

# 模式识别与深度学习 (22)

## 物体检测

王昊、左旺孟

综合楼309

视觉感知与认知组

哈尔滨工业大学计算机学院

[cswmzuo@gmail.com](mailto:cswmzuo@gmail.com)

13134506692



# 物体检测

## ➤ 物体检测

- 基于手工提取特征的方法
- 基于深度学习提取特征的方法
- 一些新的方法和进展

# 物体检测

物体种类分类 + 边界框定位

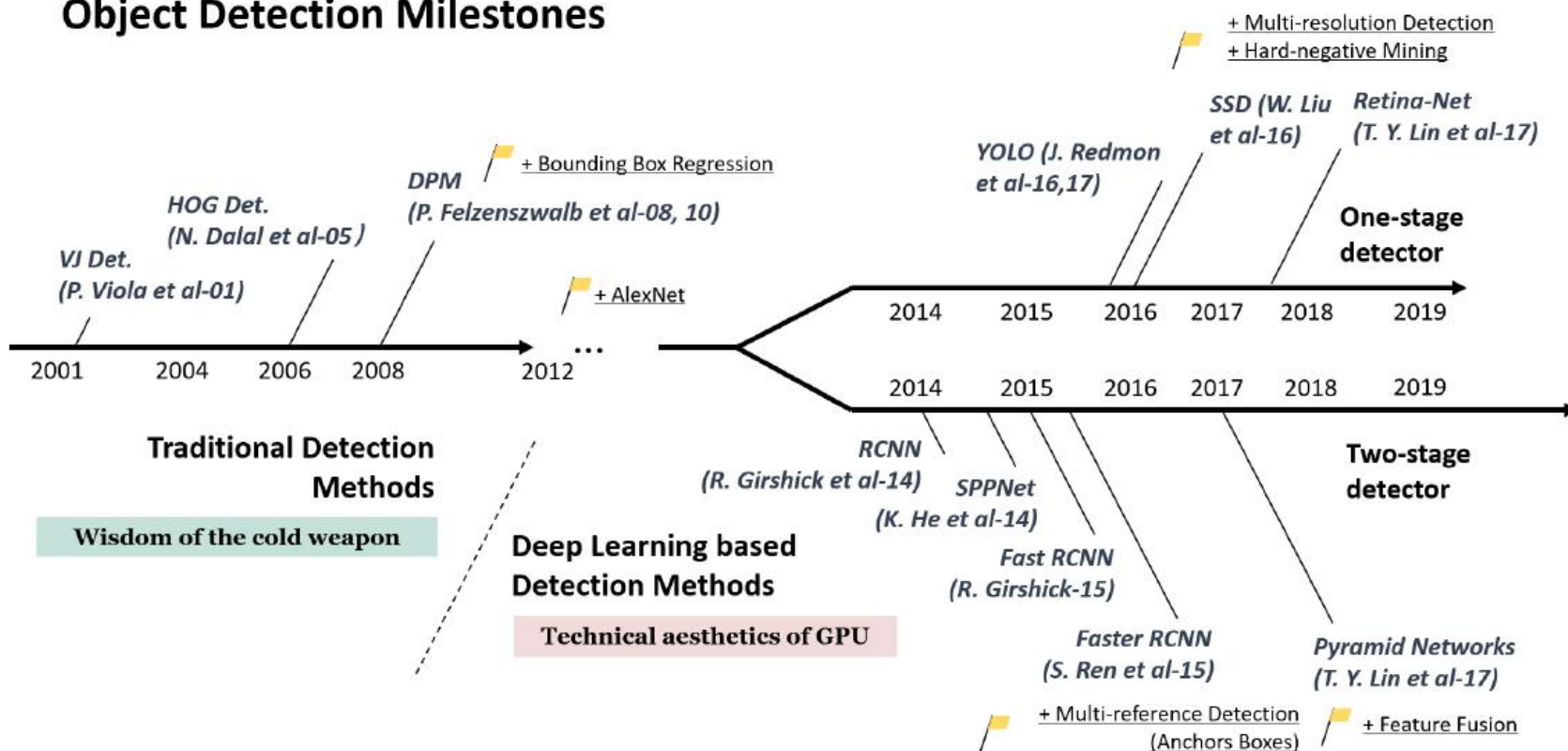


种类：猫

边界框坐标：(34,0,96,128)

# 物体检测

## Object Detection Milestones



# 基于手工提取特征的方法

## Selective Search

---

**Algorithm 1:** Hierarchical Grouping Algorithm

---

**Input:** (colour) image

**Output:** Set of object location hypotheses  $L$

Obtain initial regions  $R = \{r_1, \dots, r_n\}$  using [13]

Initialise similarity set  $S = \emptyset$

**foreach** *Neighbouring region pair*  $(r_i, r_j)$  **do**

    Calculate similarity  $s(r_i, r_j)$

$S = S \cup s(r_i, r_j)$

**while**  $S \neq \emptyset$  **do**

    Get highest similarity  $s(r_i, r_j) = \max(S)$

    Merge corresponding regions  $r_t = r_i \cup r_j$

    Remove similarities regarding  $r_i$ :  $S = S \setminus s(r_i, r_*)$

    Remove similarities regarding  $r_j$ :  $S = S \setminus s(r_*, r_j)$

    Calculate similarity set  $S_t$  between  $r_t$  and its neighbours

$S = S \cup S_t$

$R = R \cup r_t$

Extract object location boxes  $L$  from all regions in  $R$

---

1. 解决了基于滑窗方法需要提取大量预选框，计算量大的问题；
2. 对于长宽比例不固定的物体有很好的鲁棒性。

# 基于深度学习提取特征的方法

## 1. 基于深度学习的两阶段物体检测方法

- 1). 首先在图像上选取多个区域预选框;
- 2). 然后对预选框内的候选区域进行分类及坐标回归, 从而得到最终的检测结果。

主要框架: R-CNN, SPP, Fast R-CNN, Faster R-CNN, MLKP, FPN, Mask R-CNN等

## 2. 基于深度学习的一阶段物体检测方法

将图像输出到卷积神经网络中, 只经过一次前向传播, 不需要进行区域预选框选取, 直接在神经网络输出特征上进行回归, 得到其坐标值及其对应的分类置信度。

主要框架: YOLO, SSD, RetinaNet等

# 基于深度学习提取特征的方法

## 两阶段物体检测网络

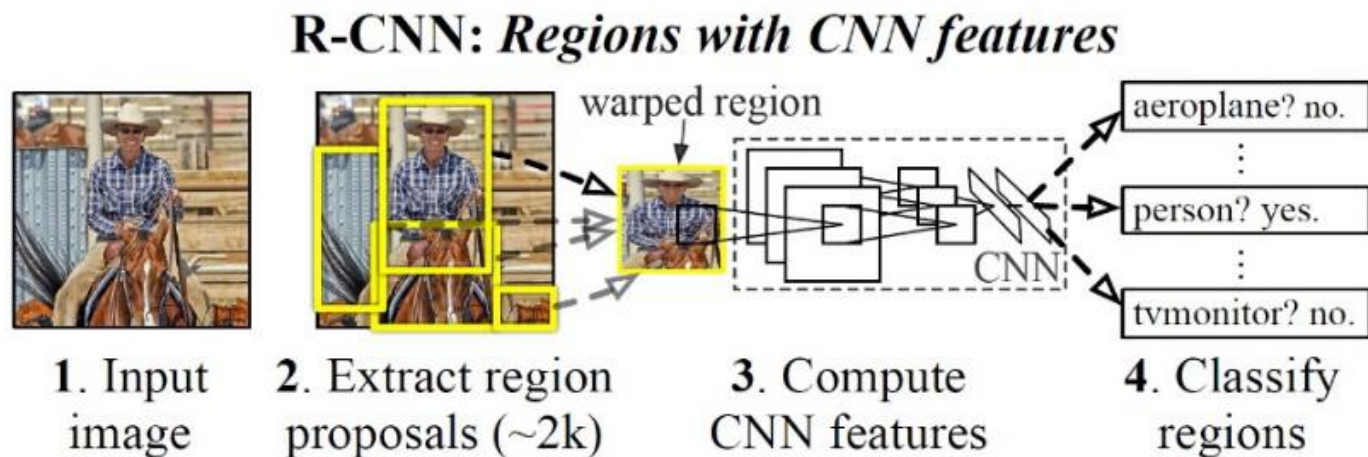
- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ Faster R-CNN
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络

# 基于深度学习提取特征的方法

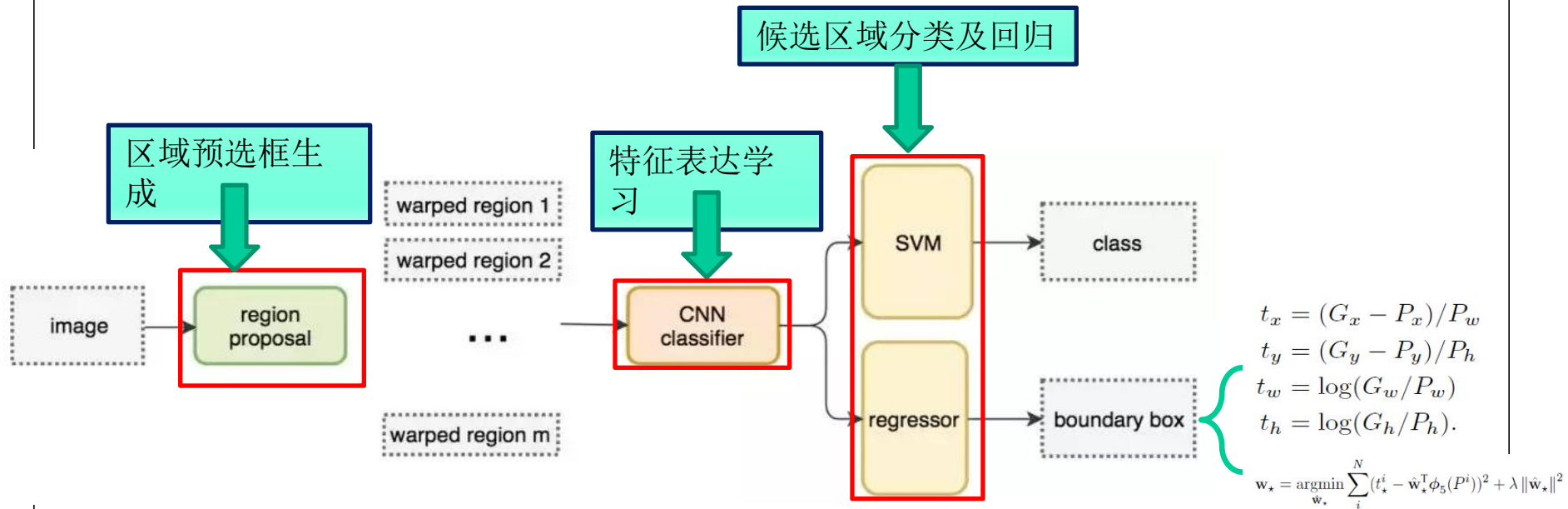
## R-CNN





# 基于深度学习提取特征的方法

## R-CNN

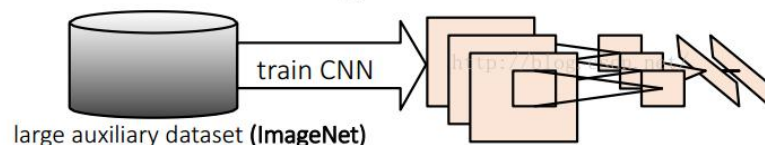


# 基于深度学习提取特征的方法

## R-CNN

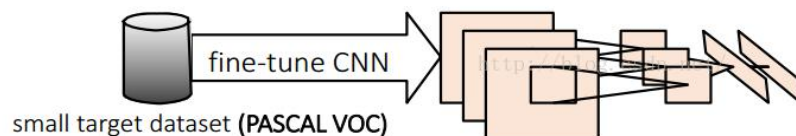
Step1: 基于监督信息的模型预训练

1. Pre-train CNN for image classification

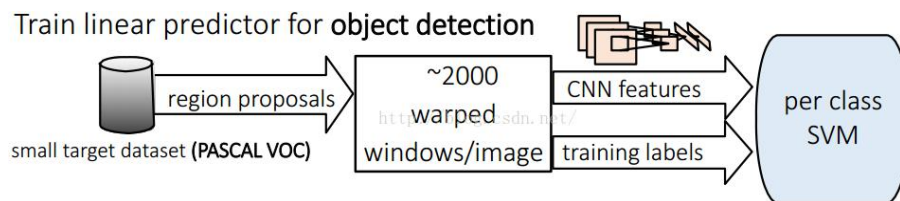


Step2: 基于特定任务的参数微调

2. Fine-tune CNN for object detection



Step3: 训练特定种类的分类器



# 基于深度学习提取特征的方法

## R-CNN

### R-CNN 缺陷:

- **测试速度慢:** R-CNN把一张图像分解成大量的候选框，每个候选框拉伸成一个固定大小的图像，并单独通过深层卷积神经网络完成特征表达。这些候选框之间往往存在大量的重叠，特征表达之间通常可以共享，因此造成了计算资源的浪费
- **训练速度慢:** R-CNN的训练分为3个独立的阶段，特别是在最后一个阶段，需要把预先提取的特征表达存储在硬盘上，然后训练SVM。这种方法需要在硬盘上大量的读写数据，会造成训练速度缓慢，使得训练性能非常低。
- **训练所需存储空间大:** R-CNN中独立的SVM分类器和回归器需要大量特征作为训练样本，需要大量的存储空间。
- **裁剪候选框造成精度损失:** 候选区域的长宽不固定，故需要对候选区域做填充到固定大小，对候选区域做cropping或者warping操作，可能会让图片不完整包含物体，及出现几何失真，这都会造成识别精度损失。

# 基于深度学习提取特征的方法

## 两阶段物体检测网络

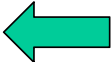
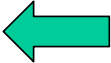
- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ Faster R-CNN
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络

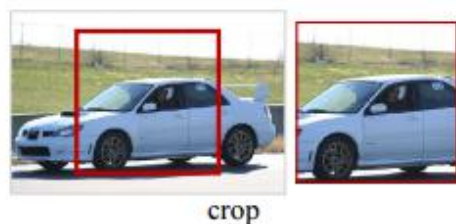
# 基于深度学习提取特征的方法

## From R-CNN to SPP

- 使用“空间金字塔变换层”将接收任意大小的图像输入，输出固定长度的输出向量，这样就能让SPPNet可接受任意大小的输入图片，不需要对图像做crop/wrap操作。  
 解决需要固定输入尺寸问题
- SPPNet提出了一种从候选区域到全图的特征映射(feature map)之间的对应关系，通过此种映射关系可以直接获取到候选区域的特征向量，不需要重复使用CNN提取特征，从而大幅度缩短训练时间。  
 解决训练速度慢的问题

# 基于深度学习提取特征的方法

## SPP



← Cropping or warping may decrease the performance



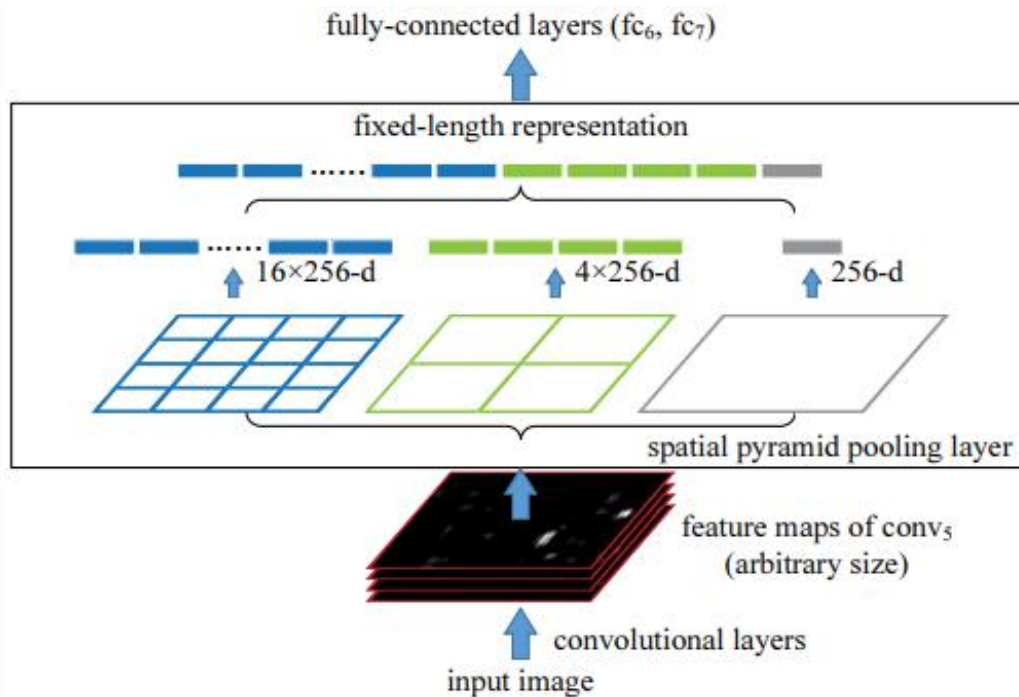
← Conventional CNN



← SPP

# 基于深度学习提取特征的方法

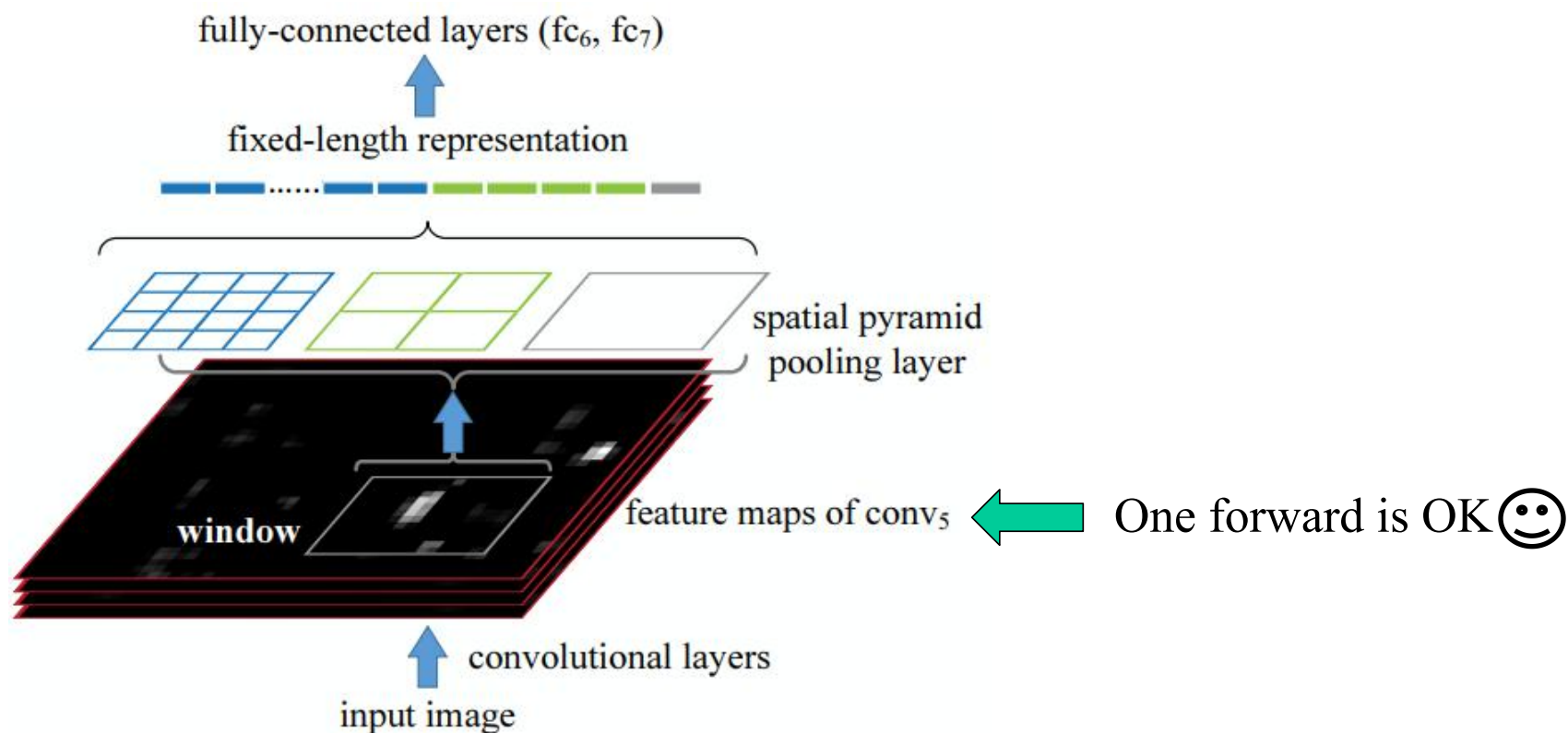
## SPP



← Arbitrary Size is OK 😊

# 基于深度学习提取特征的方法

## SPP





# 基于深度学习提取特征的方法

## SPP

It is good, but not the best ☹️

- 与R-CNN相同，训练分为3个独立的阶段，在最后一个阶段，需要把预先提取的特征表达存储在硬盘上，然后训练SVM。独立的SVM分类器和回归器需要大量特征作为训练样本，需要大量的存储空间
- 特征提取CNN的训练和SVM分类器的训练在时间上是先后顺序，两者的训练方式独立，因此SVMs的训练损失无法更新SPP-Layer之前的卷积层参数，故这些卷积层参数在训练中无法更新。

# 基于深度学习提取特征的方法

## 两阶段物体检测网络

- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ Faster R-CNN
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络

# 基于深度学习提取特征的方法

## From R-CNN/SPP to Fast R-CNN

- Fast R-CNN将整张图像归一化后直接送入深层卷积神经网络，在卷积层输出的特征图上根据候选框进行RoI-Pooling，一次性为所有候选框生成特征表达，显著提升了测试速度；  
← 解决速度慢和需要固定输入尺寸的问题
- Fast R-CNN在训练时，只需要将一张图像送入深层卷积神经网络，一次性地提取候选区域和特征表达，训练数据在GPU内存里直接输入到损失层，这样候选区域的前几层特征不需要再重复计算，同时不再需要把大量数据存储在硬盘上。  
← 解决存储空间大的问题
- Fast R-CNN把类别判断和位置回归统一用深度网络实现，不再需要额外存储。  
← 解决存储空间大和不能更新卷积层的问题

Everything goes well

# 基于深度学习提取特征的方法

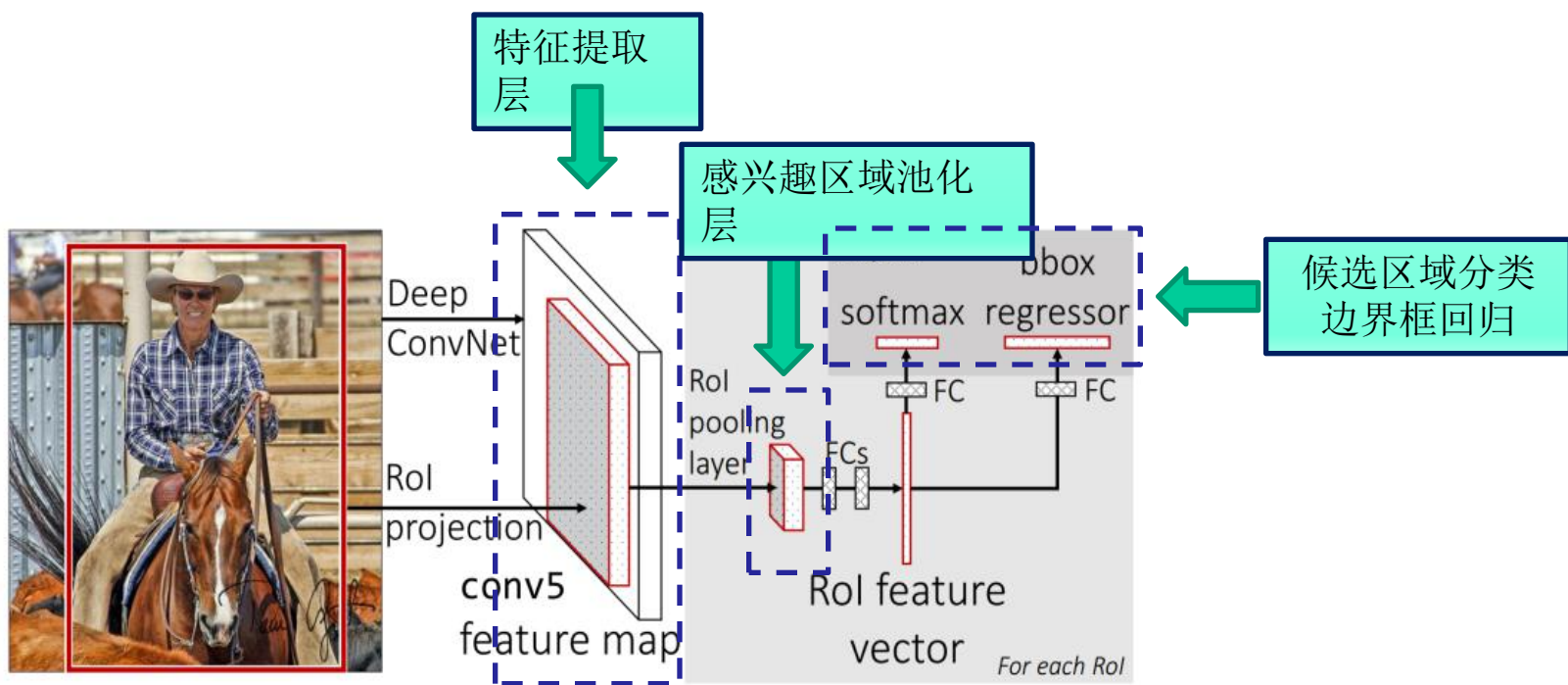
## From R-CNN/SPP to Fast R-CNN

### Fast R-CNN优点:

- 特征表达不再是对每个区域候选框单独进行，而是直接对整张图像一次性完成，减少了大量的重复计算；
- 用ROI-Pooling代替候选区域的尺寸变换，保持了候选区域比例，避免了图像变形失真带来的精度损失；
- 将分类器学习和边界框回归设计到一个统一的框架内进行联合训练，相比原来的分离训练策略，既提高了训练效率又提高了预测精度。

# 基于深度学习提取特征的方法

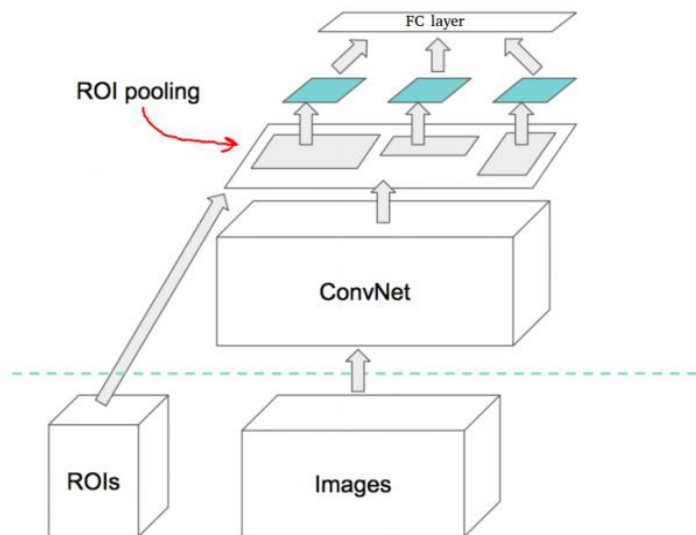
## Fast R-CNN



# 基于深度学习提取特征的方法

## Fast R-CNN

感兴趣区域池化层

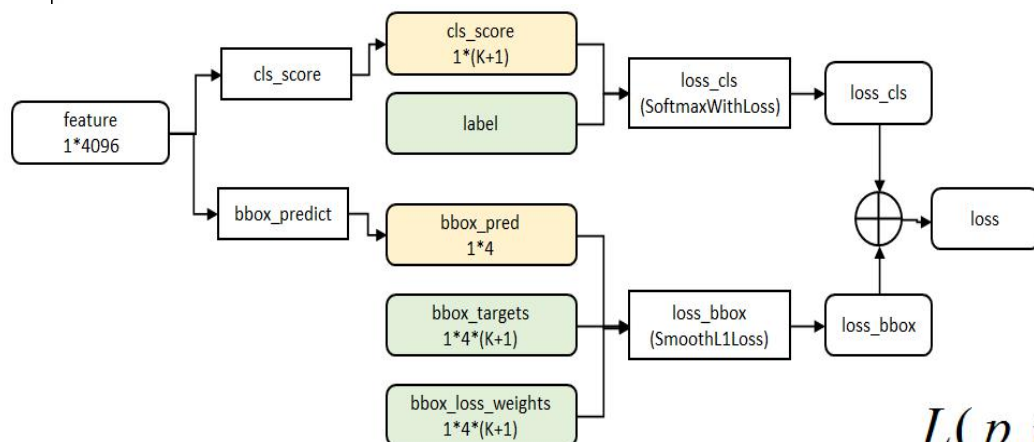


$$\frac{\partial L}{\partial x_i} = \sum_r \sum_j [i = i^*(r, j)] \frac{\partial L}{\partial y_{rj}}$$

# 基于深度学习提取特征的方法

## Fast R-CNN

候选区域分类及边界框回归



$$L_{\text{cls}}(p, u) = -\log p_u$$

$$L_{\text{loc}}(t^u, v) = [u \geq 1] \sum_{i \in \{x, y, w, h\}} \text{smooth}_{L1}(t_i^u - v_i)$$

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases}$$

$$L(p, u, t^u, v) = L_{\text{cls}}(p, u) + \lambda [u \geq 1] L_{\text{loc}}(t^u, v)$$

# 基于深度学习提取特征的方法

## 两阶段物体检测网络

- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ **Faster R-CNN**
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络



# 基于深度学习提取特征的方法

## From Fast R-CNN to Faster R-CNN

### Fast R-CNN 缺陷:

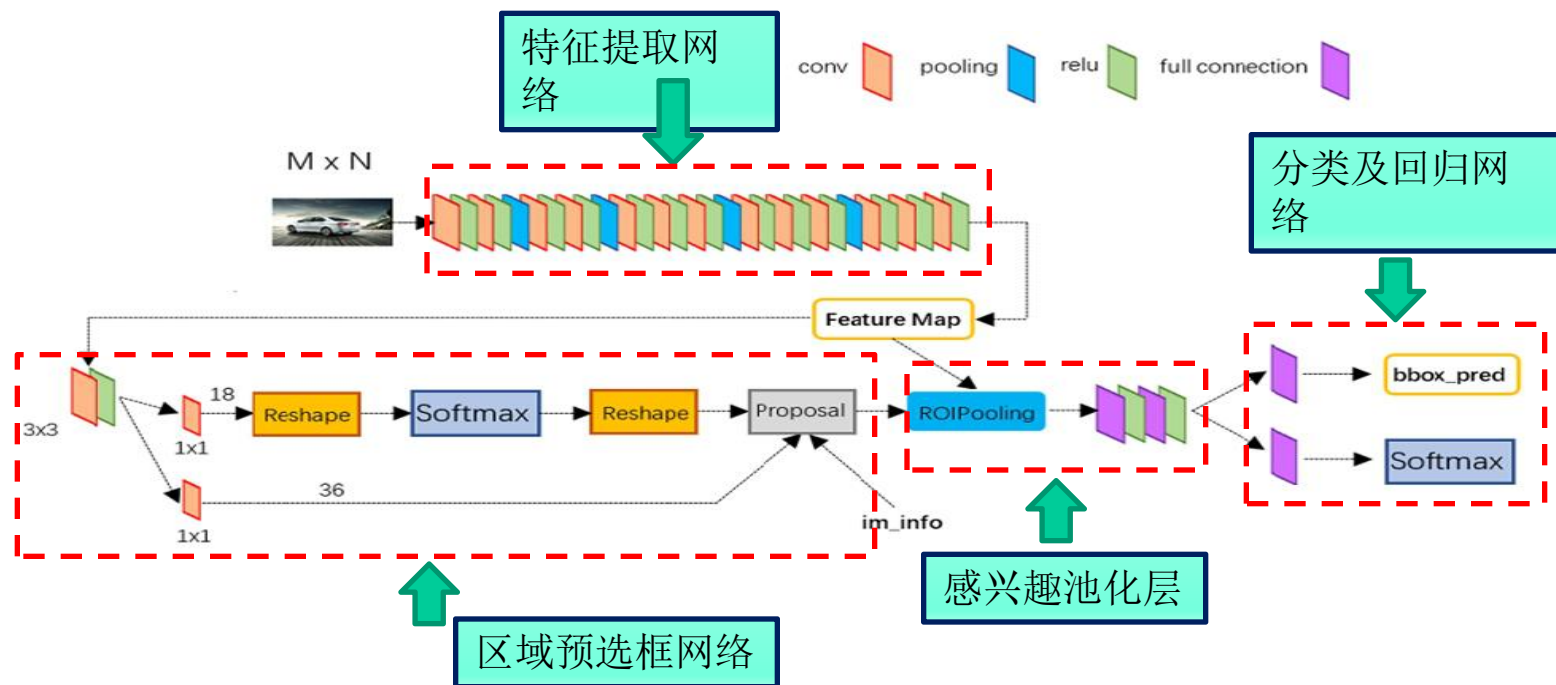
- 需要离线提取预选框;
- 无法端到端进行训练。

### Faster R-CNN优点:

- 提出了区域预选框网络 (Region Proposal Network, RPN) 代替了Selective Search方法;
- 将目标检测的四个基本步骤 (预选框生成, 特征提取, 分类及回归) 整合到一个网络结构中, 真正的实现了端到端的目标检测框架

# 基于深度学习提取特征的方法

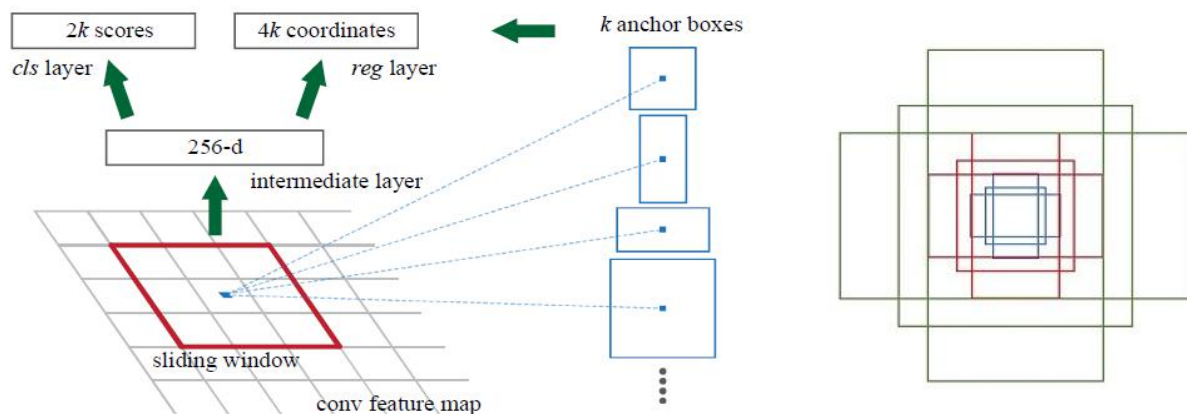
## Faster R-CNN



# 基于深度学习提取特征的方法

## Faster R-CNN

### 区域预选框网络



anchor	$128^2, 2:1$	$128^2, 1:1$	$128^2, 1:2$	$256^2, 2:1$	$256^2, 1:1$	$256^2, 1:2$	$512^2, 2:1$	$512^2, 1:1$	$512^2, 1:2$
--------	--------------	--------------	--------------	--------------	--------------	--------------	--------------	--------------	--------------

# 基于深度学习提取特征的方法

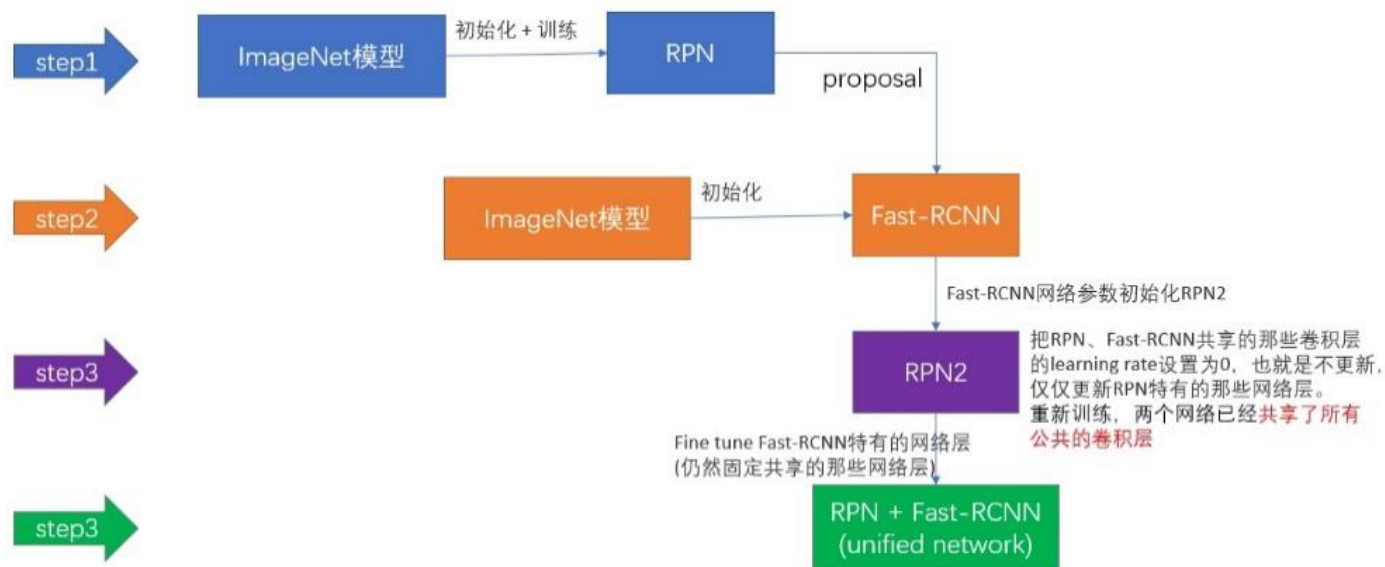
## Faster R-CNN

### 训练方法

#### 1. 端到端训练

网络中所有的参数在一次迭代中全部更新，RPN网络及Fast R-CNN网络的损失同时对网络参数进行更新，这种方法实现简单，训练速度快。

#### 2. 四步迭代法



# 基于深度学习提取特征的方法

## 两阶段物体检测网络

- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ Faster R-CNN
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络

# 基于深度学习提取特征的方法

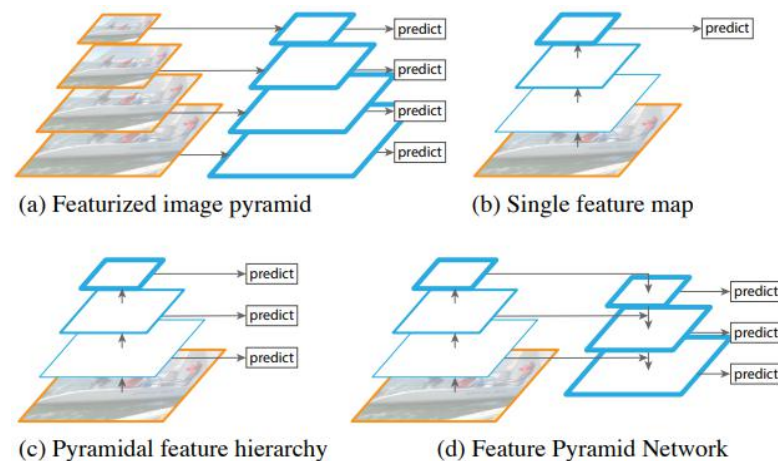
## 从单尺度预测到多尺度预测

单尺度检测：

- 单尺度特征的目标检测检测方法对无法感知检测物体尺度的变化，影响检测结果；
- 现有的多尺度检测方法时间复杂度高，训练及测试代价大。

多尺度检测：

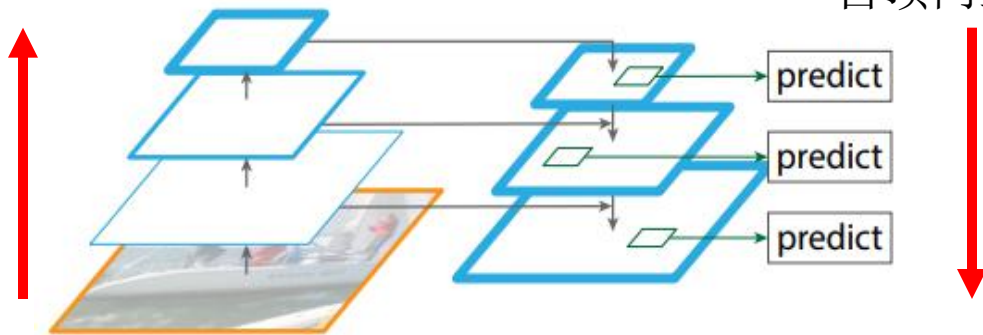
- 通过自顶向下和自底向上的结构得到得到空间分辨率不同的包含语义信息丰富的特征图；
- FPN可以端到端的进行训练，不额外增加测试时间，也大幅度的提高了目标检测结果。



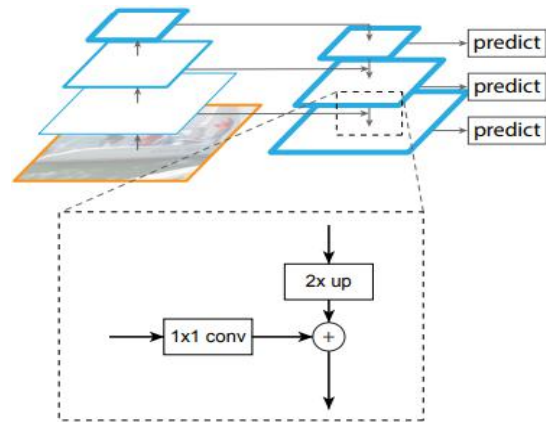
# 基于深度学习提取特征的方法

## FPN

自底向上结构



自顶向上结构



# 基于深度学习提取特征的方法

## FPN

1. 与RPN结合
2. 与Fast R-CNN结合

$$k = \lfloor k_0 + \log_2(\sqrt{wh}/224) \rfloor \quad \leftarrow$$

尺寸较大的目标会被分配到空间分辨率较低的特征层进行预测，尺寸较小的目标会被分配到空间分辨率较高的特征层进行预测，这种分配方式对小物体检测提升明显。



# 基于深度学习提取特征的方法

## 两阶段物体检测网络

- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ Faster R-CNN
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络

# 基于深度学习提取特征的方法

## 从单任务优化到多任务优化

### 单任务优化：

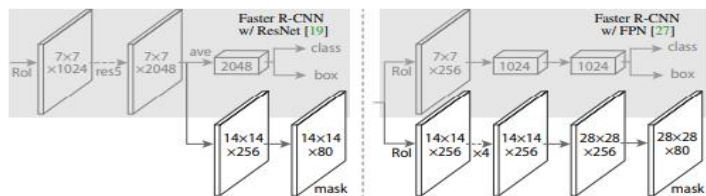
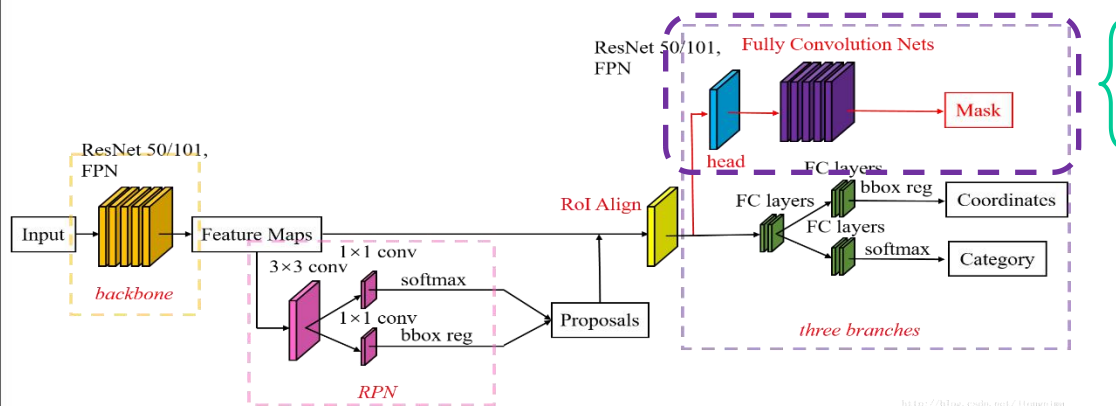
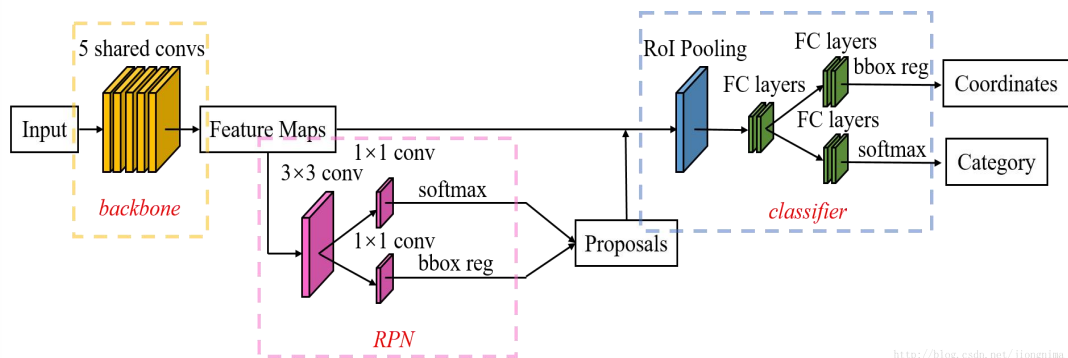
- 实例分割更具有挑战性，现有的方法都是将分割与检测两个任务单独处理；
- RoI Pooling层中的量化过程造成像素不能对齐，影响检测结果。

### 多任务优化：

- 将实例分割和物体检测两个任务在一个统一的网络中完成，且都取得了当时最好的结果；
- 解决RoI-Pooling中像素点不能对齐的问题，提出了RoI-Aligning层，可以准确的保证空间位置。

# 基于深度学习提取特征的方法

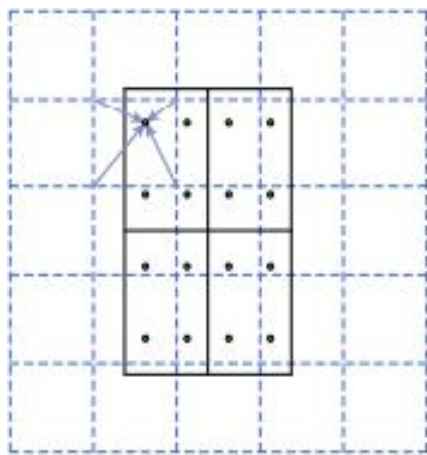
## Mask R-CNN



# 基于深度学习提取特征的方法

## Mask R-CNN

### RoI-Align层



- 1). 将区域预选框映射到卷积神经网络输出特征上，将其边界量化为整数点坐标值；
- 2). 将量化后的区域预选框区域平均分割成 $7 \times 7$ 个单元 (bin)，对每个单元的边界量化。



- 1). 遍历每一个区域预选框，保持浮点数边界不做量化；
- 2). 将区域预选框分割成 $k \times k$ 个单元，每个单元的边界也不做量化处理。

$$\frac{\partial L}{\partial x_i} = \sum_r \sum_j [i = i^*(r, j)] \frac{\partial L}{\partial y_{rj}} \quad \longrightarrow \quad \frac{\partial L}{\partial x_i} = \sum_r \sum_j [d(i, i^*(r, j)) < 1] (1 - \Delta h)(1 - \Delta w) \frac{\partial L}{\partial y_{rj}}$$

# 基于深度学习提取特征的方法

## 两阶段物体检测网络

- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ Faster R-CNN
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络

# 基于深度学习提取特征的方法

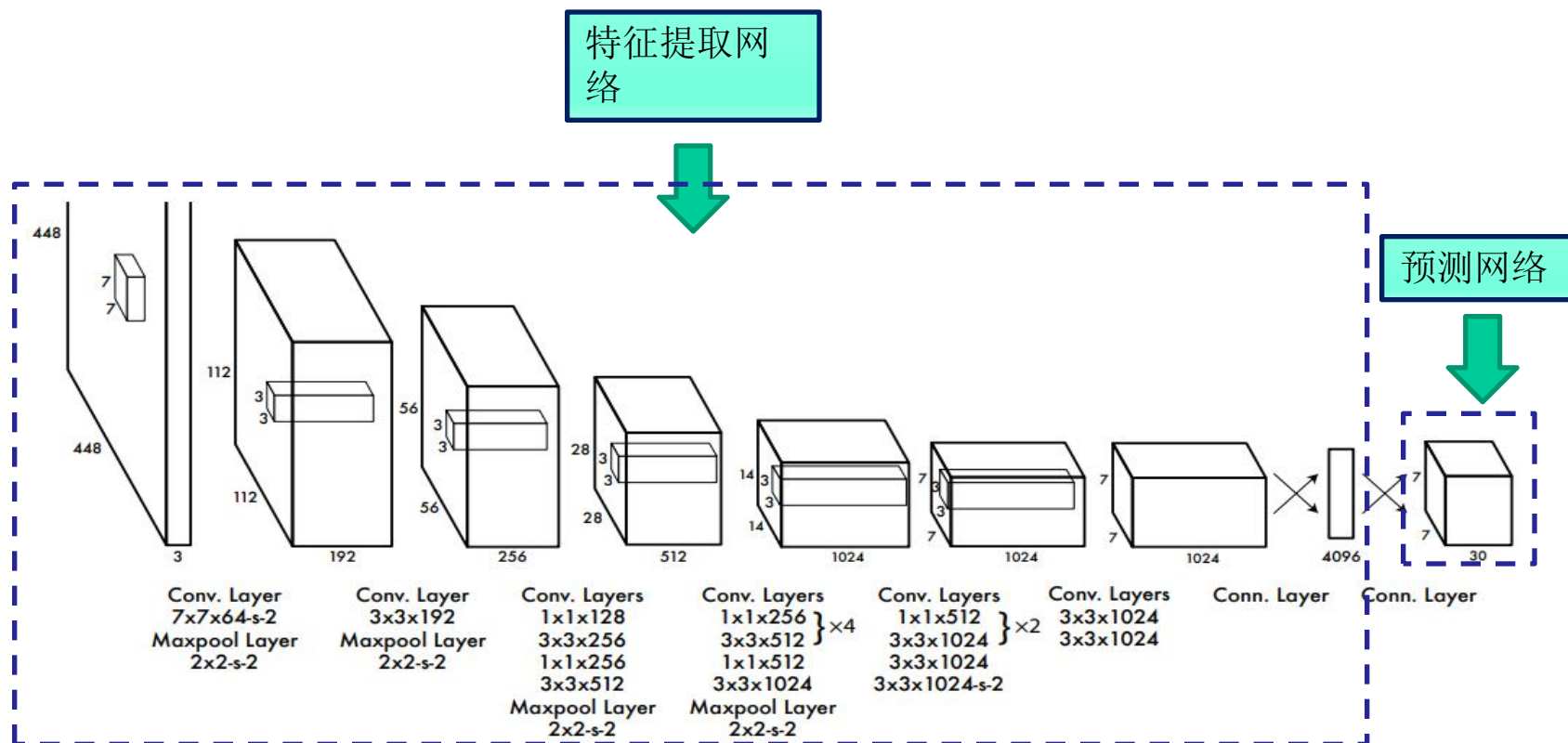
## YOLO

两阶段物体检测方法：

- 需要额外步骤的提取预选框，会增加时间复杂度，降低了测试时间，无法达到实时检测的目的。

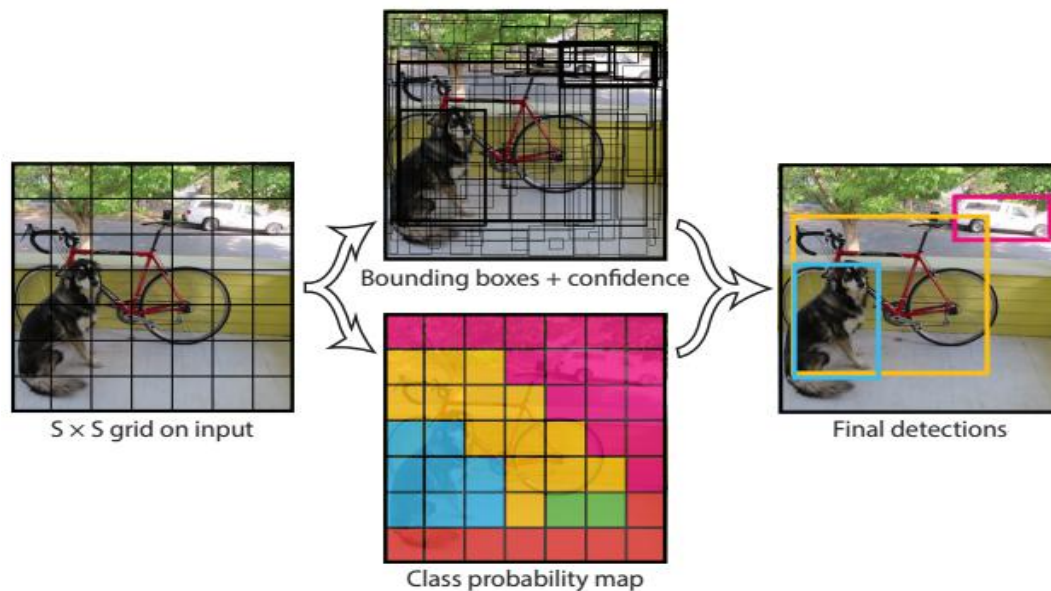
YOLO优势：

- YOLO训练速度快；
- 背景误检率低；
- YOLO的泛化能力更强，具有更好的通用性。



# 基于深度学习提取特征的方法

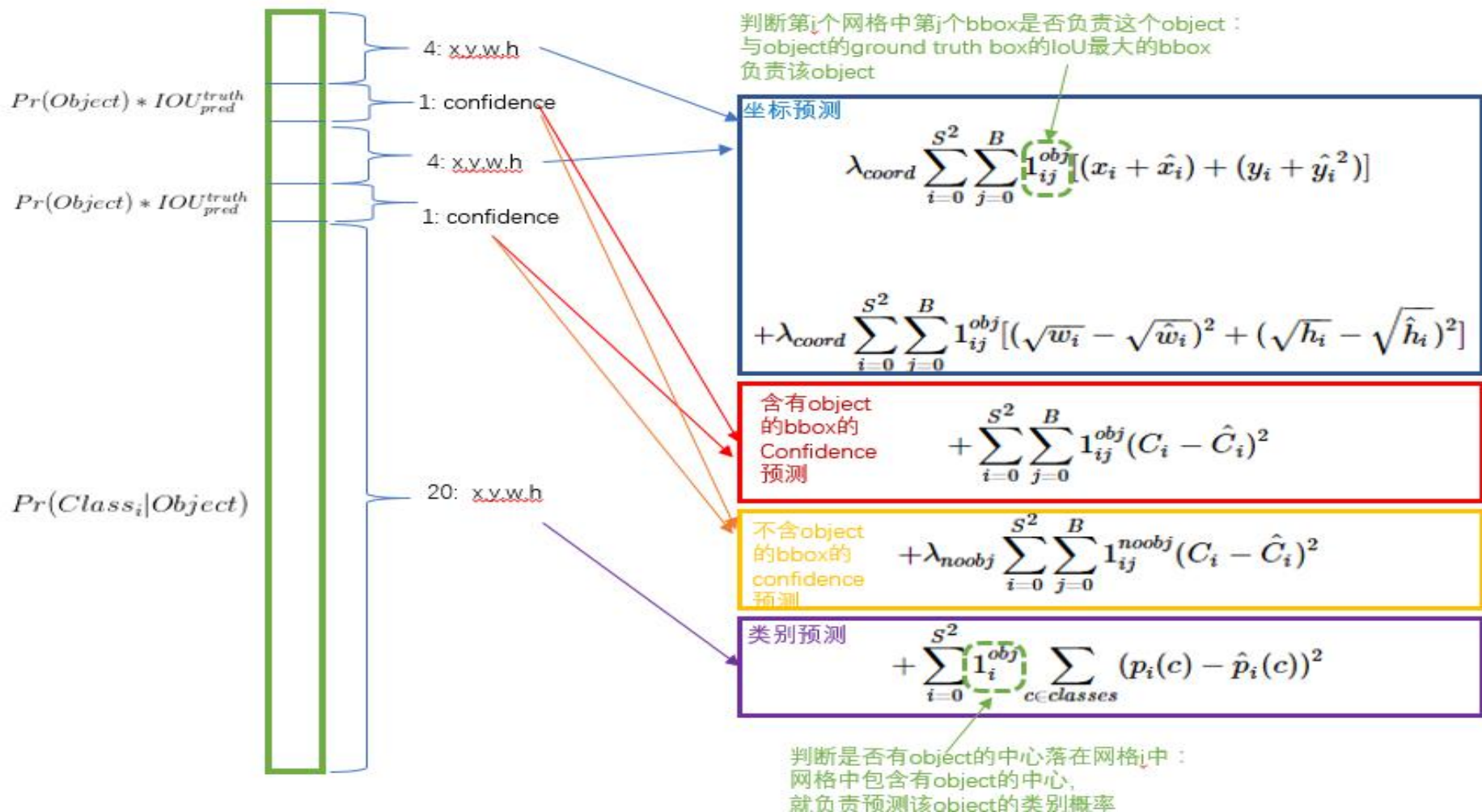
## YOLO





# 基于深度学习提取特征的方法

## YOLO



# 基于深度学习提取特征的方法

## YOLO

YOLO训练主要包括两步:

- 第一步使用ImageNet 数据集训练YOLO网络的前20个卷积层和一个全连接层，输入图像尺寸为 $224 \times 224$ 。
- 第二步使用第一步得到前20层卷积网络参数来初始化YOLO模型的前20个卷积层网络参数，然后用PASCAL VOC 数据集训练模型，此时输入图像尺寸为 $448 \times 448$ 。

# 基于深度学习提取特征的方法

## Beyond YOLO – YOLO v2

- 更好的检测准确度
- 更快的训练速度
- 更强的检测泛化能力

Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 7263-7271.

# 基于深度学习提取特征的方法

## YOLO v2

### ➤ 更好的检测准确度

- ◆ **批量正则化**：批量正则化可以加快网络的收敛速度，同时也消除了对其他正则化形式的依赖；
- ◆ **分辨率更高的预训练模型**：在YOLO v2中，首先使用分辨率为 $448 \times 448$ 的图像首先训练分类网络，然后在使用相同分辨率的图像训练检测网络，相同的输入分辨率可以使检测准确度提高4%；
- ◆ **锚点框**：YOLO v2摒弃了之前版本中的方法，而是参照两阶段目标检测框架中的锚点机制。使用k-平均聚类的方法得到每个点取锚点框的个数以及每个锚点框的面积及长宽比例；
- ◆ **多尺度训练**：YOLO v2使用多尺度输入图像的方法训练网络。在训练过程中，每个batch图像的分辨率不同，范围为： $\{320, 352, \dots, 608\}$ ；
- ◆ **使用更精细的特征预测**：使用多尺度特征进行预测，相比于只是用最后一层输出特征，准确度提高1%。

# 基于深度学习提取特征的方法

## YOLO v2

### ➤ 更快的训练速度

使用了一种新颖的网络Darknet-19处理目标检测问题。

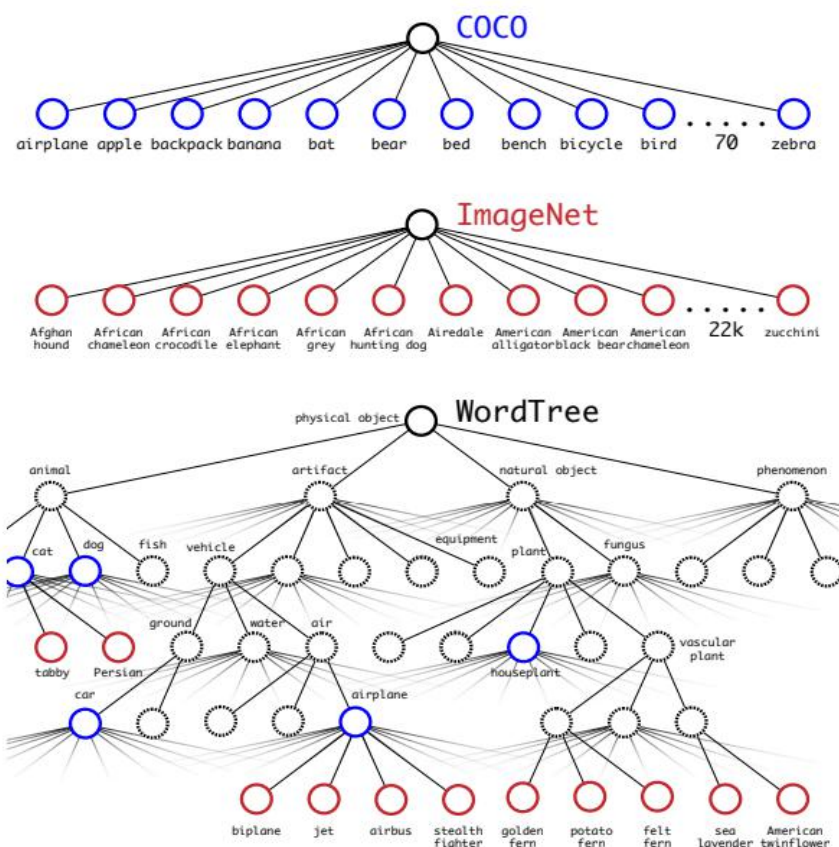
Type	Filters	Size/Stride	Output
Convolutional	32	$3 \times 3$	$224 \times 224$
Maxpool		$2 \times 2/2$	$112 \times 112$
Convolutional	64	$3 \times 3$	$112 \times 112$
Maxpool		$2 \times 2/2$	$56 \times 56$
Convolutional	128	$3 \times 3$	$56 \times 56$
Convolutional	64	$1 \times 1$	$56 \times 56$
Convolutional	128	$3 \times 3$	$56 \times 56$
Maxpool		$2 \times 2/2$	$28 \times 28$
Convolutional	256	$3 \times 3$	$28 \times 28$
Convolutional	128	$1 \times 1$	$28 \times 28$
Convolutional	256	$3 \times 3$	$28 \times 28$
Maxpool		$2 \times 2/2$	$14 \times 14$
Convolutional	512	$3 \times 3$	$14 \times 14$
Convolutional	256	$1 \times 1$	$14 \times 14$
Convolutional	512	$3 \times 3$	$14 \times 14$
Convolutional	256	$1 \times 1$	$14 \times 14$
Convolutional	512	$3 \times 3$	$14 \times 14$
Maxpool		$2 \times 2/2$	$7 \times 7$
Convolutional	1024	$3 \times 3$	$7 \times 7$
Convolutional	512	$1 \times 1$	$7 \times 7$
Convolutional	1024	$3 \times 3$	$7 \times 7$
Convolutional	512	$1 \times 1$	$7 \times 7$
Convolutional	1024	$3 \times 3$	$7 \times 7$
Convolutional	1000	$1 \times 1$	$7 \times 7$
Avgpool		Global	1000
Softmax			

# 基于深度学习提取特征的方法

## YOLO v2

### ➤ 更强的检测泛化能力

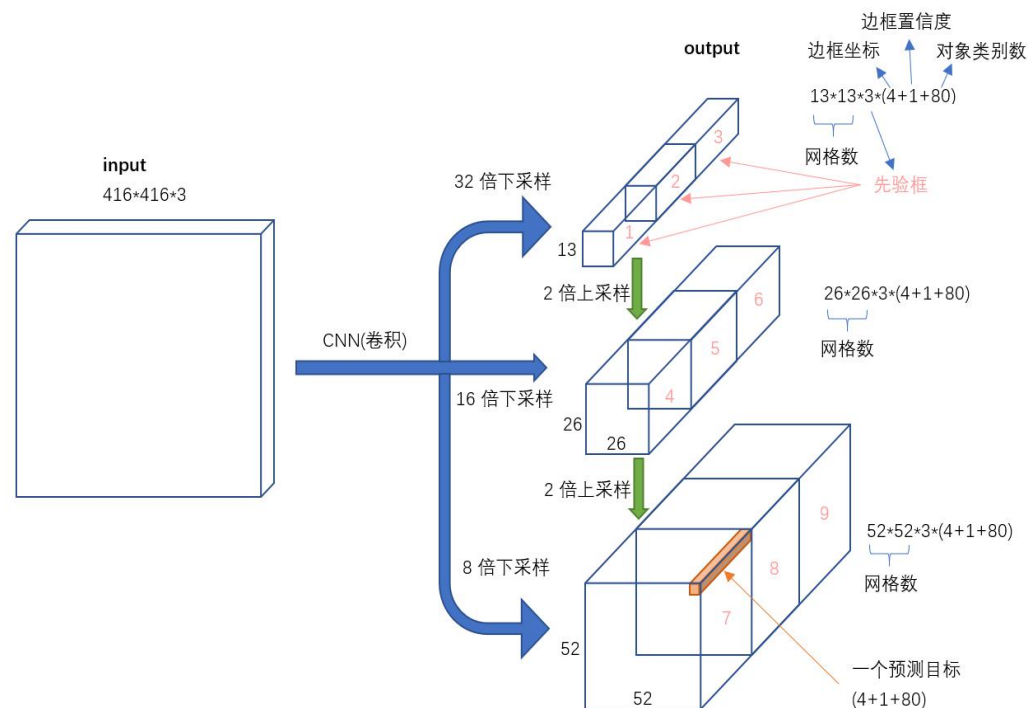
使用一种将分类数据集与检测数据集联合训练的方法。使用检测数据集学习定位物体以及对一些通用种类分类的能力。通过使用分类数据集扩展器可以检测到的种类。



# 基于深度学习提取特征的方法

## Beyond YOLO – YOLO v3

- 1) 使用了多尺度预测方法;
- 2) 改进了特征提取网络, 使用Darknet-53;
- 3) 改进分类损失, 将Softmax 损失替换为二值交叉熵损失





# 基于深度学习提取特征的方法

## 两阶段物体检测网络

- 典型架构
  - ✓ R-CNN
  - ✓ SPPNet
  - ✓ Fast R-CNN
  - ✓ Faster R-CNN
  - ✓ Feature Pyramid Network
  - ✓ Mask R-CNN

## 一阶段物体检测网络

- 典型架构
  - ✓ YOLO及其改进网络
  - ✓ SSD及其改进网络



# 基于深度学习提取特征的方法

## From YOLO to SSD

### YOLO 缺陷:

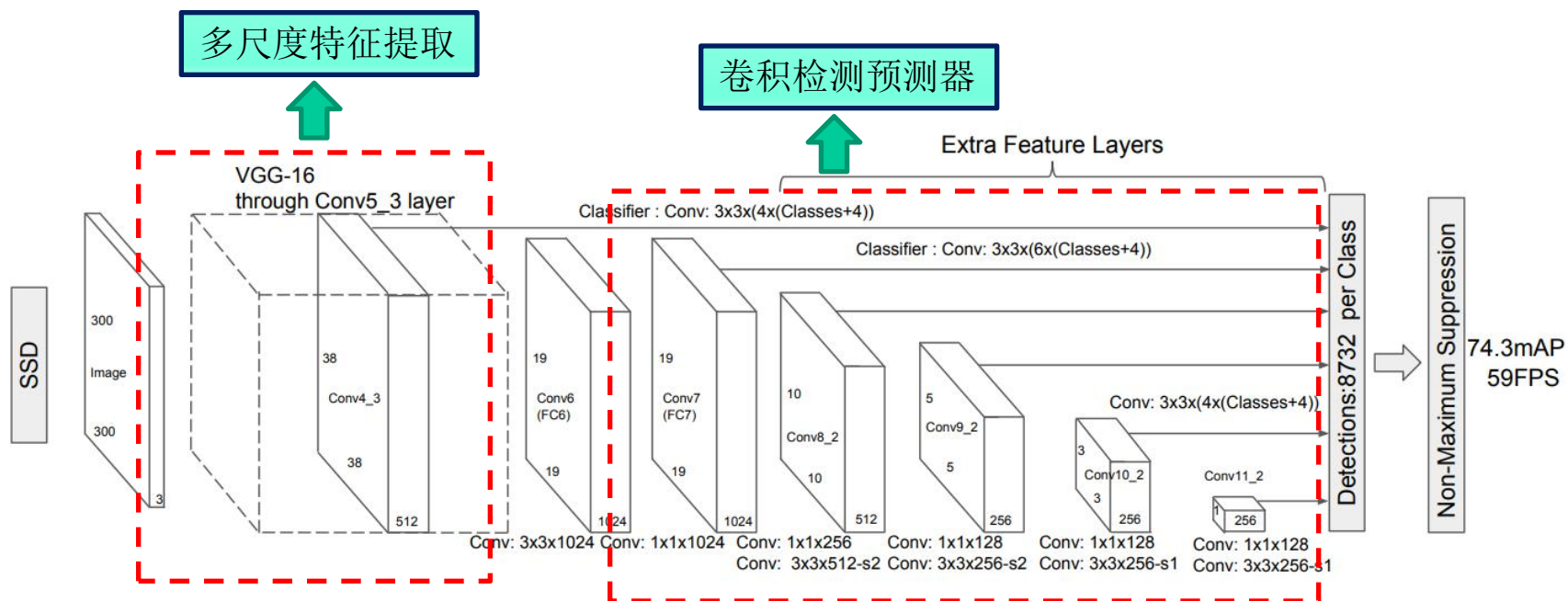
- 每个点只取两个框，不能包含图片中的所有物体；
- 只使用网络的最后一层输出特征进行检测，特征空间分辨率低，影响检测结果；
- 虽然速度快，但是精度很低，实际应用价值不大。

### SSD 优点:

- 保证测试速度的前提下，提高了检测准确度；
- 使用多尺度特征进行预测，进一步提高检测准确度；
- 摆脱对输入图片尺寸的限制，即使在输入图片分辨率很小时，也能达到很高的准确度。

# 基于深度学习提取特征的方法

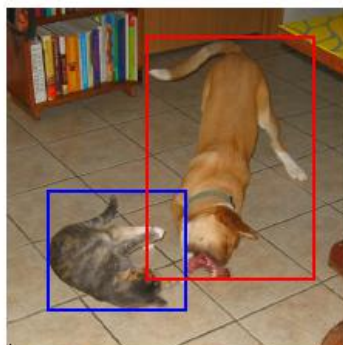
## SSD



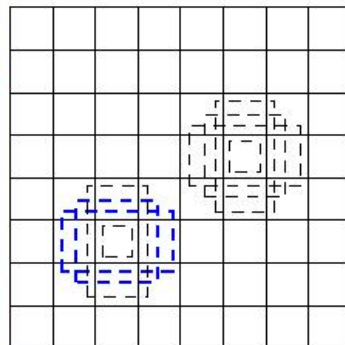
# 基于深度学习提取特征的方法

## SSD

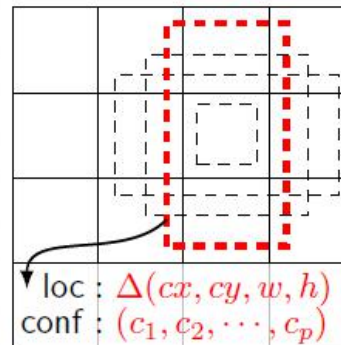
### Default Box



(a) Image with GT boxes



(b)  $8 \times 8$  feature map



(c)  $4 \times 4$  feature map

$$s_k = s_{\min} + \frac{s_{\max} - s_{\min}}{m - 1}(k - 1), \quad k \in [1, m]$$

$$\alpha_r = \left\{1, 2, 3, \frac{1}{2}, \frac{1}{3}\right\} \begin{cases} w_k^a = s_k \sqrt{a_r} \\ h_k^a = s_k / \sqrt{a_r} \end{cases}$$

# 基于深度学习提取特征的方法

## SSD

分类及回归

$$L(x, c, l, g) = \frac{1}{N} (L_{conf}(x, c) + \alpha L_{loc}(x, l, g))$$

$$L_{loc}(x, l, g) = \sum_{i \in Pos}^N \sum_{m \in \{cx, cy, w, h\}} x_{ij}^k \text{smooth}_{L1}(l_i^m - \hat{g}_j^m)$$

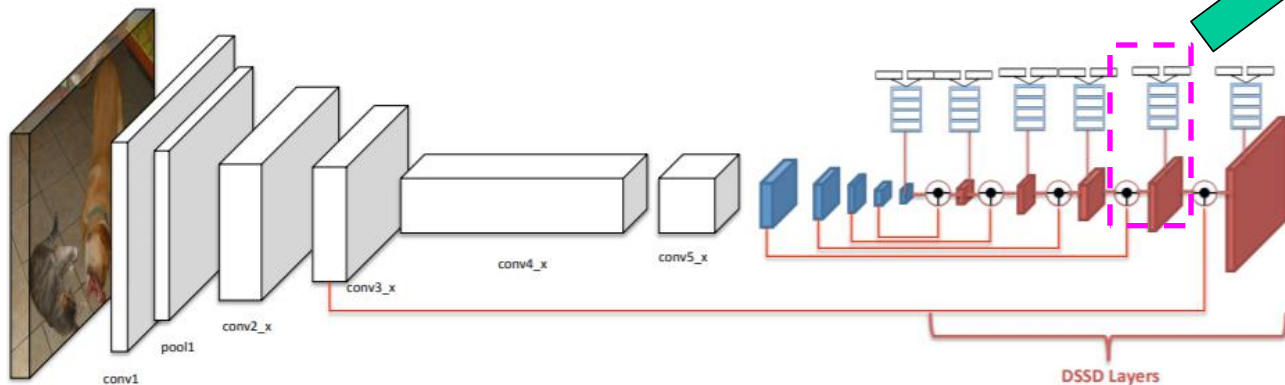
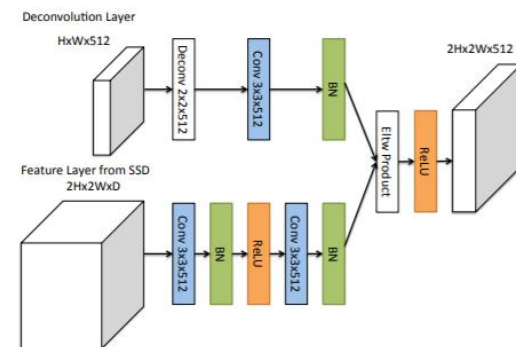
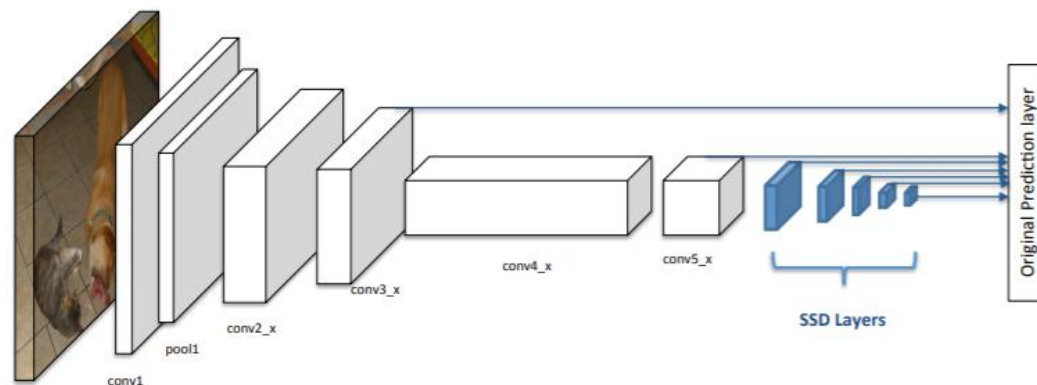
$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^w \quad \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^h$$

$$\hat{g}_j^w = \log\left(\frac{g_j^w}{d_i^w}\right) \quad \hat{g}_j^h = \log\left(\frac{g_j^h}{d_i^h}\right)$$

$$L_{conf}(x, c) = - \sum_{i \in Pos}^N x_{ij}^p \log(\hat{c}_i^p) - \sum_{i \in Neg} \log(\hat{c}_i^0) \quad \text{where} \quad \hat{c}_i^p = \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)}$$

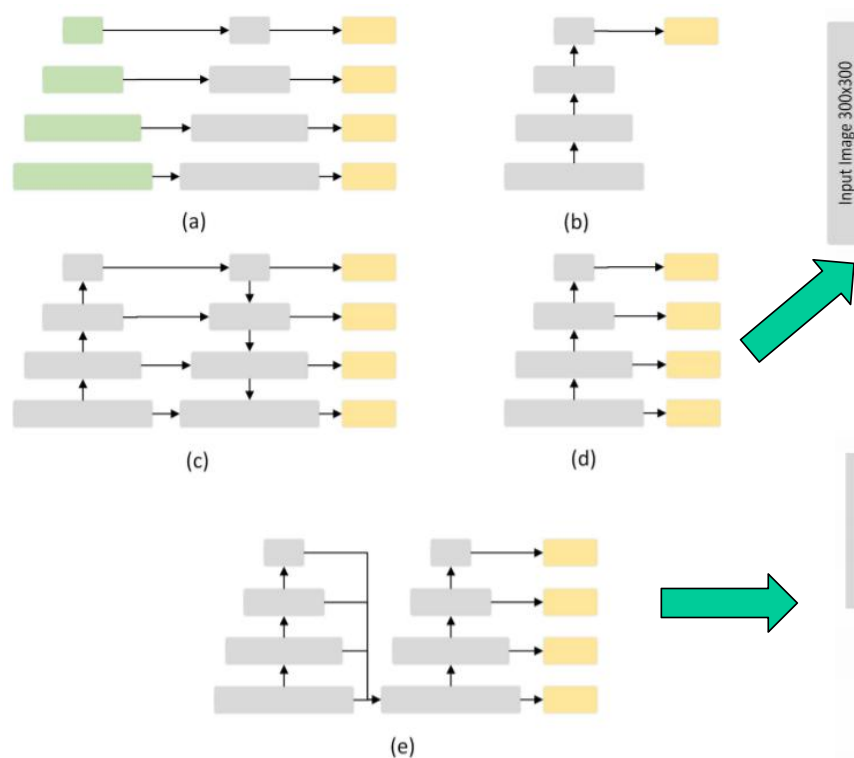
# 基于深度学习提取特征的方法

## Beyond SSD -- DSSD

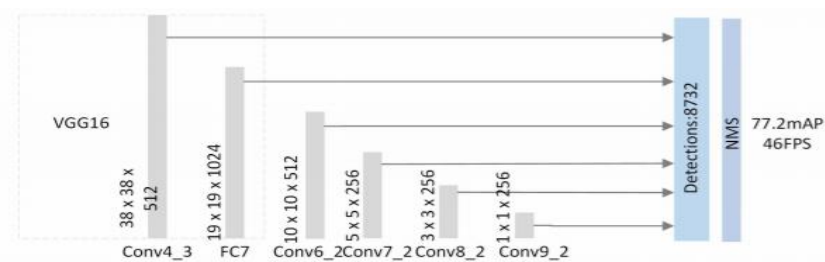


# 基于深度学习提取特征的方法

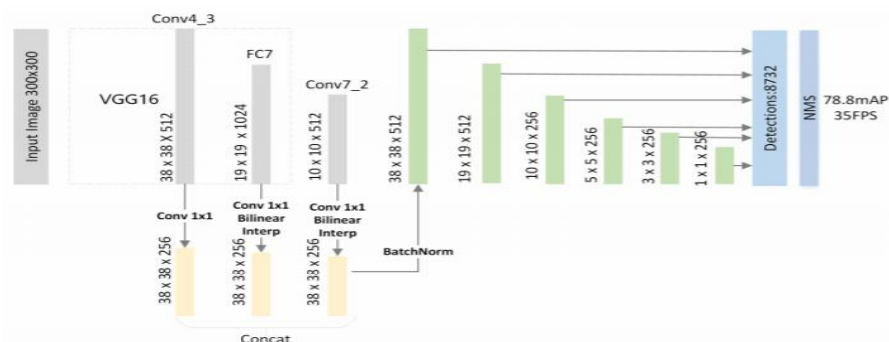
## Beyond SSD -- FSSD



SSD:

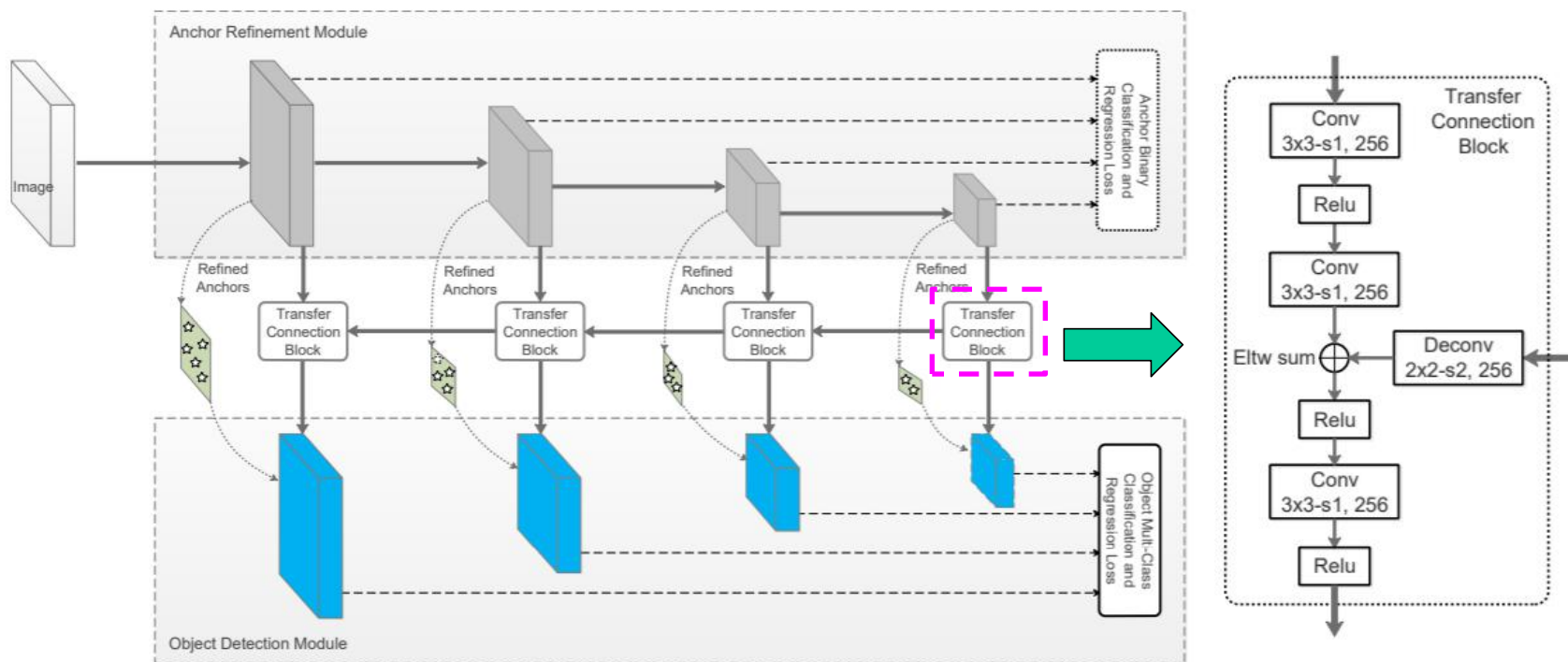


FSSD:



# 基于深度学习提取特征的方法

## Beyond SSD -- RefineNet



Zhang S, Wen L, Bian X, et al. Single-shot refinement neural network for object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 4203-4212.

# 基于深度学习提取特征的方法

## 物体检测一些新的思路

### ➤ Cascade结构

- ✓ Cascade R-CNN
- ✓ Cascade Mask R-CNN

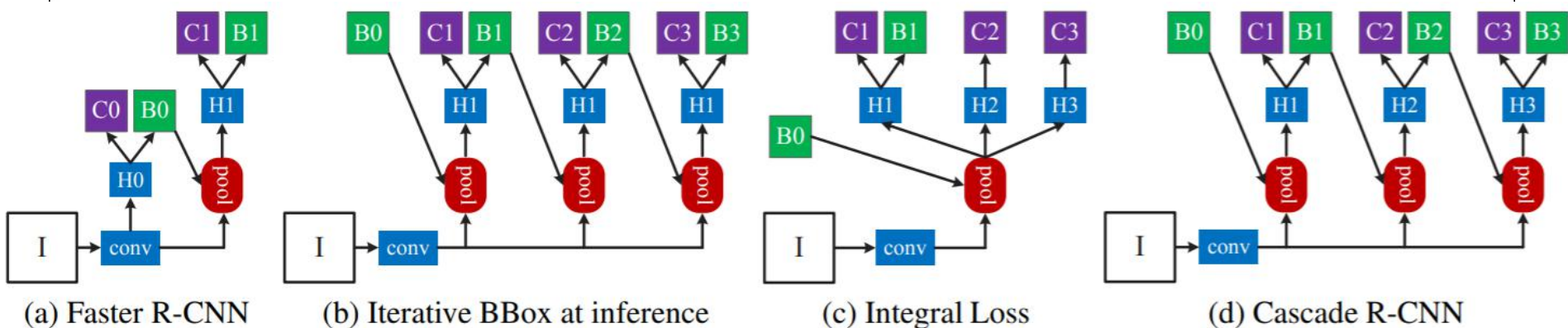
### ➤ Anchor-Free方法

- ✓ FCOS
- ✓ FoveaBox



# 基于深度学习提取特征的方法

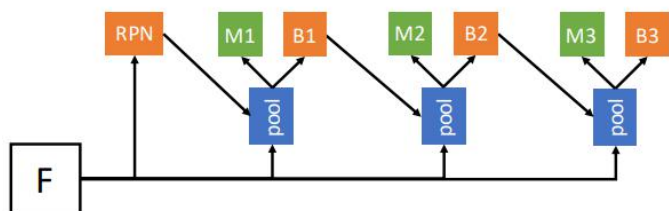
## Cascade R-CNN



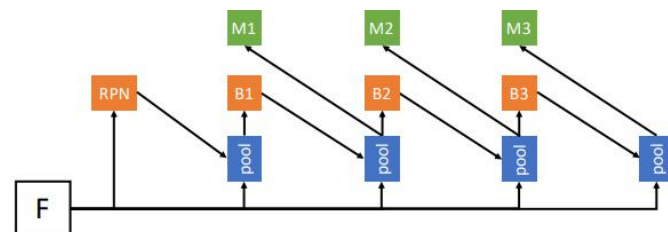
Cai Z, Vasconcelos N. Cascade r-cnn: Delving into high quality object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 6154-6162.

# 基于深度学习提取特征的方法

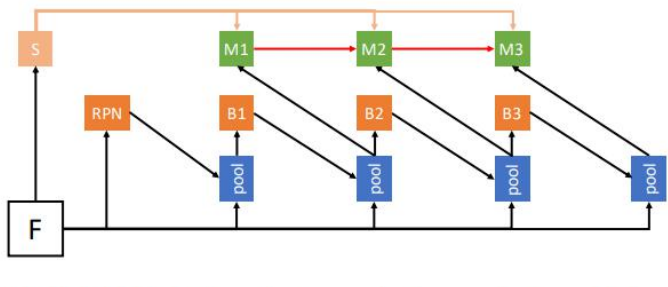
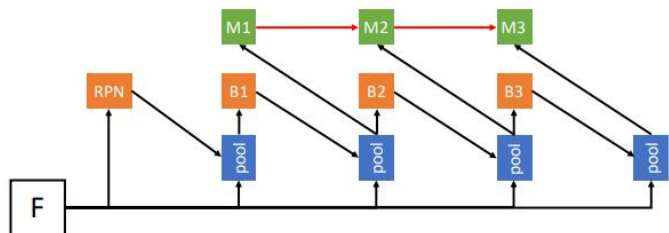
## Cascade Mask R-CNN



(a) Cascade Mask R-CNN



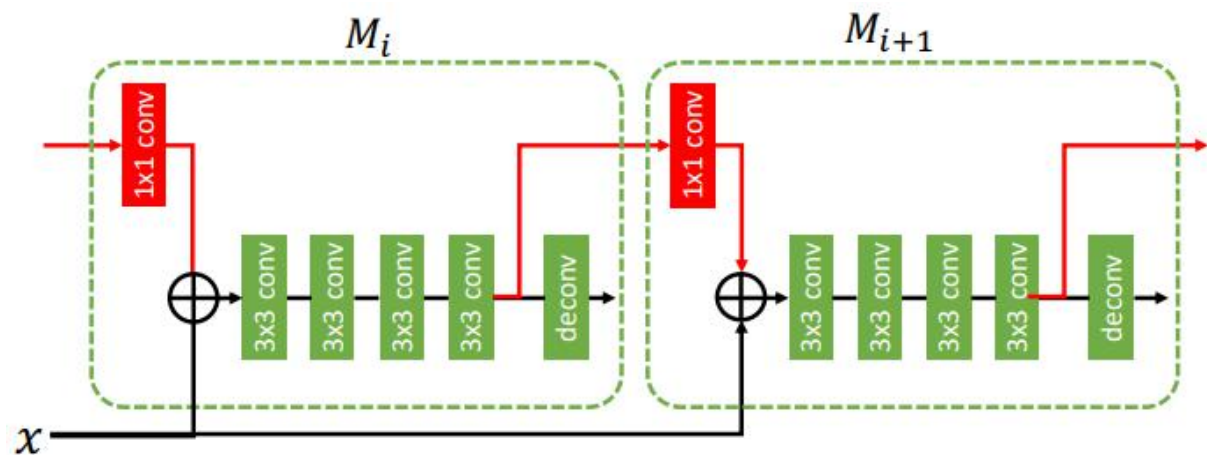
(b) Interleaved execution



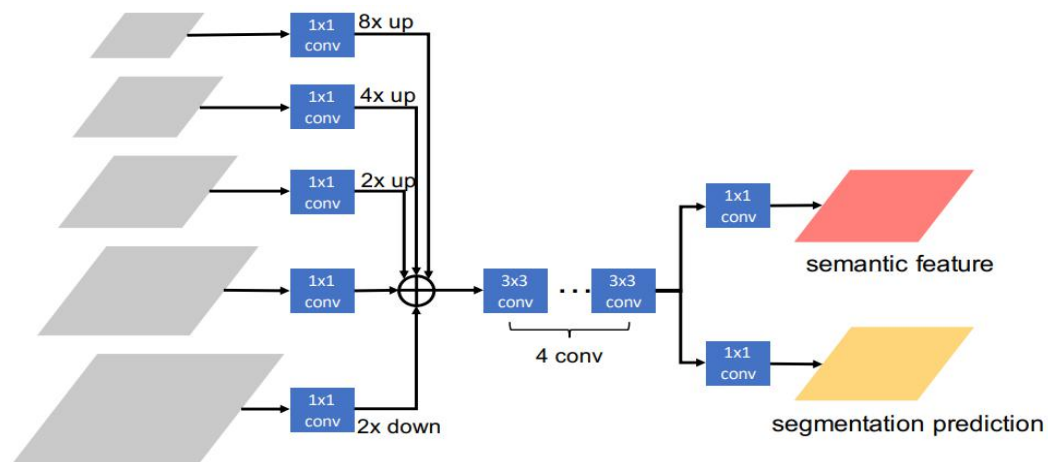
# 基于深度学习提取特征的方法

## Cascade Mask R-CNN

Cascade结构



语义辅助信息



# 基于深度学习提取特征的方法

## 目标检测一些新的思路

### ➤ Cascade结构

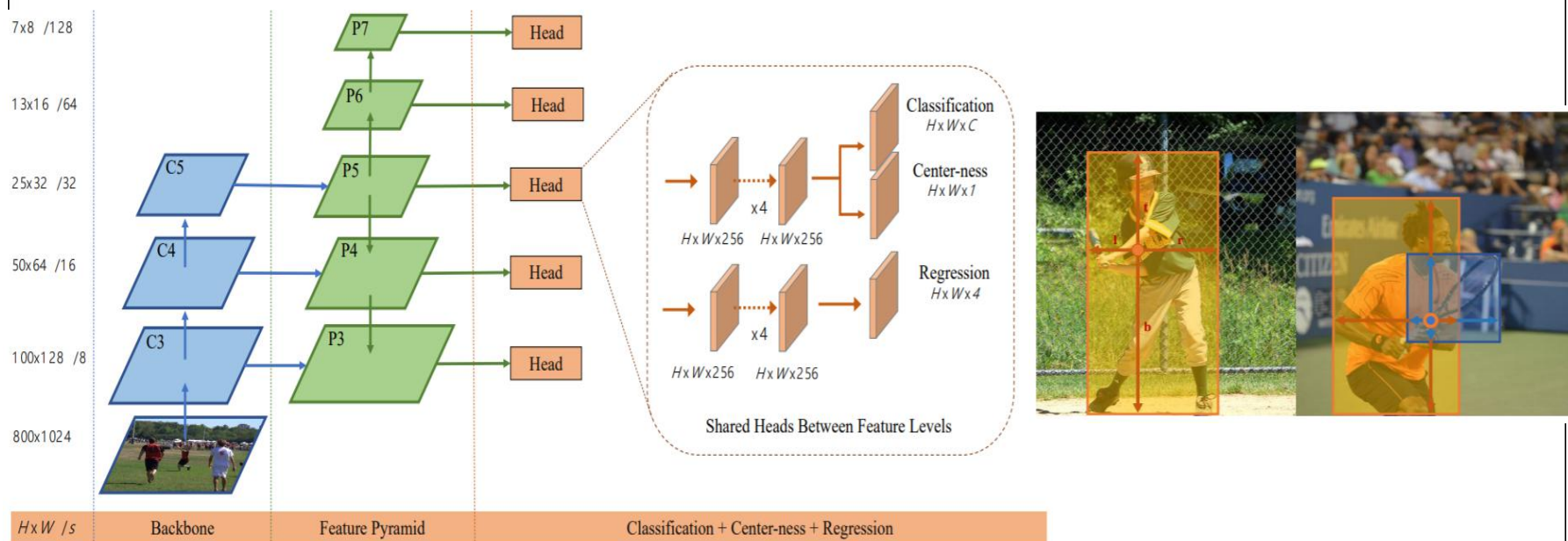
- ✓ Cascade R-CNN
- ✓ Cascade Mask R-CNN

### ➤ Anchor-Free方法

- ✓ FCOS
- ✓ FoveaBox

# 基于深度学习提取特征的方法

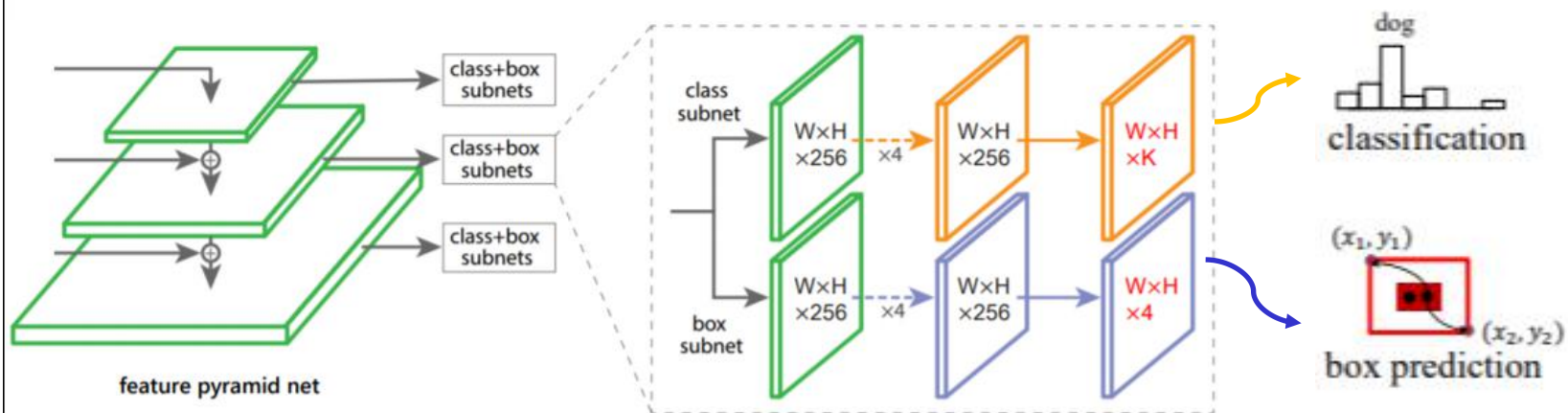
## FCOS



Tian Z, Shen C, Chen H, et al. FCOS: Fully Convolutional One-Stage Object Detection[J]. arXiv preprint arXiv:1904.01355, 2019.

# 基于深度学习提取特征的方法

## FoveaBox



谢谢！