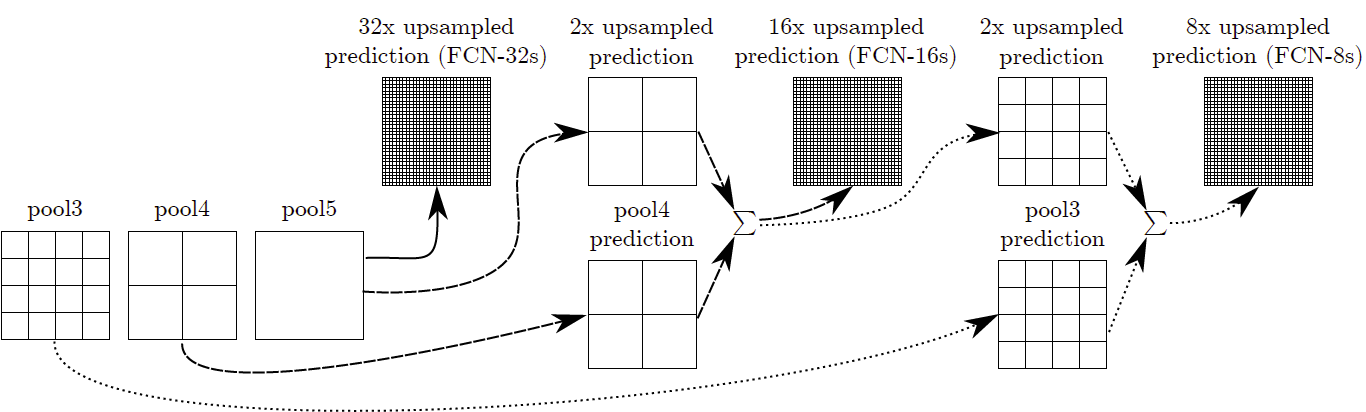
相比於下採樣是使圖片越來越小的操作(如捲積、池化)，上採樣是使得整個圖片越來越大的操作，例如在論文中經過卷積和池化以後，圖像的解析度縮小成1/32倍，對於最後一層的輸出特徵圖，需要進行32倍的上採樣，以放大得到原圖一樣的大小。這個上採樣是通過反卷積（deconvolution）實現的，反卷積是透過將原圖像做填充(padding)後，再使用一般的捲積方法得到輸出圖像，輸出圖像會較輸入圖像大，但也會較輸入圖像填充後的圖像小，即反卷積其實就是有填充的捲積。如下圖●所示，左圖為一般常見的捲積，藍色為輸入綠色為輸出；右圖則為反捲積，有較多的填充。



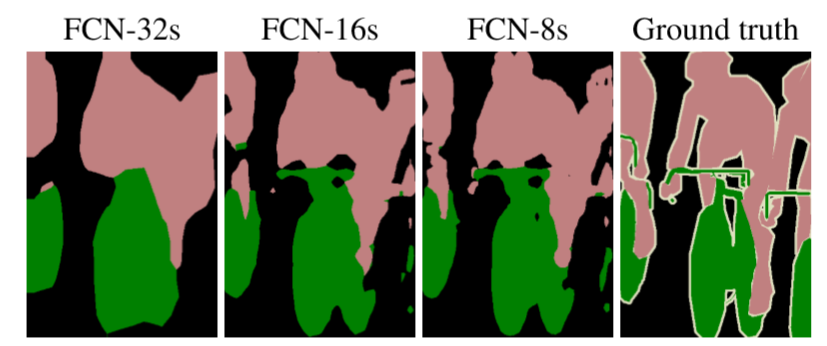
圖●、捲積(左)與反捲積(右)

### 跳躍結構 skip architecture

在論文中，圖像最終的解析度縮小成1/32倍，需要進行32倍的上採樣，以放大得到原圖一樣的大小，但卻因此使得輸出的分類圖(label map)較為粗糙，在下圖●中名為FCN-32s。原因為經過多次下採樣而較深層的特徵圖(feature map)可以獲得較複雜的特徵結構(features)，但也會丟失較多的空間位置信息(location information)；相反的，較淺層則擁有較多的空間位置信息。為解決空間位置信息丟失的問題，在論文中提出了跳躍結構(skip architecture)，將淺層的特徵圖直接提取作為最後的輸出特徵，階段性進行上採樣(每次只做2x)再合併，合併後再透過上採樣，最終得到與原圖相同的解析度。從下圖●可以看出，FCN-8s在跳躍結構中融合了淺層的空間位置信息，可以恢復出較佳的邊緣輪廓。



圖●、跳躍結構的合併過程



圖●、經過跳躍結構處理後的結果

## 6.2 U Net

從FCN延伸Olaf Ronneberger等人提出了U-Net網路結構，U-Net網路結構前半部分作用是特徵提取(feature extraction)，後半部分是上採樣(upsampling)，這樣的結構也叫做編碼器-解碼器結構(encoder-decoder)，由於對稱性的結構加強了還原性，並且沿用了FCN的跳躍結構加上不同的拼接方法。此網路整體結構類似於大寫的英文字母U如下圖●所示，故得名U-Net。



圖●、U-Net網路結構

### 拼接concatenate

相比於全卷積網路(FCN)的跳躍結構(skip architecture)是採用直接相加(sum)特徵圖的方式，U-Net採用了完全不同的特徵融合方式—拼接(concatenate)，U-Net採用將特徵在通道(channel)維度拼接在一起，形成更厚的特徵。而相比於FCN融合時使用的對應點相加，則並不會形成更厚的特徵。如下圖●將U-Net網路結構的一部分取出來看，其中28\*28\*1024的B透過反捲積成2x大小56\*56\*512的B’，而後在將下採樣過程得到的64\*64\*512的特徵圖透過跳躍連結(skip connection)至上採樣過程，並裁切(crop)成56\*56\*512的A’以符合B’大小，最終將A’(56\*56\*512)與B’(56\*56\*512)在通道維度拼接成56\*56\*1024的向量(tensor)。



B

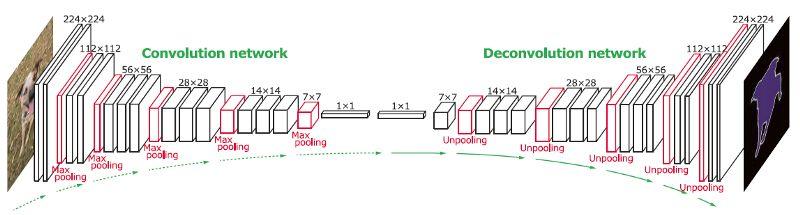
A’ B’

A

圖●、U-Net網路中拼接的範例

## 6.3 DeconvNet

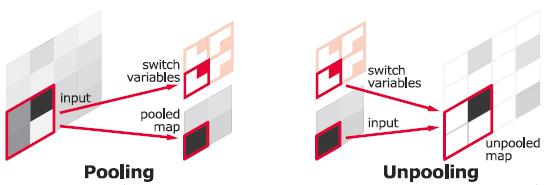
Hyeonwoo Noh等人提出了反捲積網絡（DeconvNet）由反捲積層(deconvolution)和上池化層(unpooling)組成。DeconvNet網絡使用VGG作為其骨幹框架(backbone)。第一部分是捲積網絡(convolution network)，沿用自FCN具有常見的捲積和池化層；第二部分是反卷積網絡(deconvolution network)，這是本文中的一個新穎部分，而這兩者的中間有全連接層(fully connected layer)作為銜接。



圖●、DeconvNet網絡架構

### 反池化unpooling

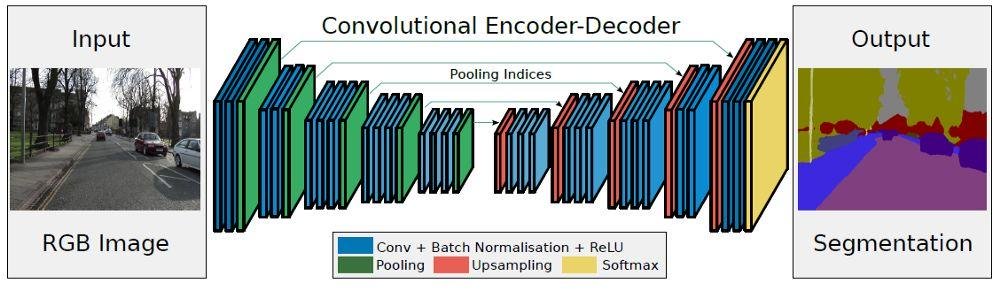
在捲積網絡的下採樣過程，最大池化(max pooling)過程中模型會記住最大池化時每個最大激活值的位置信息（如下圖●左），這樣的位置信息在論文中稱作池化索引(pooling index)。而在反捲積網絡的上採樣時，反池化(unpooling)會使用池化索引來還原（如下圖●右），將特徵放回池化時有最大激活值的位置。



圖●、DeconvNet中的池化與反池化

## 6.4 SegNet

Vijay Badrinarayanan等人提出了SegNet，該網路架構具有編碼器網絡和相應的解碼器網路，接著是按最終像素的分類層(見下圖●)。在編碼器處包含捲積和最大池化，網路結構採用VGG-16共有13個捲積層，但沒有全連接層(fully connected layer)；而在進行2×2最大池化(max pooling)時，也有儲存池化索引(pooling index)。SegNet其架構與DeconvNet有很大的相似之處，最大差異在於SegNet沒有全連接層，而全連接層所編碼的資訊過於抽象，解碼以後會引入大量雜訊。因此SegNet可以獲得更好的分割效果，同時又減少了(全連接層的)參數量。



圖●、SegNet網路架構

## 6.5 DeepLab

### 空洞捲積 atrous convolution

## 6.6 Mask RCNN

## 6.7 Panoptic Feature Pyramid Networks