SSD(Single Shot MultiBox Detector)

* 目標檢測分為One-Stage & Two-Stage，
* One-stage為均勻在圖片不同位置進行密集抽樣，採用不同尺度和長寬比，使用CNN提取特徵後進行分類與回歸，由於是直接抽樣方式進行特徵提取，運行速度快，但訓練困難，因為正樣本與負樣本(背景)數量不平均，導致模型準確偏低，例如) YOLO, SSD
* Two-stage為通過RPN產生一系列候選框，(先通過CNN卷積後獲得feature map，根據Region proposal network產生所有候選框進行分類前景和背景圖與4個位置信息，經由ROI pooling後去除不為類別框)，難收斂(存在沒用的類別(背景))，無用的region數量大，不能為softmax分類器效能提升(預測都是背景)，大量耗費計算時間
* SSD主要三個重點為

1. **Multi-scale feature map for detection多尺度特徵圖預測**: 擷取不同特徵層進行預測目標位置，擷取的層包含Conv4\_3(38x38), Conv7(19x19), Conv8\_2(10x10), Conv9\_2(5x5), Conv10\_2(3x3), Conv11\_2(1x1)六個feature layers，前面層feature map尺寸大適合檢測小目標(可劃分更多cell，因為先驗框尺寸小)，經過stride or pooling 逐漸降低feature map尺寸適合檢測大目標
2. **Convolutional predictors for detection檢測卷積預測器:** 透過3x3xp小卷積生成類別分數與相對於default box座標
3. **Default boxes and aspect ratios:** 在回歸與分類的不同層定義不同default box個數，feature map cell 設置尺度or aspect ratios 不同的default box，aspect ratios為{1,2,3,1/2, 1/3}，預測BB(Bounding Box)以default box為基準，減少訓練難度，每個feature map cell有k個default box，有(c+4)\*k個預測值，需(c+4)\*k conv. 在SSD300共可預測=8732 Bounding Boxes

* **如何生成預測框?**
  + 為了適應不同尺寸的目標，擷取不同feature map層不同的receptive field尺寸檢測，由於default box不需要對應每層實際的receptive field，透過tiling平鋪概念將特定feature map對應到特定的尺寸，scale of default box=[0.2, 0.37, 0.54, 0.71, 0.88, 1.05]，以及width & height轉換後的default box，

Conv4\_3: min\_size = 300\*0.1 = 30, max\_size = 300\*0.2 = 60

Fc7: min\_size = 300\*0.2 = 60, 300\*0.37 = 111

Conv6\_2: min\_size=300\*0.37=111, max\_size = 300\*0.54=162

Conv7\_2: min\_size = 162, max\_size = 300\*0.71 = 213

Conv8\_2: min\_size = 213, max\_size = 264

Conv9\_2: min\_size = 264, max\_size = 315

**Aspect\_ratios = [[2], [2,3], [2,3],[2,3],[2],[2]]**

**\*\***每個feature map cell 會產生2個不同大小的正方形(小邊長為min\_size，大邊長為sqrt(min\_size\*max\_size))默認框，而每個aspect\_ratios會增加兩個默認框，以conv4\_3為例，默認生成兩個默認框，aspect\_ratios個數為1，又生成兩個長方形(height = 1/sqrt(aspect\_ratio)\*min\_size, width=sqrt(aspect\_ratio)\*min\_size)默認框，共四個，因此6個feature map分別生成 4,6,6,6,4,4個默認框，擷取的層包含Conv4\_3(38x38), Conv7(19x19), Conv8\_2(10x10), Conv9\_2(5x5), Conv10\_2(3x3), Conv11\_2(1x1)，對應相乘默認框後共8732個框

* + 以及使用horizontal flip(水平翻轉)，random crop & color distortion(隨機剪裁加顏色扭曲)，randomly sample a patch(隨機採集塊域獲取小目標訓練樣本)，製作數據多樣性，
* **生成預測框後如何匹配相對應的GT?**
  + 使用Jaccard()簡稱IoU(Intersection over Union)，將預測框選擇任何一個GT threshold >0.5 ， 一個GT可搭配多個預測框，而一個預測框只能搭配一個GT，但由於預測框太多，且大多數預測框皆為負樣本，接著過濾屬於背景預測框，過濾threshold<0.5，過濾後的框進行解碼得到真實位置參數(為防止預測框超出圖片，使用clip)，透過置信度由高到低排列選取top-N，進行NMS(提取置信度高的目標檢測框，抑制置信度低的誤檢框)過濾重疊較大的預測框，使得正負樣本在1:3，此比例優化速度快與訓練穩定，減少不必要的計算
* **如何定義loss?**
  + 以locatization loss & confidence loss加權和**，**
  + Lloc針對正樣本進行計算，使用smooth L1 loss計算位置誤差
  + Lconf採用softmax loss

\*\*\*\*\*\*\*NMS\*\*\*\*\*\*\*\*

NMS(Non-Maximum Suppression)非極大值抑制

抑制不是極大值元素，在目標檢測中，**提取置信度高的目標檢測框，抑制置信度低的誤檢框**，目標框數量由anchor數量決定，nms用來去除重複的框(重複框定位到同一目標)，

以下為平行於座標軸:

1. **標準nms**
2. **Soft-nms**: 將相鄰低智信度檢測框分數均強制歸零，如同類目標比較密集，存在遮擋，如密集人群，用此方式處理容易產生漏洞，soft-nms吸取nms教訓，算法執行過程中，不是簡單對IoU > threshold檢測框刪除，而是降低分數
3. **Adaptive nms**: 與soft-nms解決問題一樣，而soft-nms智信度threshold = Nt主要依賴人工設定，不能適應目標壅擠程度不同的情況，adaptive nms根據目標密度來自動設定置信度threshold

以下為傾斜框(rotate box, rbox)的nms:

1. **INMS**: 當前遍歷rbox與剩餘rbox進行交集運算得到相應的相交點集合，並根據判斷相交點集合組成凸邊形面積
2. **PNMS(Polygon nms)**: 物體無法用矩形表達，而是多邊形，計算IoU方式相同，得到兩個多邊形相交點集，判斷相交點集合組成的凸邊形面積，計算兩多邊形IoU，可使用boost酷的geometry的polygon相關函數實現，也可使用shapely.geometry的polygen
3. **MNMS(mask nms)**: 分割區域

\*\*\*\*\*\*\*NMS\*\*\*\*\*\*\*\*