http://dmirlab.com

Four basic tasks in computer vision

Speaker : 章浩

Time : 2018.11.18

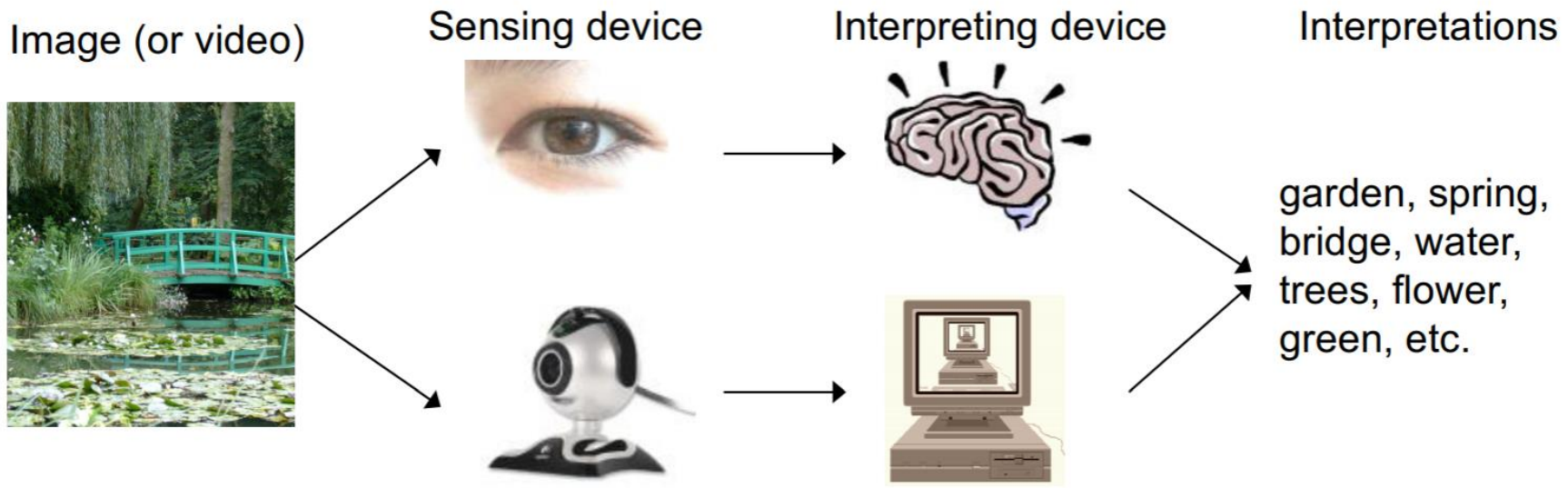
Outline

- Brief to CV
- Classification
- Localisation
- Object Detection
- Segmentation

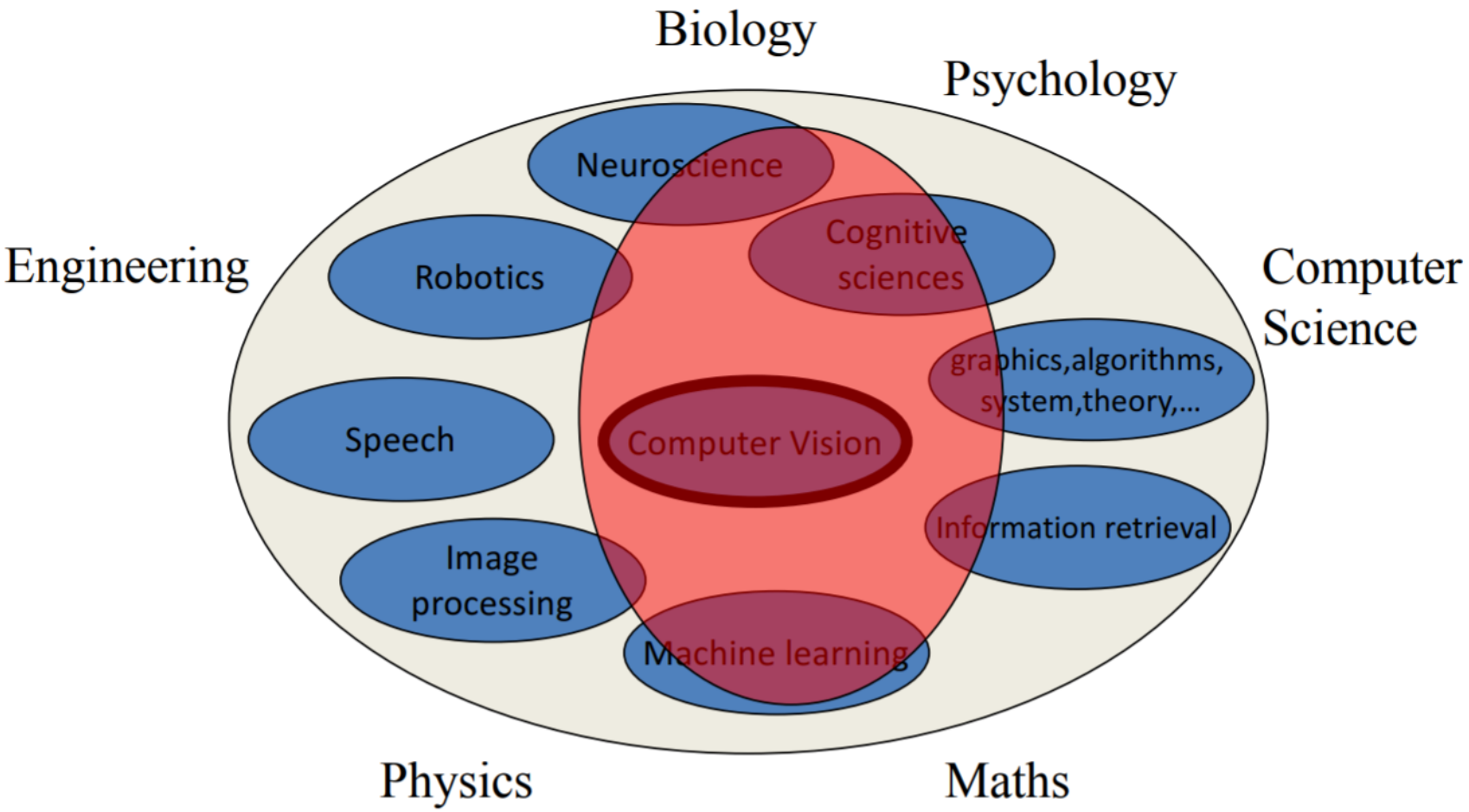
Outline

- Brief to CV
- Classification
- Localisation
- Object Detection
- Segmentation

What is Computer Vison



What is it related to



The goal of computer vison

To bridge the gap between pixels and “meaning”

What we see

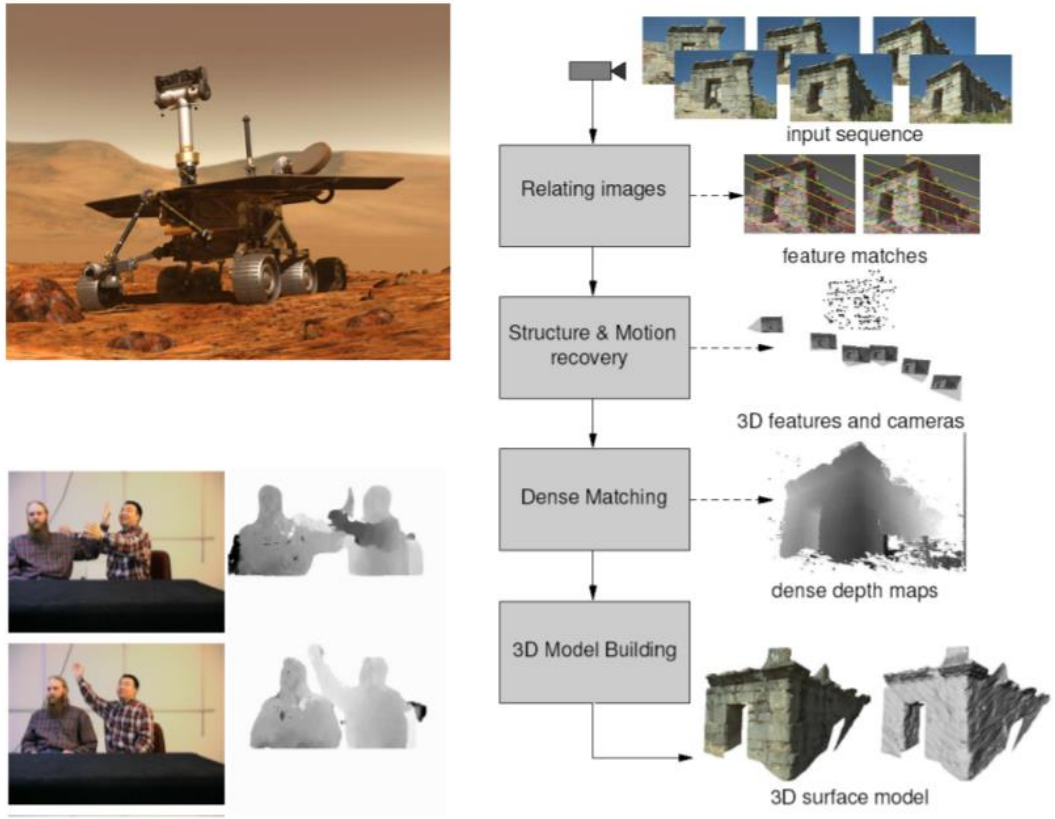
0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

What a computer sees

What kind of information can we extract from an image?

- Metric 3D information
- Semantic information

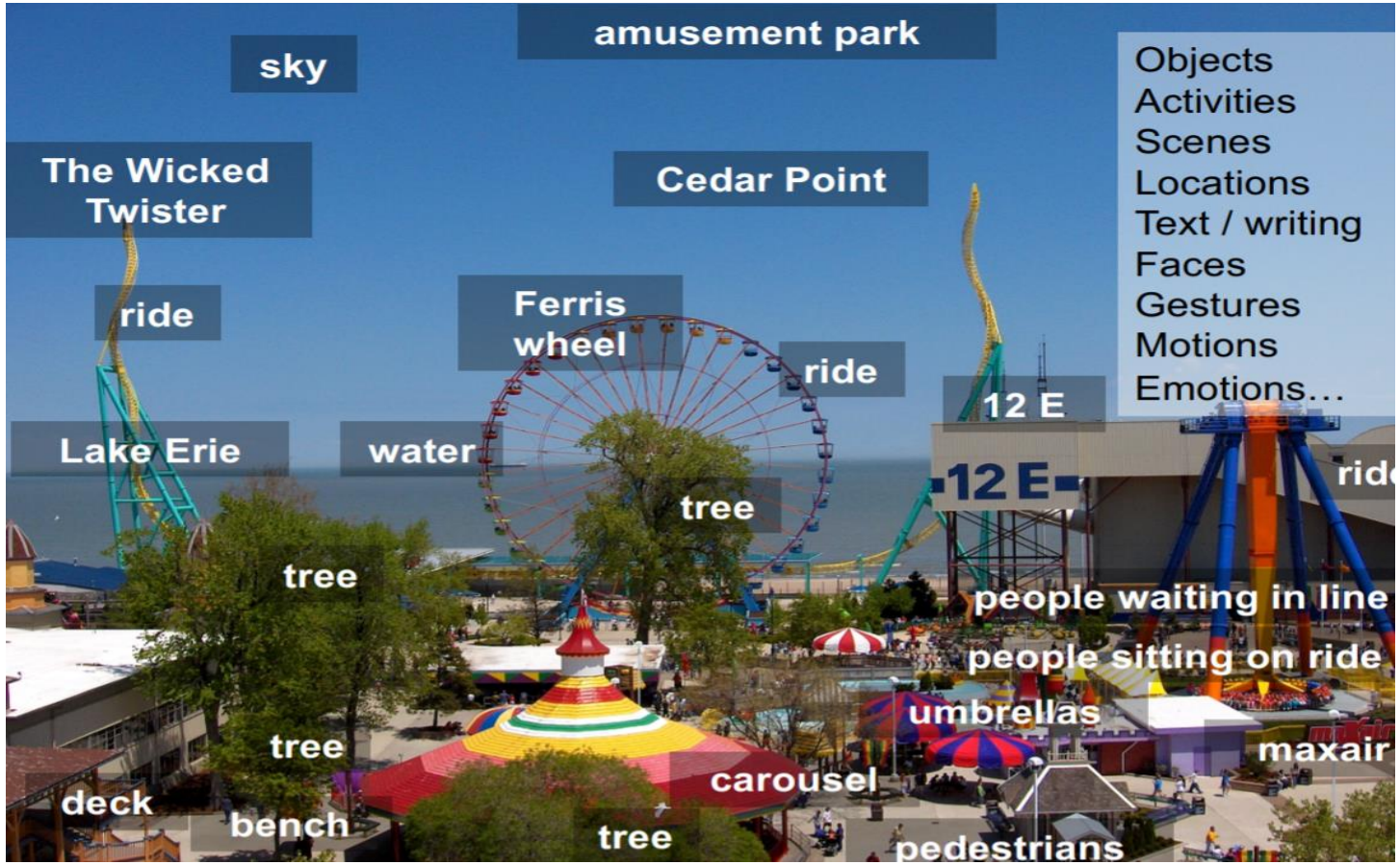
Vision as measurement device



Pollefeys et al.

Goesele et al.

Vision as a source of semantic information



Four basic tasks

- Classification
- Localisation
- Object Detection
- Segmentation

Outline

- Brief to CV
- Classification
- Localisation
- Object Detection
- Segmentation

Task description

- Definition:
- Challenges:
- Solutions:

Task description

- Definition:

(assume given set of discrete labels)
{dog, cat, truck, plane, ...}

→ cat

- **固定**感兴趣的category集合，输入一张图片，预测它的catergory，数学模型为：**multi**_class classification

Task description

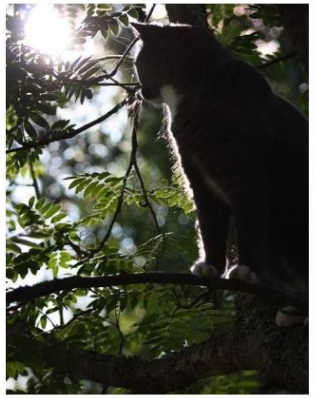
- Challenge:
  - view point variation(不同视角)
  - Illumination (光线干扰)
  - occlusion (前景遮挡)
  - background clutter (背景混合)
  - scale (一图多尺寸)
  - deformation (同一个obj有各种形变)

Task description

- Challenge: **View point variation**

Task description

- Challenge: **Illumination**



Task description

- Challenge: **Occlusion**

This image is CC0 1.0 public domain

This image is CC0 1.0 public domain

This image by jonsson is licensed under CC-BY 2.0

Task description

- Challenge: **Background Clutter**

This image is CC0 1.0 public domain

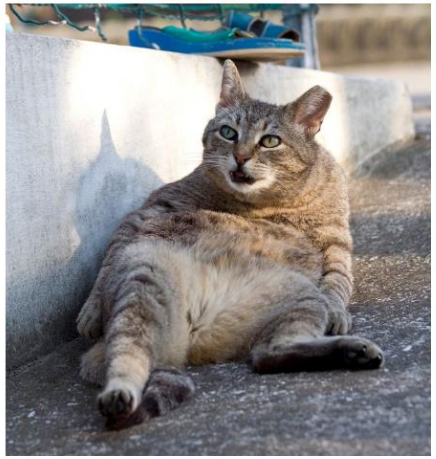
This image is CC0 1.0 public domain

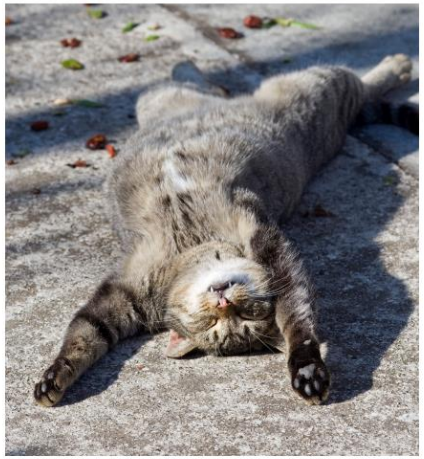
Task description

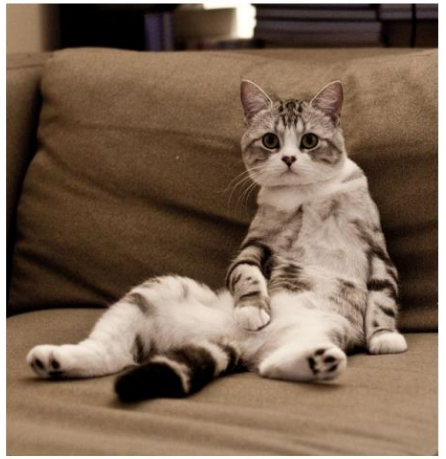
- Challenge: **Scale**

Task description

- Challenge: **Deformation**







Task description

- Solutions:

考虑到上述challenge，classifier需要满足若干**物理不变性**，即**invariant to**

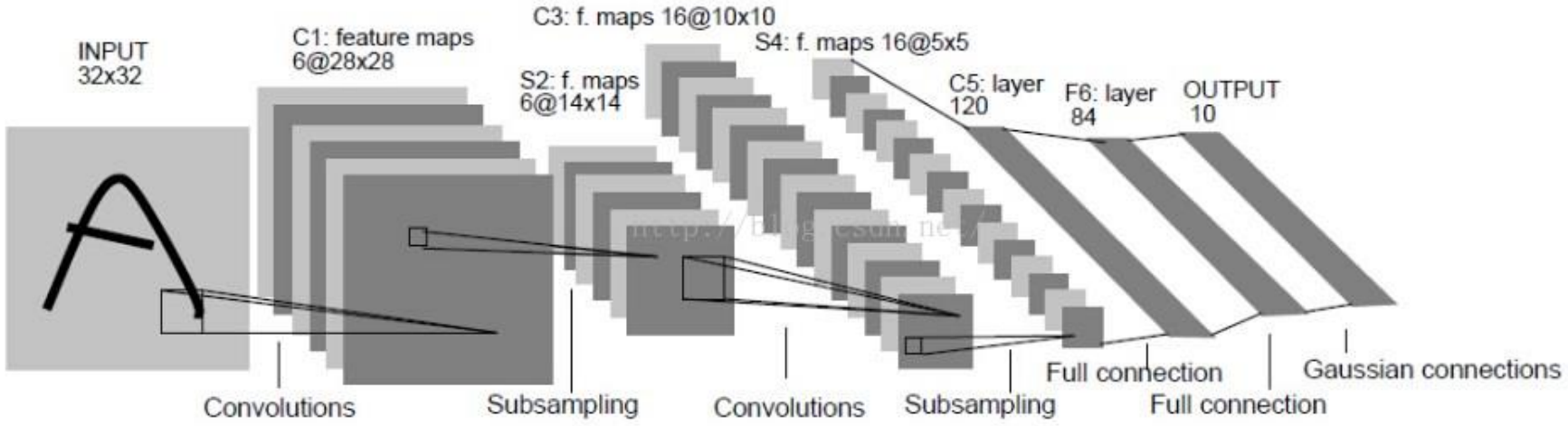
- rotation(旋转)
- translation(空间平移)
- scaling(尺寸放缩)
- lighting(光照)
- occlusion(前景遮挡)
- background clutter(后景与obj融合)
- deformation(形变)

CNN for image Classification

1. CNN的网络结构
   - 起源
   - 直观理解
2. 几个重要概念：
   - 卷积
   - 池化
   - 感受野
   - Train CNN

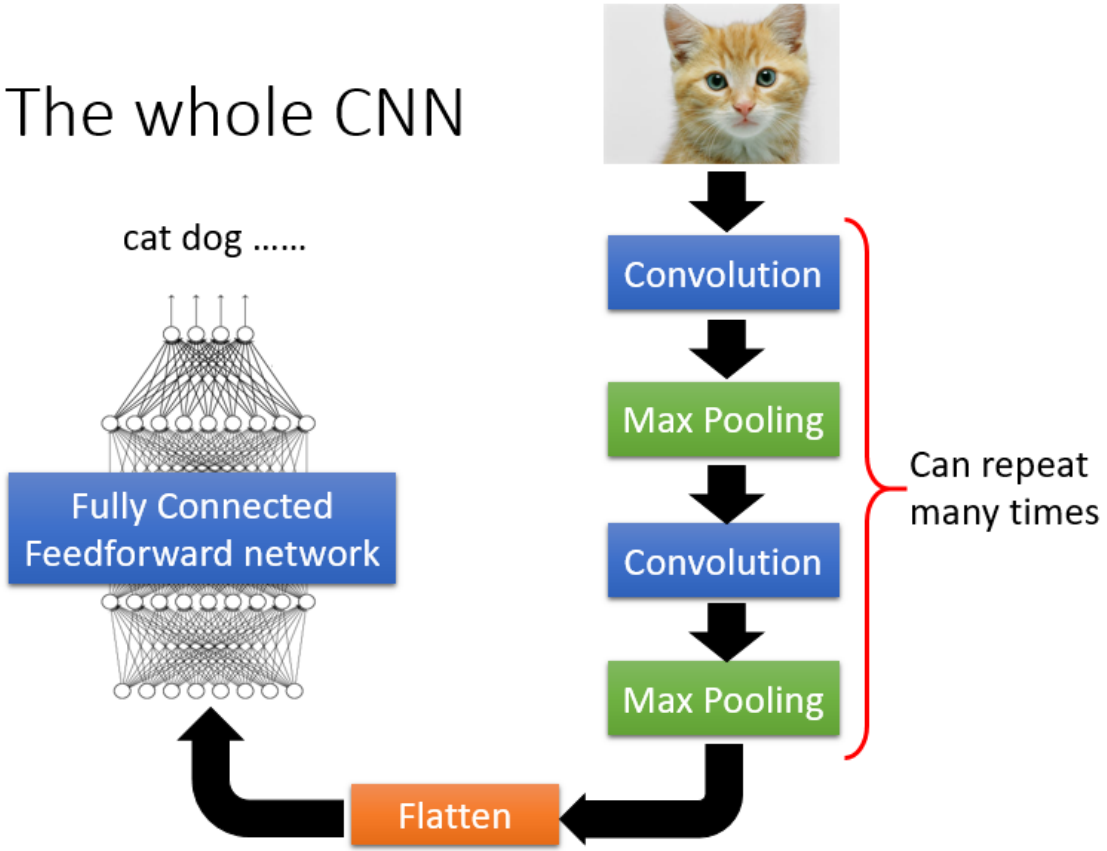
CNN for image Classification

- 起源: LeNet5 by Lecun, 1989



CNN for image Classification

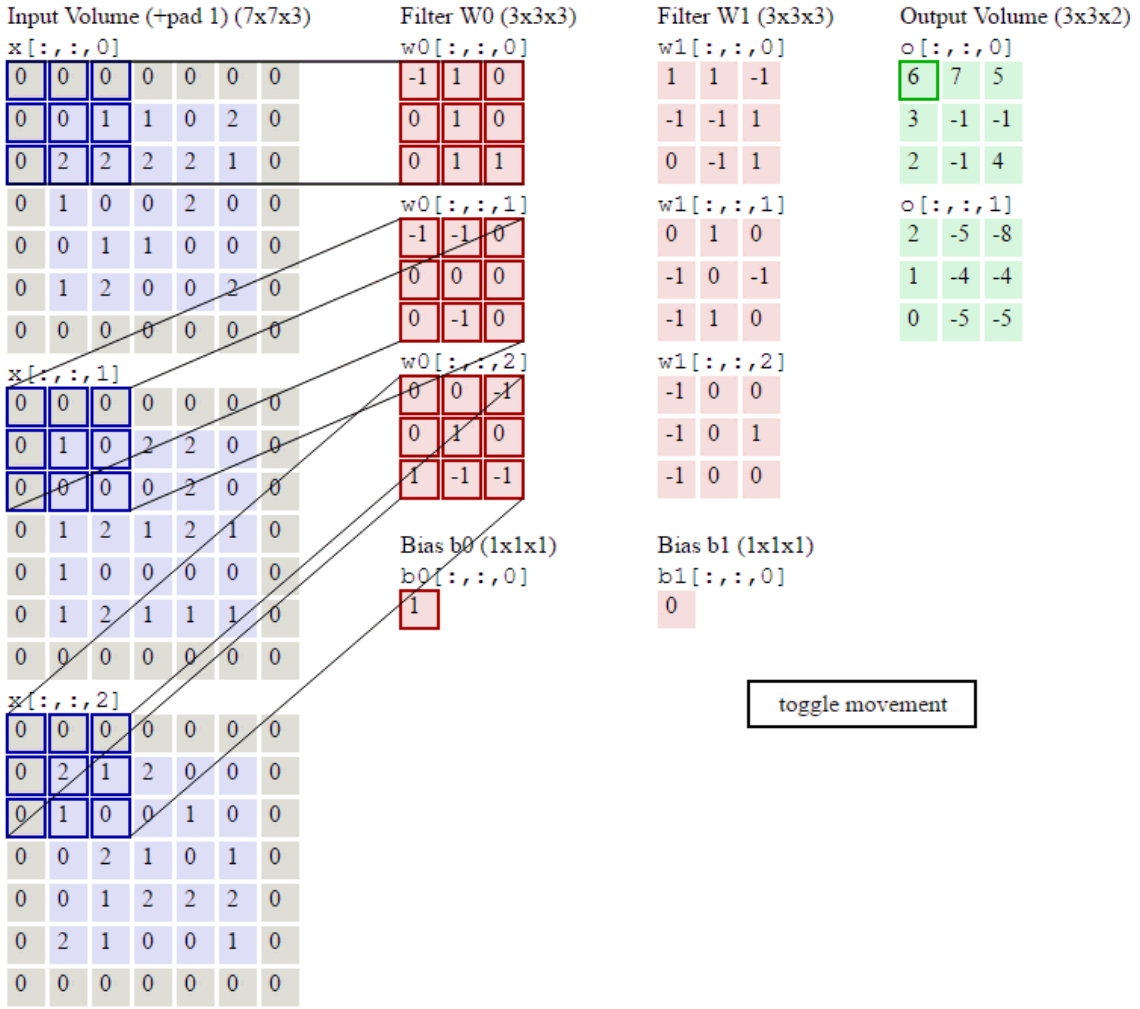
- 直观理解



CNN for image Classification

- 卷积：
  - **计算**：卷积等价于**向量点积**，过程如下：

CNN for image Classification

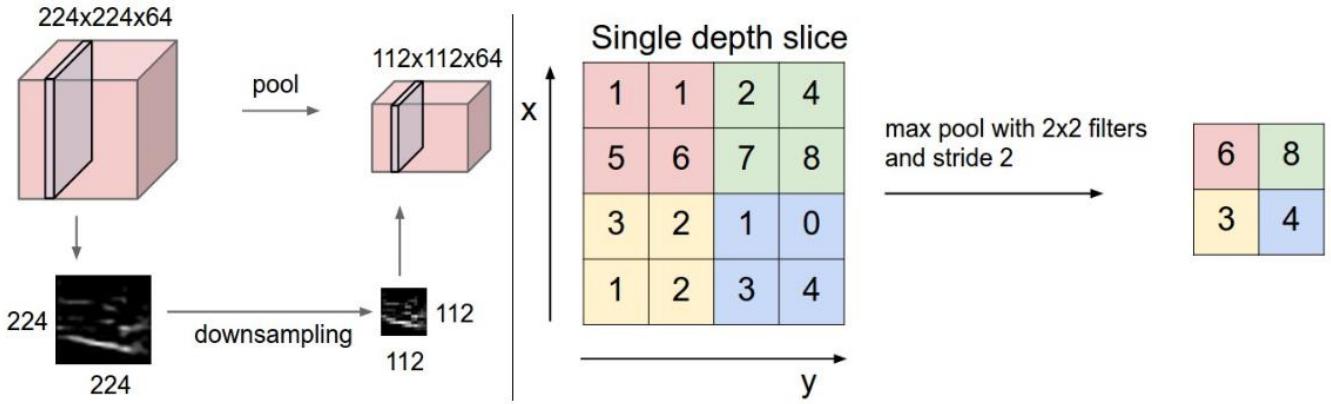


CNN for image Classification

- 卷积(Convolution)：
  1. **Role of convolution**：从**template matching**的观点出发，能与filter做卷积之后产生**最大输出**的那个feature与filter**最相似**。所以检测features的目标**转化**为寻找一组filters (filters又通过Loss来optimize)
  2. **Small filter**：**局部连接**(相比较FC)，更加关注相邻像素间的关系。在视觉识别中，关键性的图像特征、边缘、角点等只占据了整张图像的一小部分，所以局部连接不仅**符合像素空间的语义特点**，而且大大**减少了网络的参数量**
  3. **Shared filter**: 同一图片不同位置可以存在相同的特征，所以**权值共享**满足了图片中object的**空间平移不变形**

CNN for image Classification

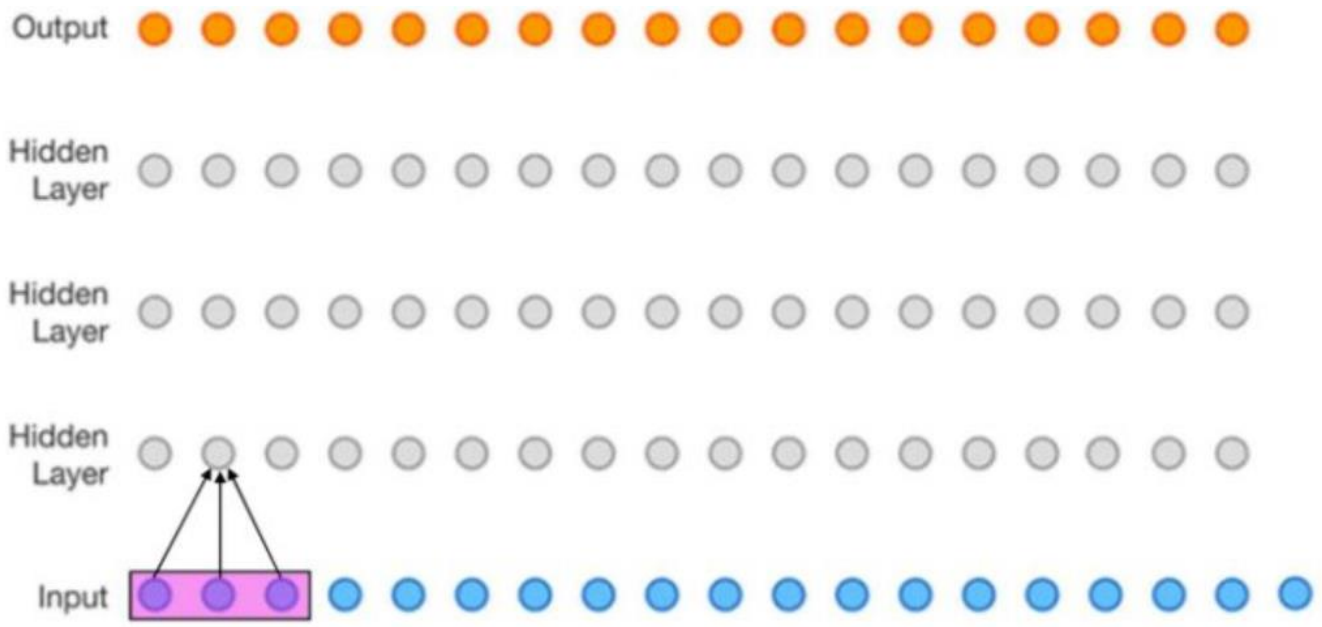
- 池化(Pooling):
  - **计算**：对卷积之后输出的一张output volume先分为若干patch，再对每个patch取max elem以代替整个patch，输出**缩小**后的output volume，如图:



1. **How to pool**: 从template matching这个viewpoint出发，正确的features应该是activation value最大者，故而一般选择**max pooling**
2. **Role of Pooling**: 不改变depth，**减小spatial resolution** & 一定程度满足**旋转不变形**，能容忍待匹配特征的细微偏移

CNN for image Classification

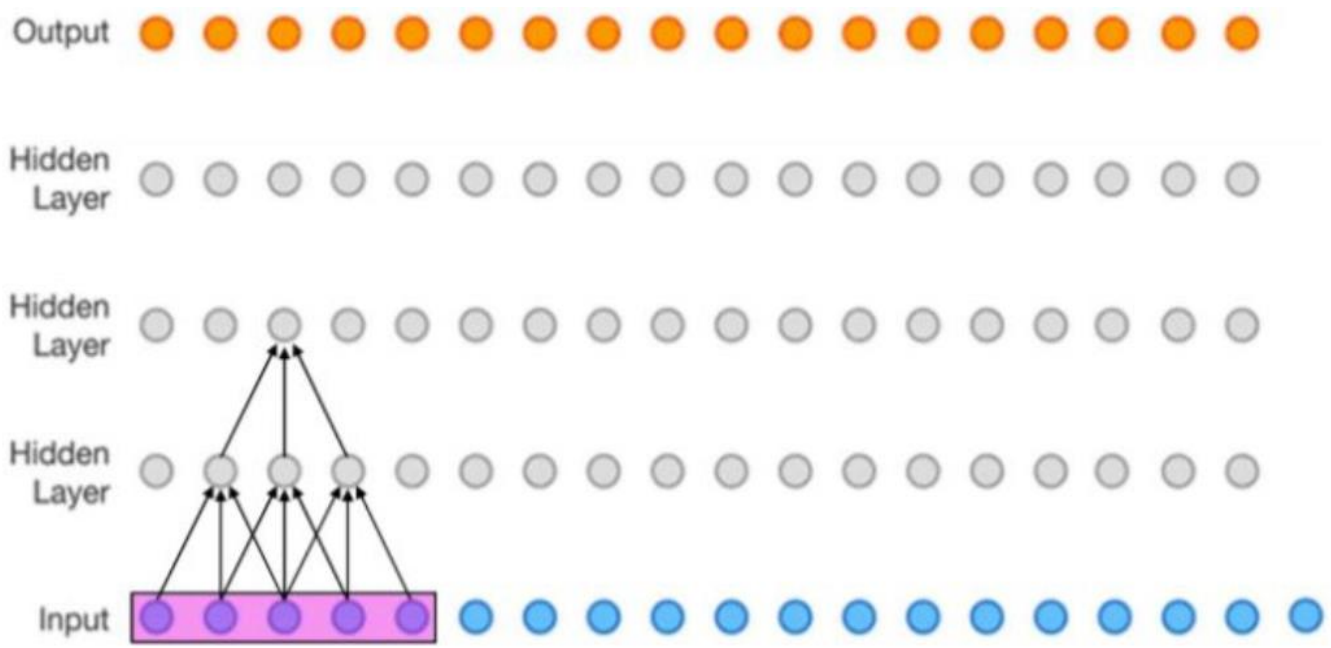
- Receptive Field(感受野)
  - **Key idea**: 上层特征能感受到输入图像的区域，描述了当前feature的power



第1层感受野[1]

CNN for image Classification

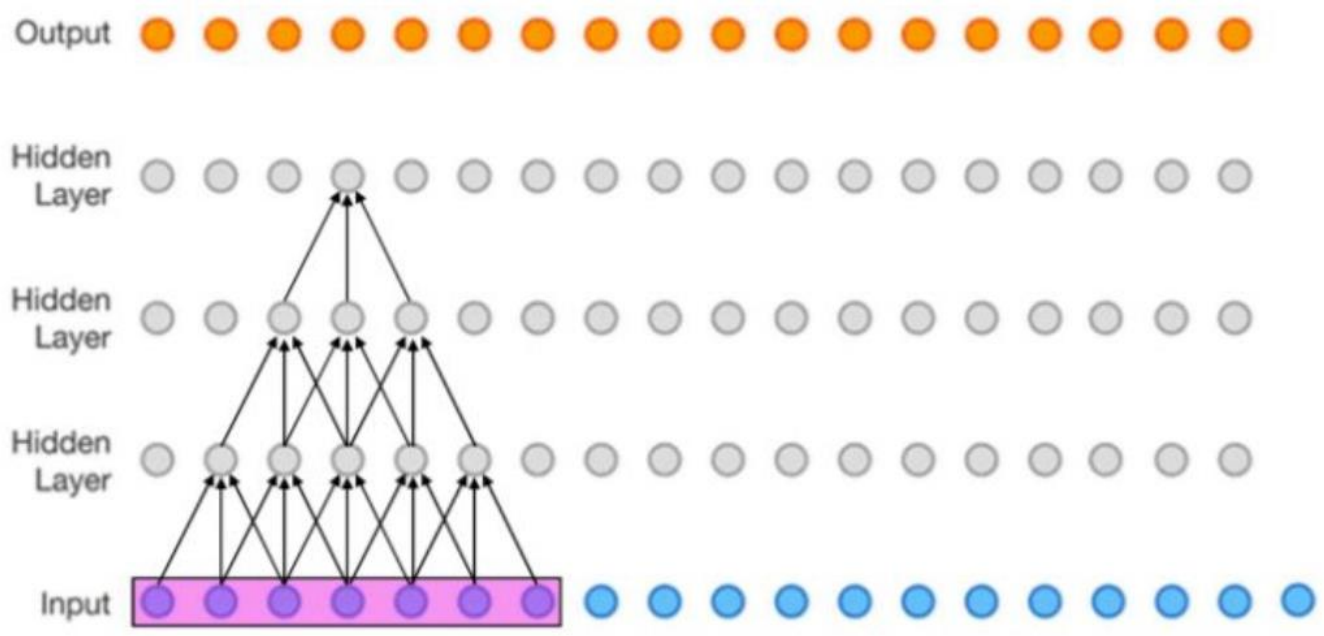
- Receptive Field(感受野)
  - **Key idea**: 上层特征能感受到输入图像的区域，描述了当前feature的power



第2层感受野[1]

CNN for image Classification

- Receptive Field(感受野)
  - **Key idea**: 上层特征能感受到输入图像的区域，描述了当前feature的power



第3层感受野[1]

CNN for image Classification

- Train CNN:
  - 参数初始化难题 (BN前靠调参，BN后不存在这个问题)
  - 梯度弥散和梯度爆炸的问题 (BN前靠小心选择各种激活函数，BN后这个问题基本被解决;注：BN指的是batch_normalization,2015,google)
  - 超参难以选择
  - 优化器难以选择
  - 学习率难以选择
  - 上述3个问题是DNN的顽疾，暂时没有“实际意义上包治百病”的大招，只有自己调参，又名炼丹。下面有一张来自cs231n-2017 & ML2017-LHY的一个丹方供参考:
    - **Optimization**: **Adam,** SGD+Momentum, Nesterov, RMSProp
    - **Regularization**: **Dropout**, Add noise then marginalize out
    - **Init Parameters: Xavier**
    - **Activation Function: ReLU,** Leakly ReLU, MaxOut
    - **Batch Normalization(CNN中非常有效)**

Meilstones of CNNs

- AlexNet
- VGG16
- GoogLeNet
- ResNet

Meilstones of CNNs

- AlexNet
  - Motivation
  - Architecture
  - Advantage
  - Drawback
  - Possible improvements

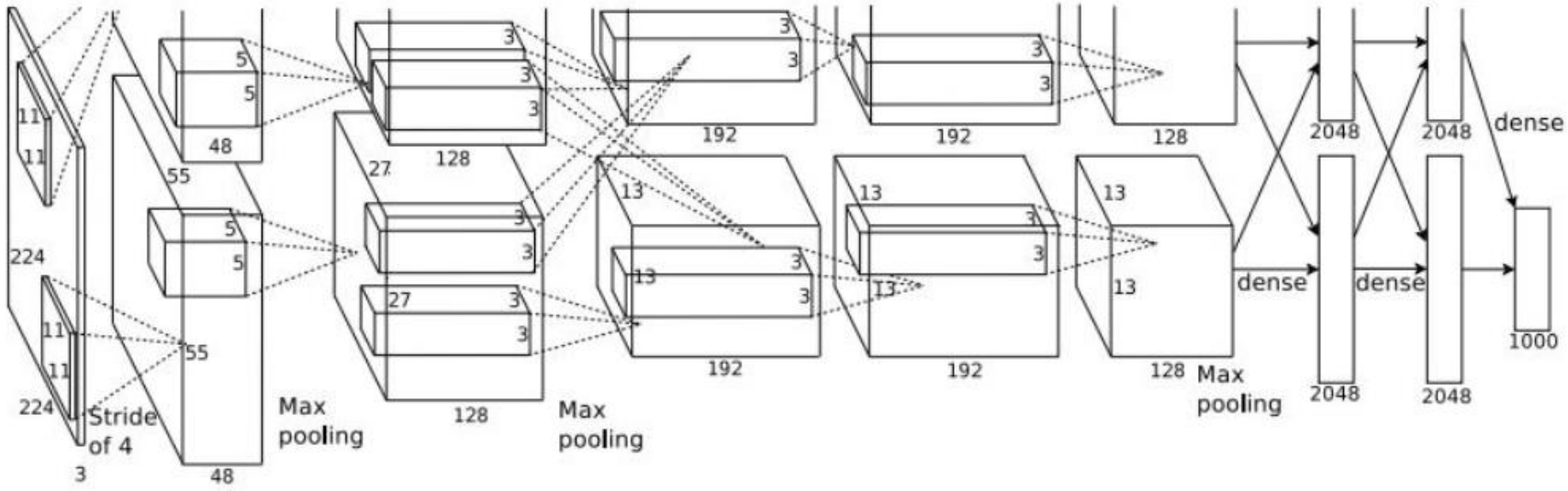
Meilstones of CNNs

- AlexNet
  - **Motivation**:

1. 避免手工提取特征,考虑**数据驱动**，采取end-to-end的训练方案，希望**通过网络结构提取**比手动提取的features更加有效的features
2. 试图满足前面提到的若干**物理不变形**

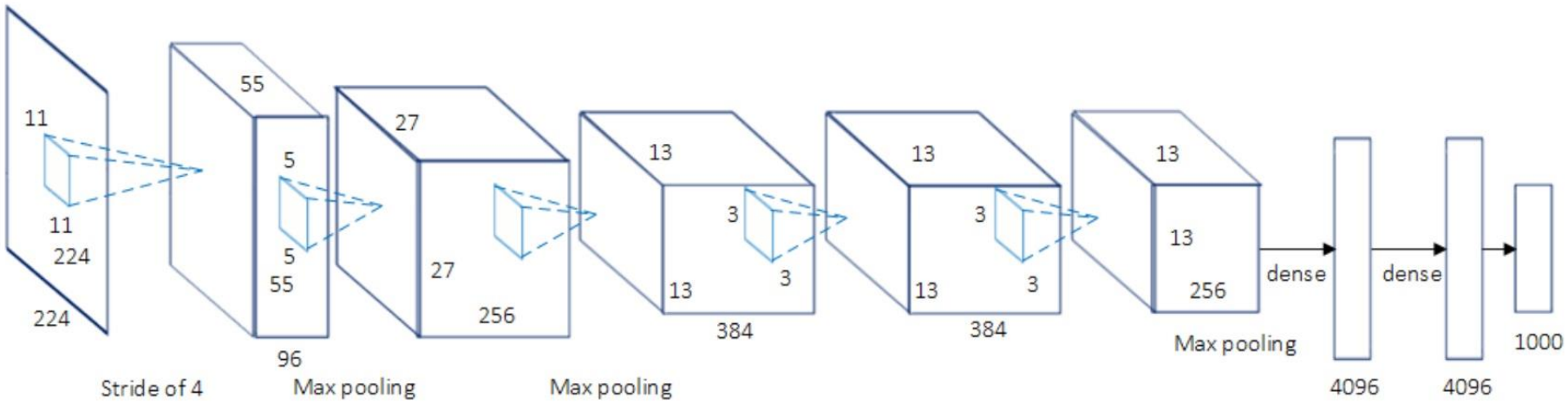
Meilstones of CNNs

- AlexNet
  - **Architecture**



Meilstones of CNNs

- AlexNet
  - **Architecture**



Meilstones of CNNs

- AlexNet
  - **Advantage**

1. **Convolution Layer**：相比FCNN，引入卷积 (small filter, shared weights)，大大降低了参数规模和计算量
2. **分层提取features**：相比其他方法，由于卷积层的重复堆叠，从低层到高层逐步提取的features由**简单的颜色边缘等物理特征**到**复杂的语义特征**，强大的特征表达实现了更高的准确率
3. 首次使用了**ReLU**激活函数，使之有更好的梯度特性、训练更快
4. 首次在CNN中使用了**随机失活(dropout)** 这个正则化手段来**防止overfit**
5. 大量使用Data Augmentation(**数据扩充)**技术：增加数据集的同时增强了算法的鲁棒性

Meilstones of CNNs

- AlexNet
  - **Advantage**

6. 开创性：第一次在超大数据集上成功训练了7层的 DNN(AlexNet之前的FCNN最多2层，而LeNet-5处理的数据集较为mini)。开启了深度学习的时代，也启发了用GPU运行深度学习的新思路.

AlexNet 是ILSVRC2012-winner : 16.4%, 而2011的winner是25.8%，错误率降低了**10%**！

Meilstones of CNNs

- AlexNet
  - **Drawback**

1. **计算量和参数量**：仅仅7层就有60 million个权重参数(最后面的FC Layer贡献了绝大部分的参数)
2. **Filter-size过大**：大小为7x7，对于相邻像素间的关系有点照顾不周，故而小尺度features容易漏检

Meilstones of CNNs

- AlexNet
  - **Possible improvements**

1. **减少参数，降低计算量：**一个直观的思路就是使用更小的filter.
2. 探索**难以训练的本质原因：**e.g. 参数初始化难题，梯度消失和梯度爆炸等，这些对应于每一个Train-Trick.
3. 修改结构**加深CNN：**训练比7层AlenNet更深的CNN，通过增大**模型复杂度**进而获得更高的准确率

Meilstones of CNNs

- VGG16
  - **Motivation**

1. 希望通过加深CNN来获得更高的模型复杂度，从而提高精确度
2. 希望减小神经元的权重参数
3. 继承AlenxNet所有的优点：CNN Layer, Pool Layer，ReLU,等等

Meilstones of CNNs

- VGG16
  - **Architecture**

VGG16

Meilstones of CNNs

- VGG16
  - **Advantage**

1. **smaller filter size**：1,small filter会更关注相邻像素间的关联，对于小尺度features的检测会更有效；2,通过理论证明，3个3x3的fliter相比1个7x7的fliter，effictive respective field是相同的；3,并且通过理论计算，具有更少的神经元权值参数；4,更少的参数确保了搭建更深的网络不至于参数爆炸
2. **more deeper**: smaller filter支持了VGG的16层，更深的CNN提高了模型复杂度，进而获得了更高的精确率(因为具有更强的non-linear approximate 能力)

Meilstones of CNNs

- VGG16
  - **Drawback**

1. 那个很费参数的FC Layer依旧存在
2. 只有一个3x3的filter，检测features的能力有限

Meilstones of CNNs

- VGG16
  - **Possible improvements**

1. 用其他Classifier取代FC Layer
2. 加入更多尺度的filter
3. 加入新的filter要优化网络结构确保，不至于参数爆炸

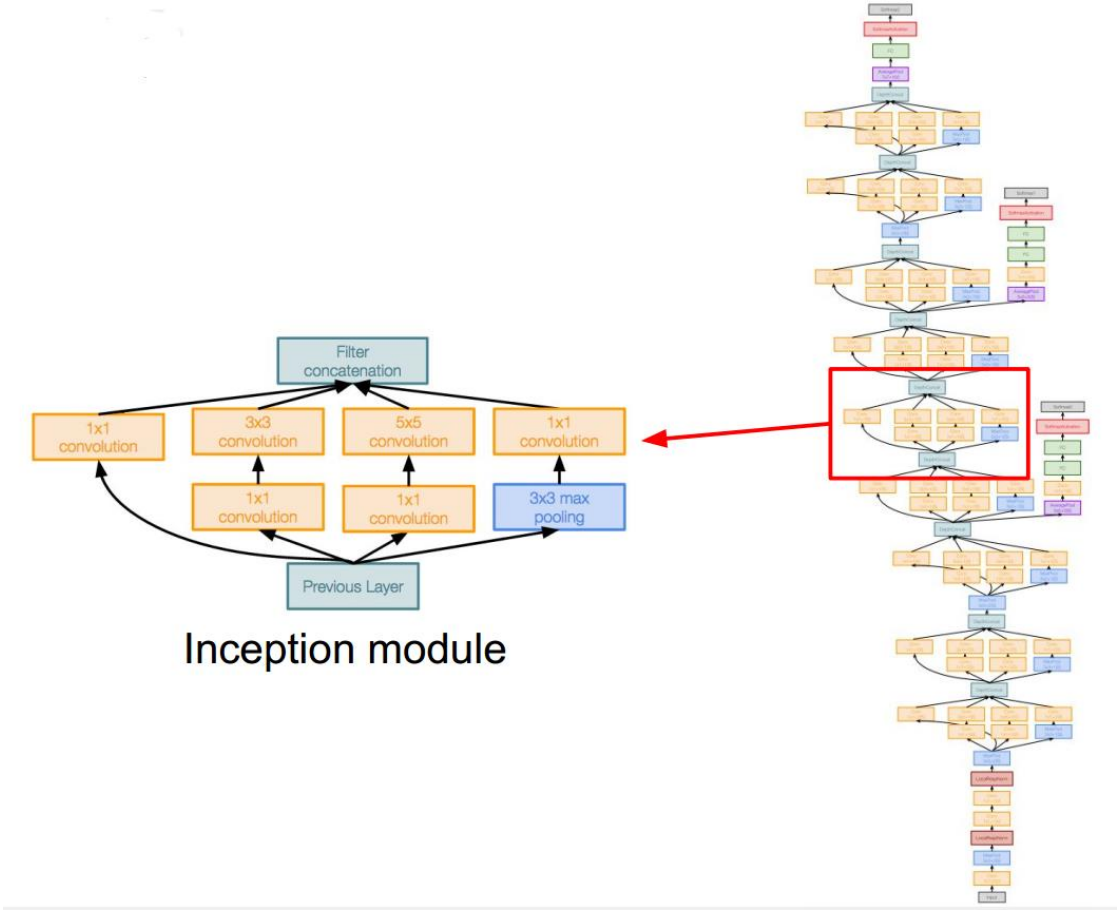
Meilstones of CNNs

- GoogLeNet
  - **Motivation**

1. 希望减少参数量
2. 希望提高特征提取的灵活性
3. 增加组件同时又能控制计算量不至于爆炸

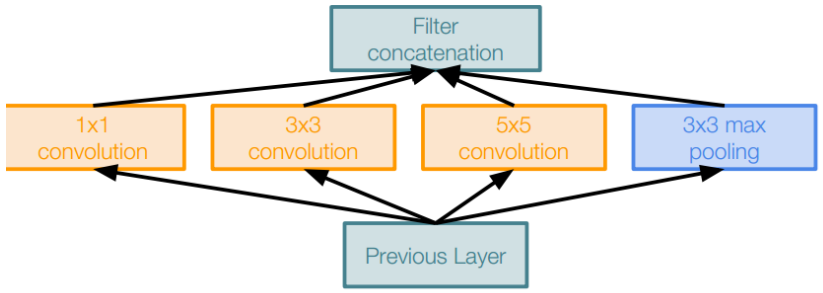
Meilstones of CNNs

- **Architecture**



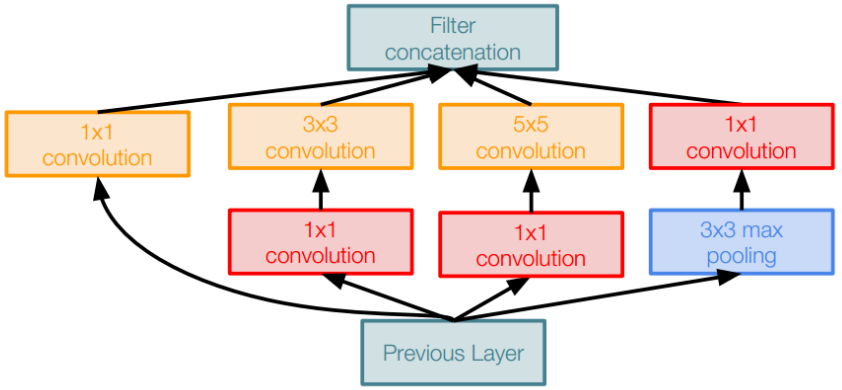
Meilstones of CNNs

- **Architecture**



Naive Inception module

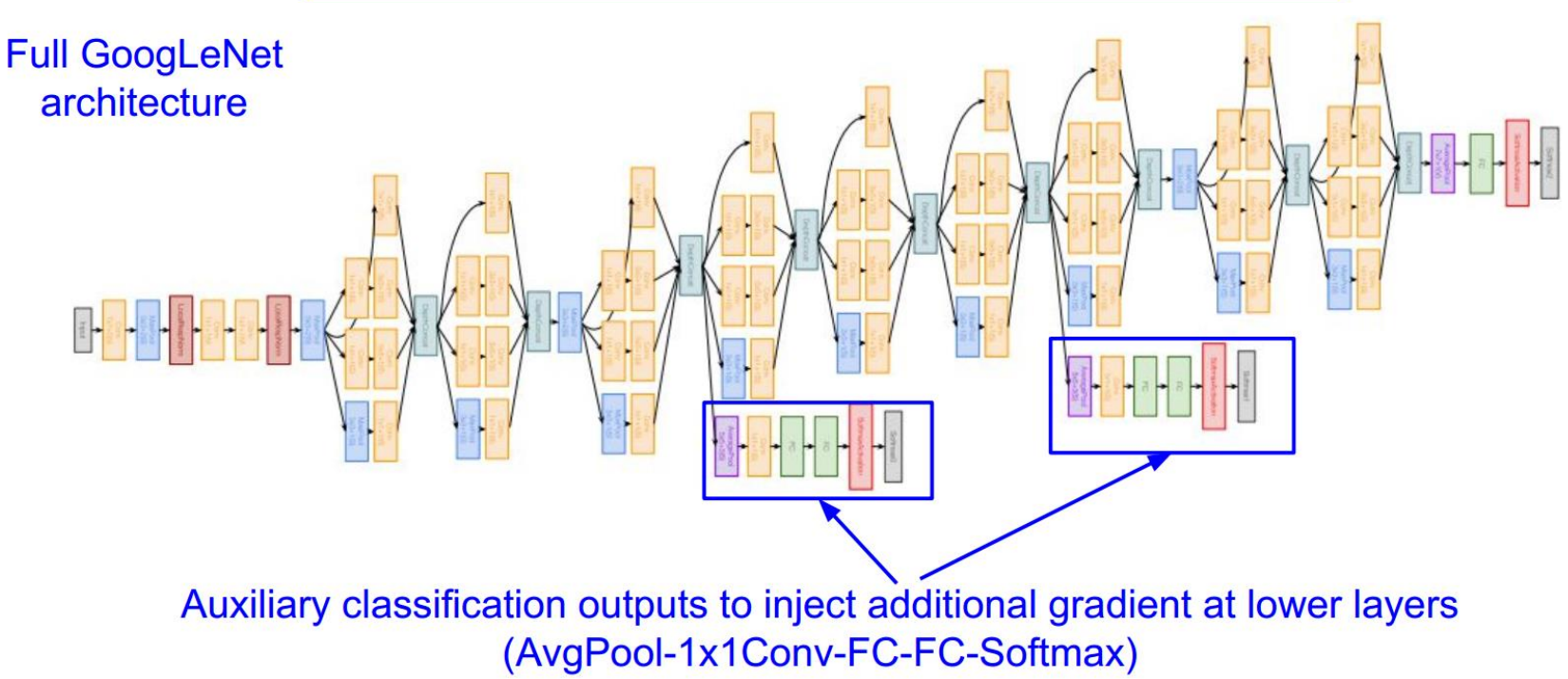
1x1 conv "bottleneck" layers



Inception module with dimension reduction

Meilstones of CNNs

- **Architecture**



Meilstones of CNNs

- GoogLeNet
  - **Advantage**
    1. **Deeper network**：22层
    2. **多尺寸的filters**：有1x1,3x3,5x5，提取features更加灵活，更低的漏检
    3. **Efficient "Inception module"**：通过这个精巧设计的module确保增加filter的同时，不会导致计算量爆炸(1x1的filter降低spatial resolution进而减少计算量)
    4. **结构简单**：反复堆叠**Inception module**
    5. **No FC Layers**:：only 5 millon paras, 12x less than AlexNet

Meilstones of CNNs

- GoogLeNet
  - **Drawback**

1. **hard-optimization**：22层巨大的网络导致优化空间巨大，梯度反向传播的过程也更加艰难。为了对抗梯度消失和梯度爆炸，加入了很多train tricks；
2. **初始化难设置问题**：依旧存在；不好的初始化参数值会导致梯度很难快速下降，梯度卡壳，进而loss optimization难以收敛

Meilstones of CNNs

- GoogLeNet
  - **Possible improvements**

1. train problem：2015年Google提出Batch_Normalization，相当程度上一举解决了CNN中【梯度消失，梯度爆炸，权值参数初始化难题】
2. 尝试更深的CNN：继续加大模型复杂度，但是很难，因为hard-optimization

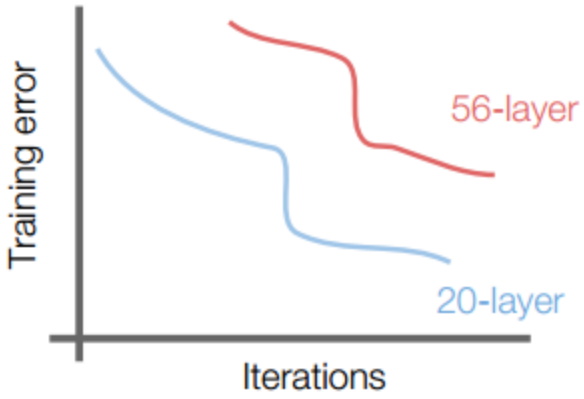
Meilstones of CNNs

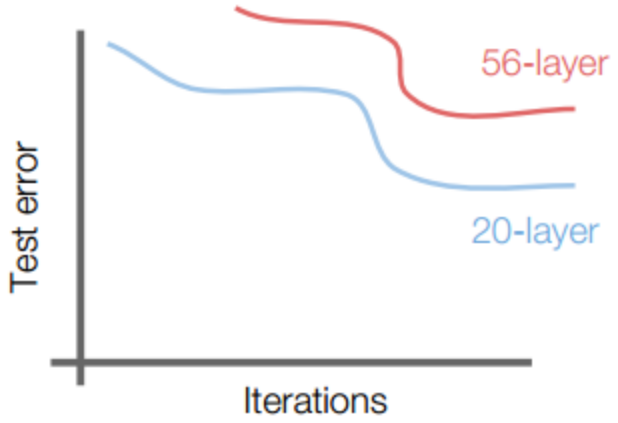
- ResNet
  - **Motivation**

1. 基于已有的BN Layer**解决：已被证实**导致hard-optimization的三个因素: (1)**不恰当的参数初始化**进而导致梯度无法如预期般高速下降，e.g. plateu,local_minima; (2)**梯度消失**; (3)**梯度爆炸**
2. 试图解决，当时业界存在的一个巨难问题：加深CNN反而降低精度

Meilstones of CNNs

- ResNet
  - **Strange**

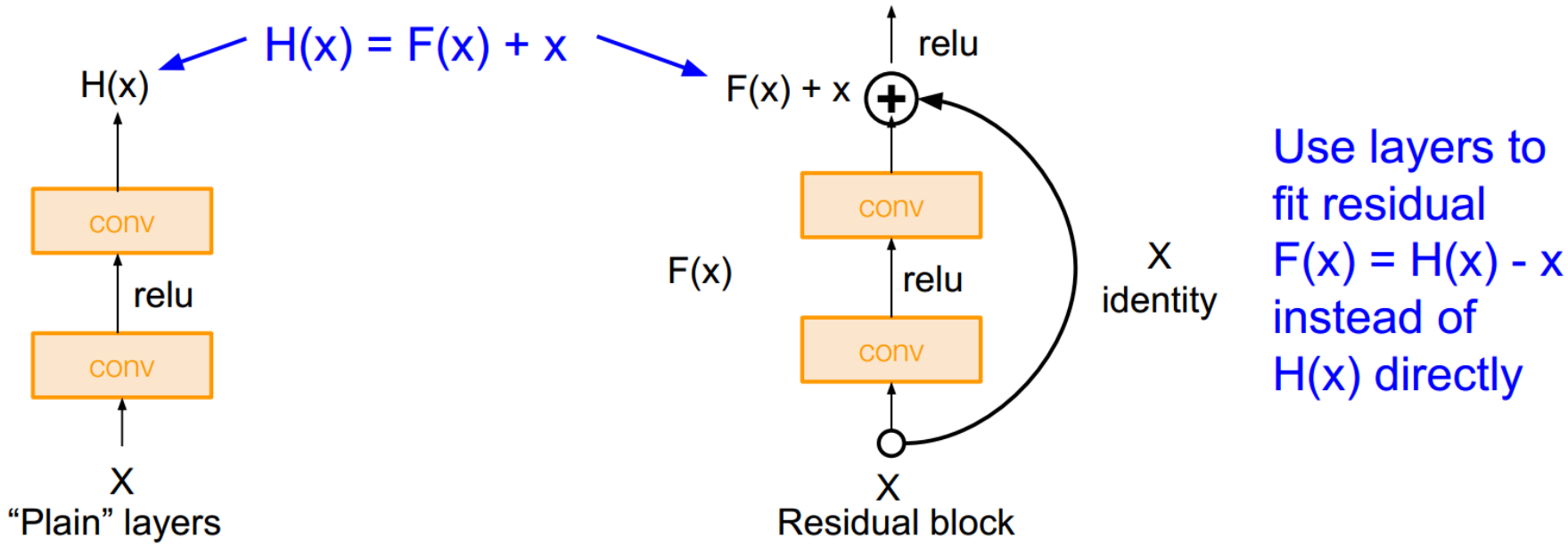




More **Deeper**, **Worse** Accuracy **!!!**

Meilstones of CNNs

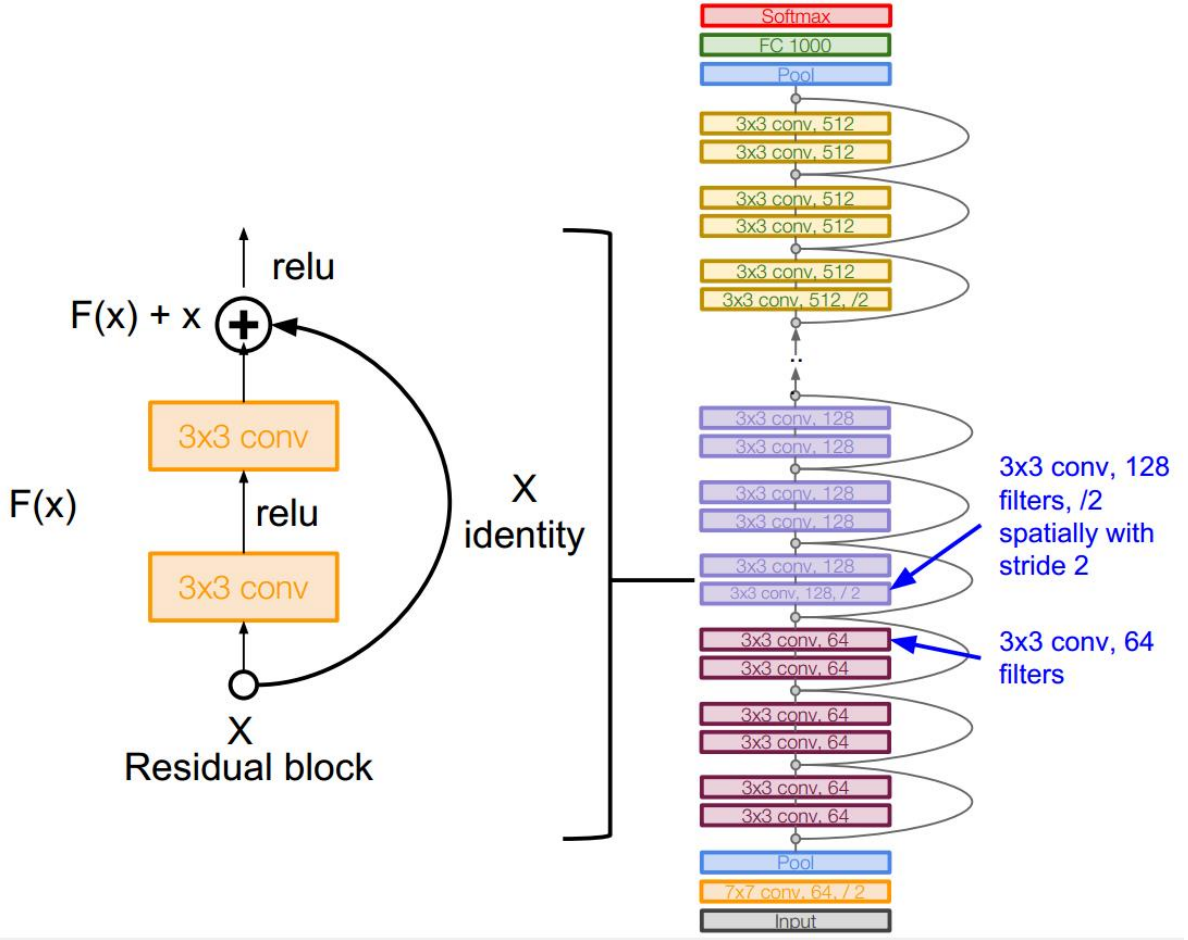
- **Architecture**



Use layers to fit residual F(x) = H(x) - x instead of H(x) directly

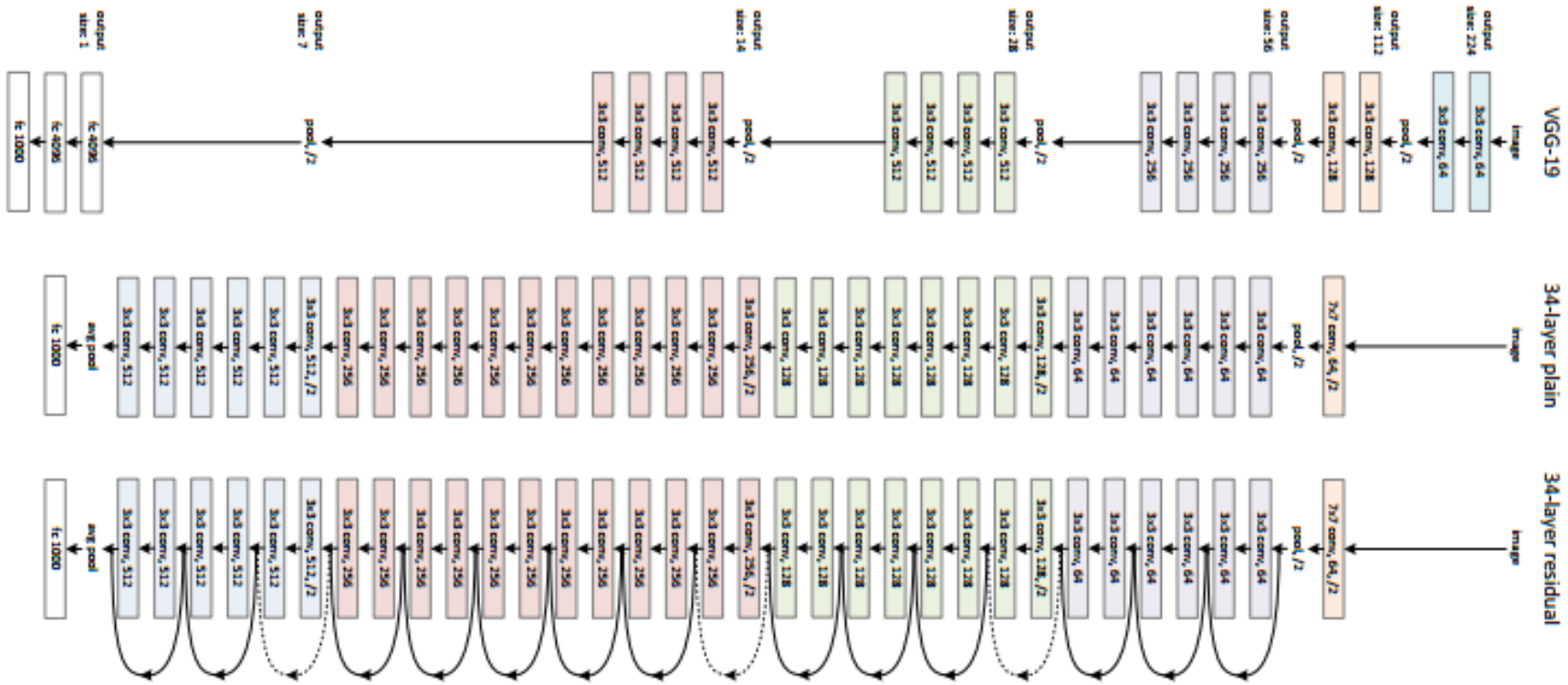
Meilstones of CNNs

- **Architecture**



Meilstones of CNNs

- **Architecture**



Meilstones of CNNs

- ResNet
  - **Advantage**

1. **Residual block**: 额外添加的跨边确保了梯度流能在极深的网络中正常传播，解决了之前那个"反常“(more deeper, worser performance)
2. 解决**超深**网络中梯度消失和梯度爆炸：借助**identify map**(基于**Residual block**)和Google提出的**Batch_Normalization**
3. **More Deeper**：超级Deep，152层！
4. **bottleneck layer:** 确保了计算量不会爆炸: 借鉴了GoogLeNet的Inception module大大减小计算量
5. 就ImageNet数据集而言，ResNet的精确度**首次**超越人类

Meilstones of CNNs

- ResNet
  - **Drawback**

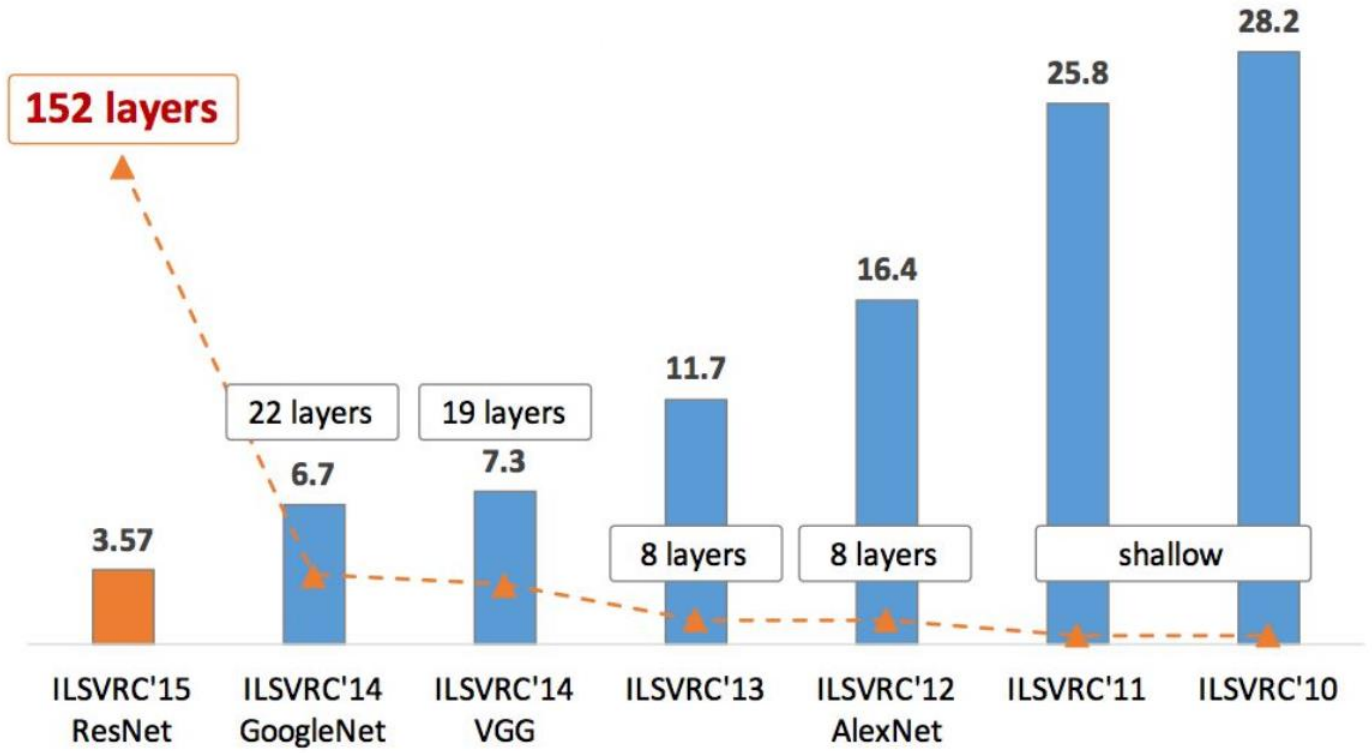
仅作为通用图片领域的classification的BaseNet而言，ResNet的识别精确度已经超过了人类。暂无

Meilstones of CNNs

- ResNet
  - **Possible improvements**
    1. 通用领域的图像识别的速度更快，精度更高(这一点几乎无法改进)
    2. 细分领域的特殊问题(e.g. 红外图像，卫星图像的识别难点)
    3. 图像分类中的一些通用难题. (e.g. 小目标检测，复杂光照条件下的算法鲁棒性，复杂遮挡，复杂前景，同类object混淆)

Meilstones of CNNs

- 性能对比



Outline

- Brief to CV
- Classification
- Localisation
- Object Detection
- Segmentation

Task description

- **Definition**:

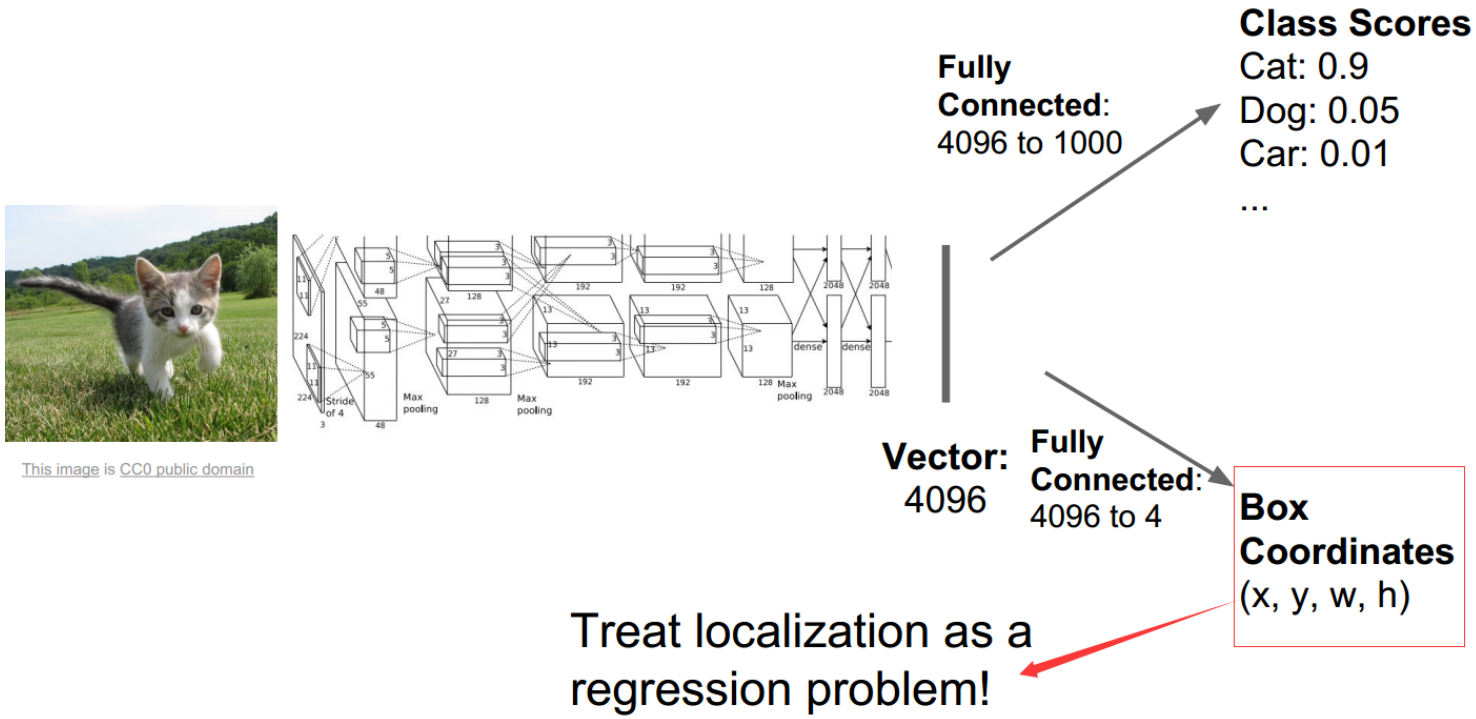
  输入图片，输出图像中感兴趣目标的类别及其矩形检测框的4点坐标

- **Category**:
  - 通用场景中定位图片中感兴趣的目标
  - 人体位姿定位/人脸定位
  - 一切**fixed output category** 的问题均可规约到这个problem
- **Solution**:

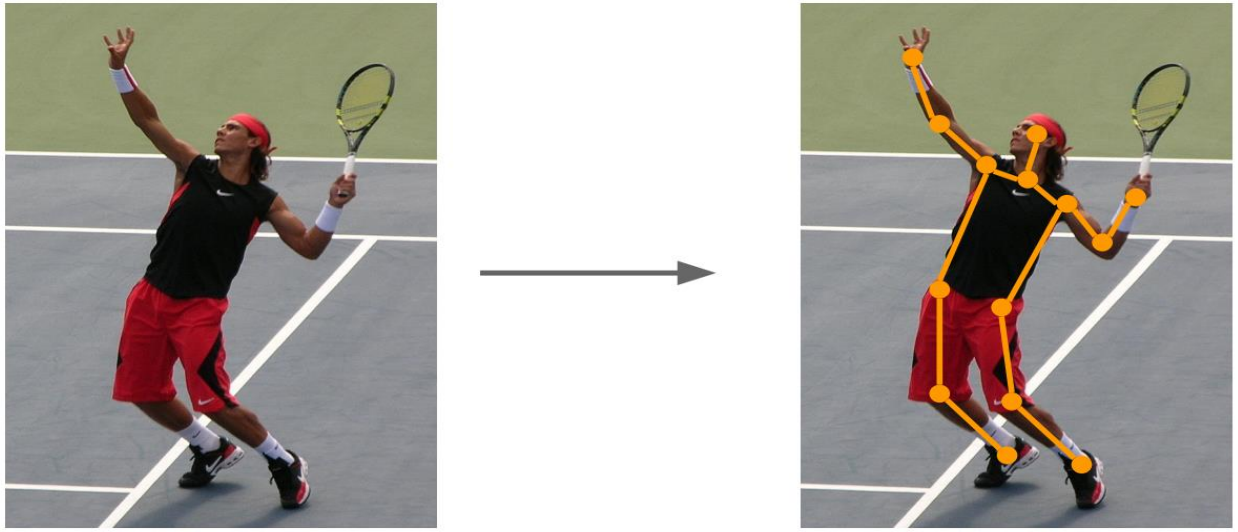
  多任务学习，网络带有两个输出分支。一个分支用于做**图像分类**，和单纯图像分类区别在于这里还另外需要一个**“背景”类**。另一个分支用于判断目标位置，即完成**回归任务**输出四个数字标记包围盒位置，该分支输出结果只有在分类分支判断不为“背景”时才使用。数学模型：
  multiclass_classification + regression task of **fixed output category**

Examples-1

Examples-1


This image is CC0 public domain
Fully Connected: 4096 to 1000
Class Scores
Cat: 0.9
Dog: 0.05
Car: 0.01
...
Vector: 4096
Fully Connected: 4096 to 4
Box Coordinates (x, y, w, h)
Treat localization as a regression problem!

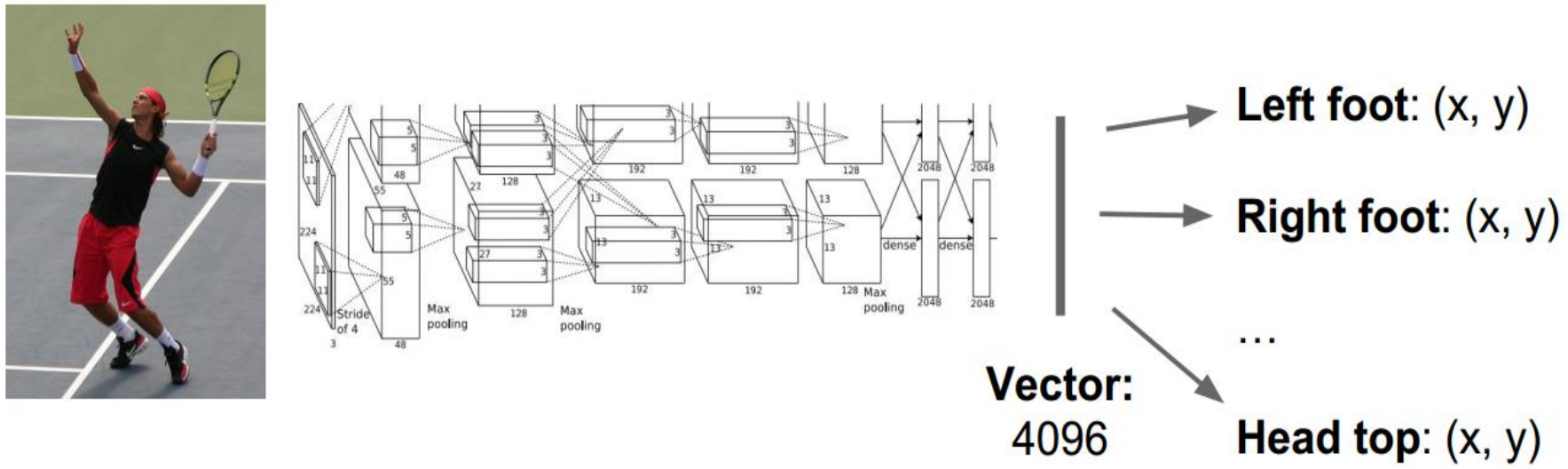
Examples-2



Represent pose as a set of 14 joint positions:

Left / right foot
Left / right knee
Left / right hip
Left / right shoulder
Left / right elbow
Left / right hand
Neck
Head top

Examples-2


Left foot: (x, y)
Right foot: (x, y)
…
Head top: (x, y)
Vector:
4096
Stride of 4
Max pooling
Max pooling
Max pooling
dense
dense

Outline

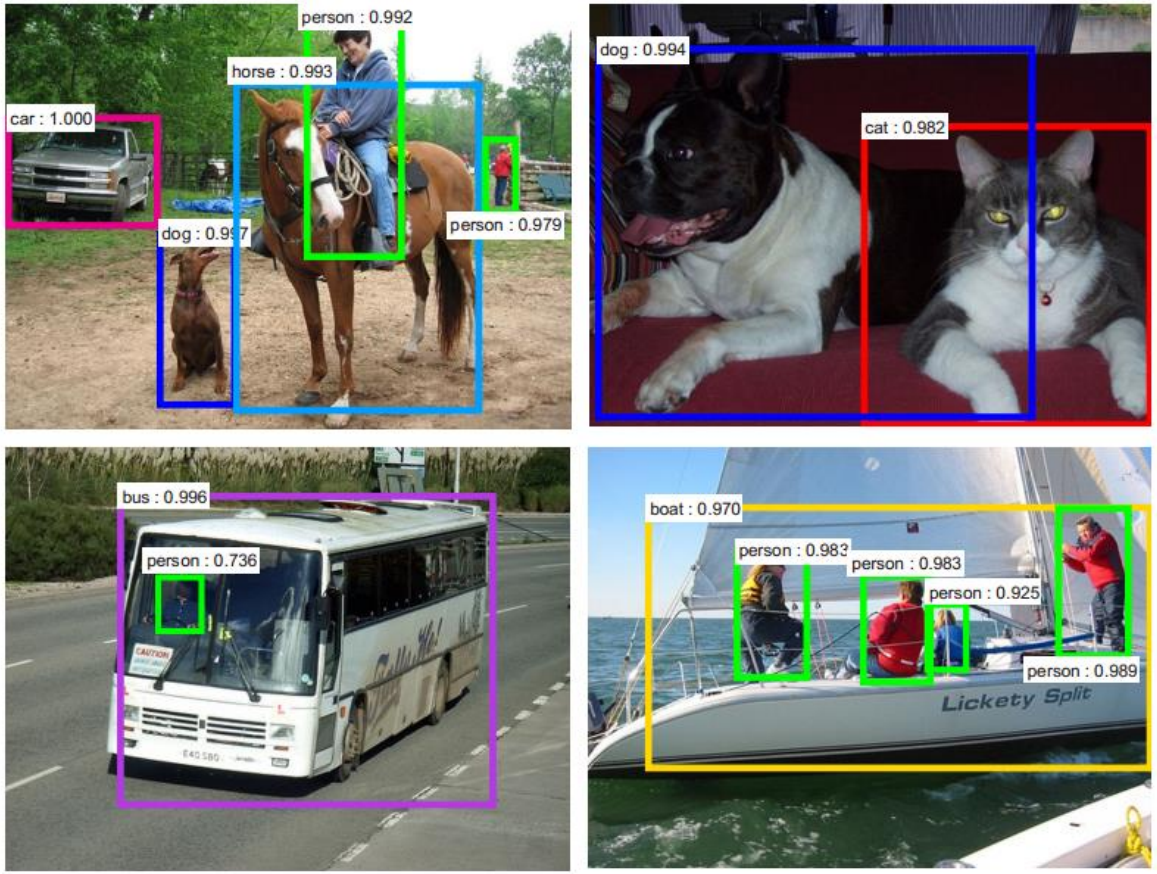
- Brief to CV
- Classification
- Localisation
- Object Detection
- Segmentation

Task description

- Definition(2d-object detection)
- Challenges
- Key motivations

Task description

- object detection(2d)



Task description

- object detection(2d)-形式化：

  - Detection box: (x,y,w,h)
  - Input image: raw numpy array with a seris of ground_truths & categories

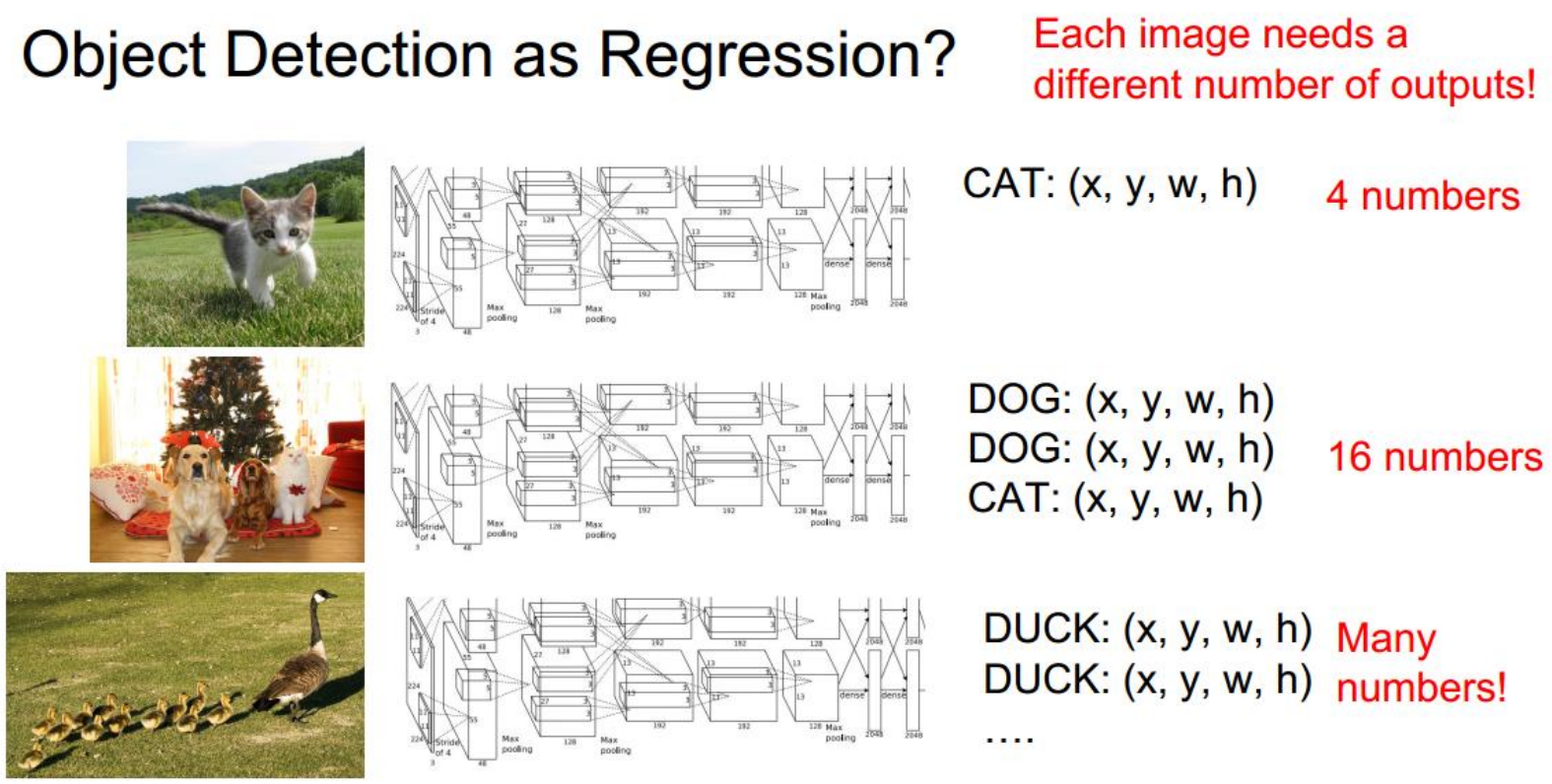
  - Predcit of model: (delta_x,delta_y,delta_w.delta_h) & category
  - Output detection box： (x,y,w,h)
  - Output category: one class (multi_class)

Task description

- Challenge
  - 测试图片中的**3个变化**值:
    1. object数目及类别是变化的
    2. object出现的位置是变化的
    3. object的尺寸大小是变化的
  - 图片识别中需要满足的若干复杂的**物理不变形**
  - 检测方法自身的缺陷：比如CNN天生对小目标识别不work

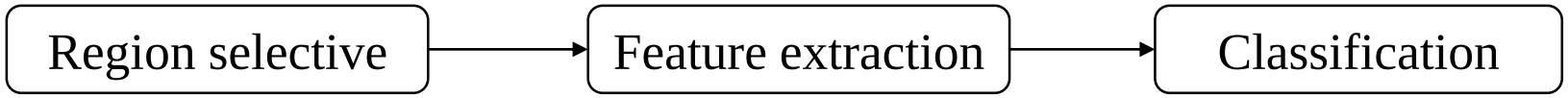
Task description

- Challenge



Task description

- Key motivation-1
  1. 借助Slide Window，把图片“分割” 为若干个“合适大小”的子区域
  2. 在一个小的子区域，假设只有一个object，就可以简单套用Regression task of **one** category
  3. 体现了算法设计中的**分而治之**的思想
  4. Traditional pattern:



Task description

- Key motivation-2
  1. Slide window体现的**分治思想是OK**的，而是Slide Window这种【**滑动窗口去寻找**所有可能存在object的位置】的**方式是实际不可行**的，加上限制又会降低recall & precision
  2. 所以，要定义**基于图像语义来提取候选框**的高效算法，例如Selective Search(基于颜色、边缘方向梯度、面积、形状等低级物理层次语义)
  3. 上述motivation催生出object detection中，RCNN 系列方法

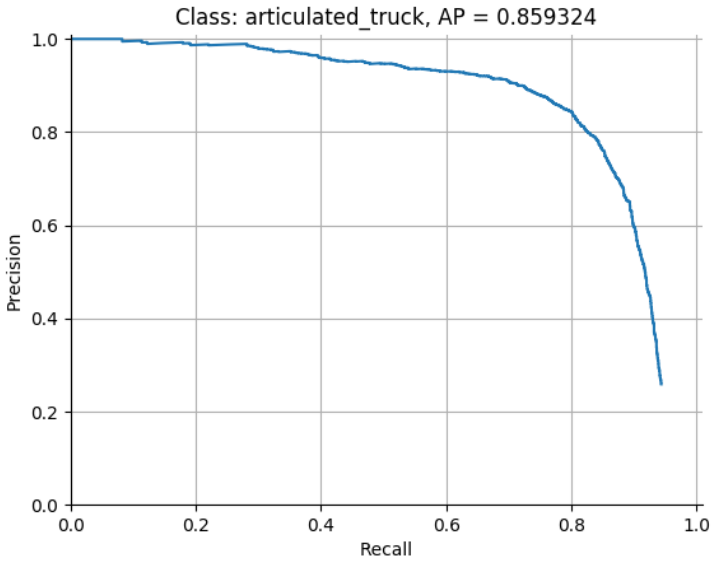
Task description

- Key motivation-3
  1. Selective Search类的Region Selective Aigorithm效果很好，唯一的问题是：相比Conv ops**速度太慢**，且**无法借助GPU**并行加速
  2. 可以考虑基于 **密集采样 + CNN** 来实现Region Proposals Selection
  3. 上述motivcation产生了Faster-RCNN & YOLO & SSD 系列方法

Task description

- Key motivation-3

1. 有了RPs，如何从RP中提取 **powerful feature** 来表征这个RP从而提高检测效率(recall & precision)，RCNN中首次引入ConvNet作为feature extractor.
2. 检测的效率：**precise & recall(所有改进的最根本的两个motivation)**，可结合为一个指标mAP(mean of AP)



Methods

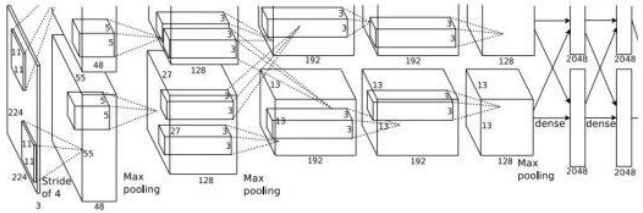
- Slide Window
- Two Stage methods
  - RCNN
  - SPP-Net
  - Fast RCNN
  - Faster RCNN
- One Stage methods
  - YOLO
  - SSD
- More…

Slide Window

- Key idea:
  1. Divide a image into **many small** crops
  2. Apply a CNN to these different crops of the image, CNN classifies each crop as **object or background** and regress its **position**

Slide Window

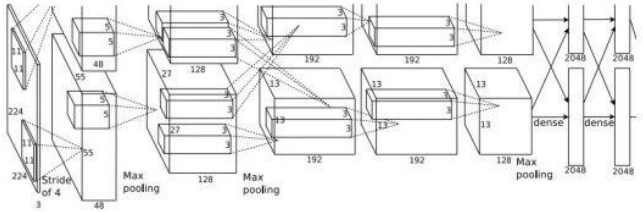
- Key idea:



Dog? YES
Cat? NO
Background? NO

Slide Window

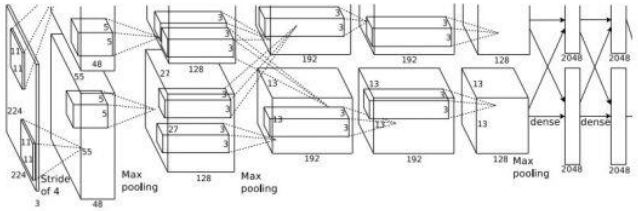
- Key idea:



Dog? YES
Cat? NO
Background? NO

Slide Window

- Key idea:



Dog? NO
Cat? YES
Background? NO

Slide Window

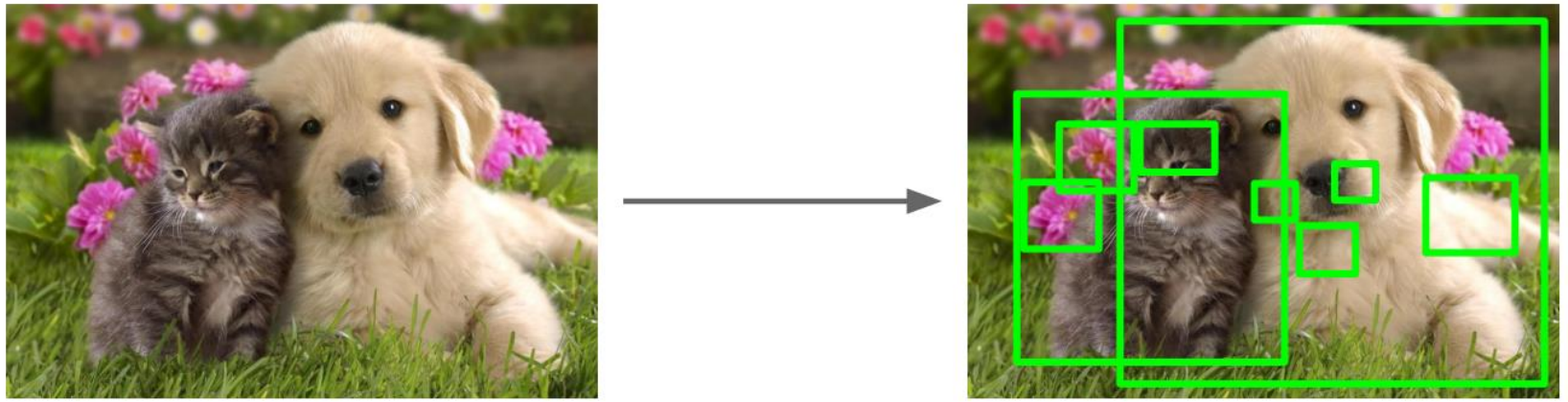
- Key idea:
  - **问题：**
    - 这些crops**怎么选取**？即(crops的**数量**，crops的**尺寸**，crops的**位置，滑动方式**)，且不说暴力穷举是个组合爆炸，就只谈这些crops之间的overlap就产生了相当多的冗余计算
  - **改进：**
    - 吸收Slide Window中分治思想，但不采用滑动的方式，而是**挖掘图像语义特点**，直接生成候选区域

Two Stage Method

- Key idea:
  1. Step1:
     - 基于一个Region Proposals selective algorithm从输入图片中提取若干个候选区域RPs
  2. Step2:
     - 针对每个RPs，基于ConvNet做检测框中的object的**多分类**与检测框位置的**回归修正**

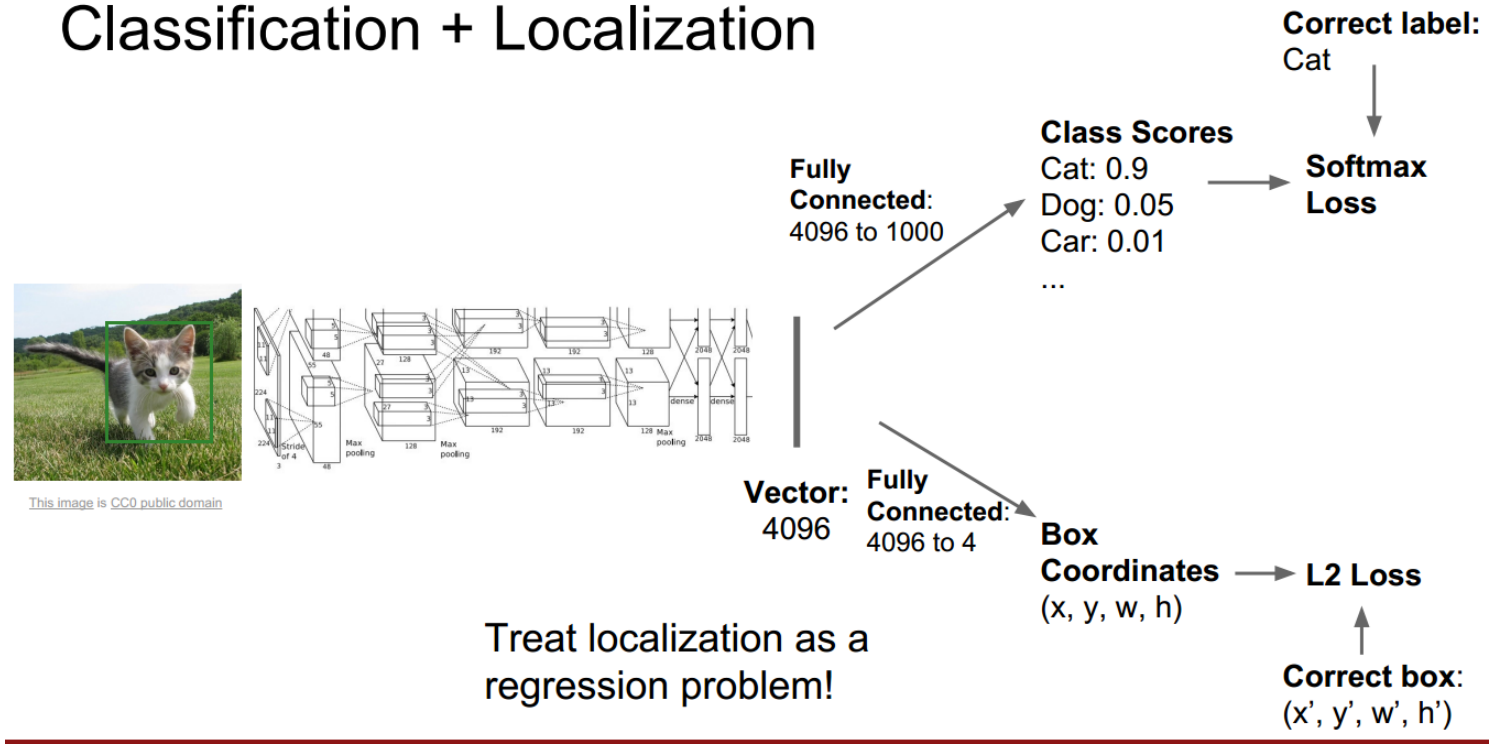
Two Stage

- Key idea:



Two Stage

- Key idea:



Two Stage Method

- Method:

1. RCNN
2. SPP-Net
3. Fast-RCNN
4. Faster-RCNN

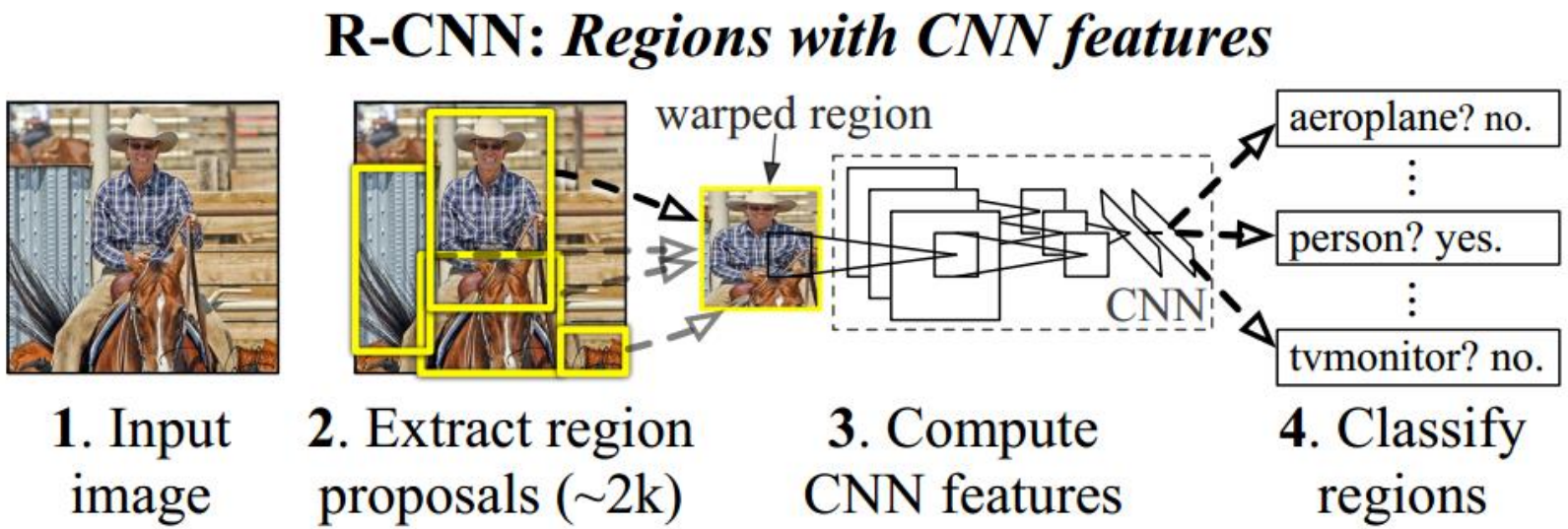
Two Stage Method

- Method:

1. RCNN
2. SPP-Net
3. Fast-RCNN
4. Faster-RCNN

Two Stage

- **Architecture**



Two Stage

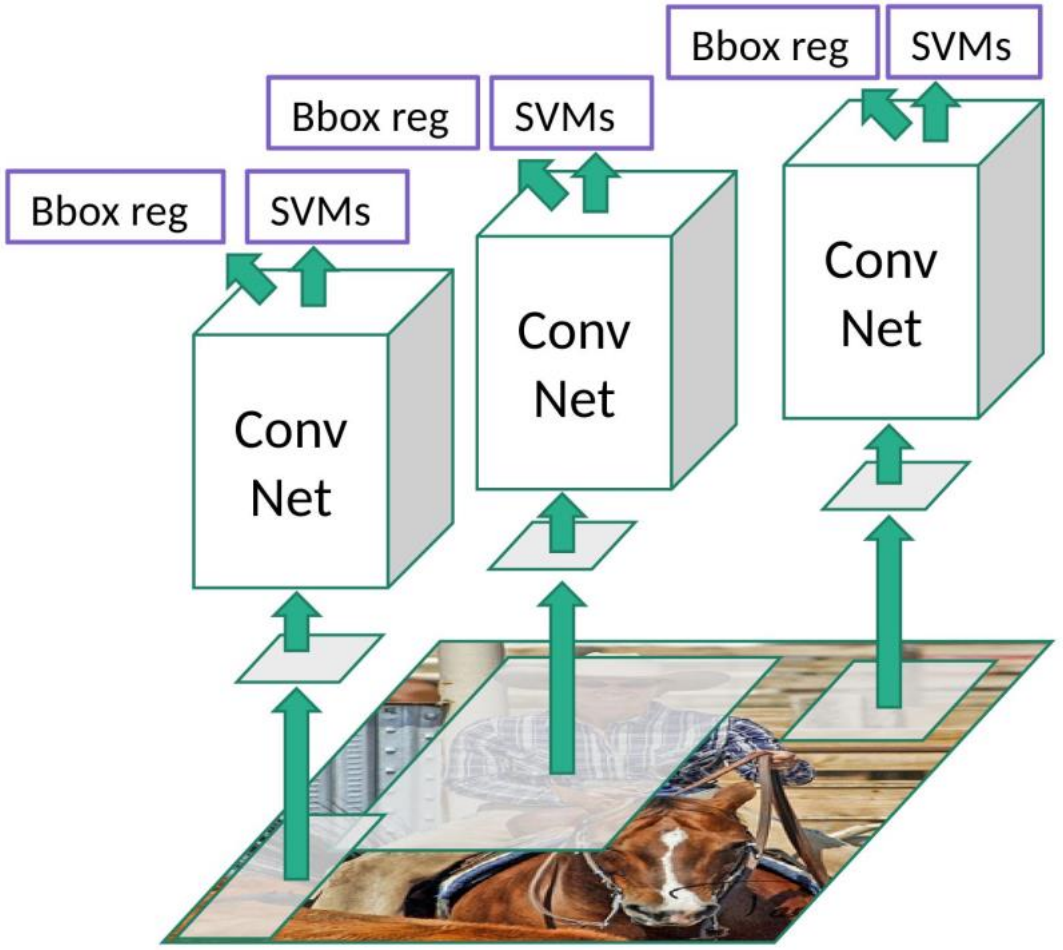
1. 输入测试图片
2. 利用selective search算法在图像中提取2000个左右的region proposals
3. 将每个region proposal整理为(wrap/crop)成227x227的大小并输入到CNN(AlexNet or VGG16)，将CNN的fc7层的输出作为特征。
4. 将每个region proposal提取到的CNN特征输入到SVM进行分类
5. 将每一个RP输入到训练好的对应的一组Linear Regressor(一共训练20组)，得到RP修正后的预测位置
6. 对上述结果集施加非极大值抑制(NMS)，然后将预测输出的类别信息与位置信息施加到原图片，得到处理后图片的输出

Two Stage

- Pros.
  1. Better RP selection: **Selective Search**
  2. Better feature extractor**:** **AlexNet**

- Cons.
  1. Overlap's Conv computation: ~2000 RPs之间大量的overlap产生**大量的重复卷积**计算
  2. Crop/Warpped: resize directly may **loss information**(resize的原因是Conv Layer后面有FC layer)

Two Stage


Bbox reg
SVMs
Bbox reg
SVMs
Bbox reg
SVMs
Conv Net
Conv Net
Conv Net

Two Stage

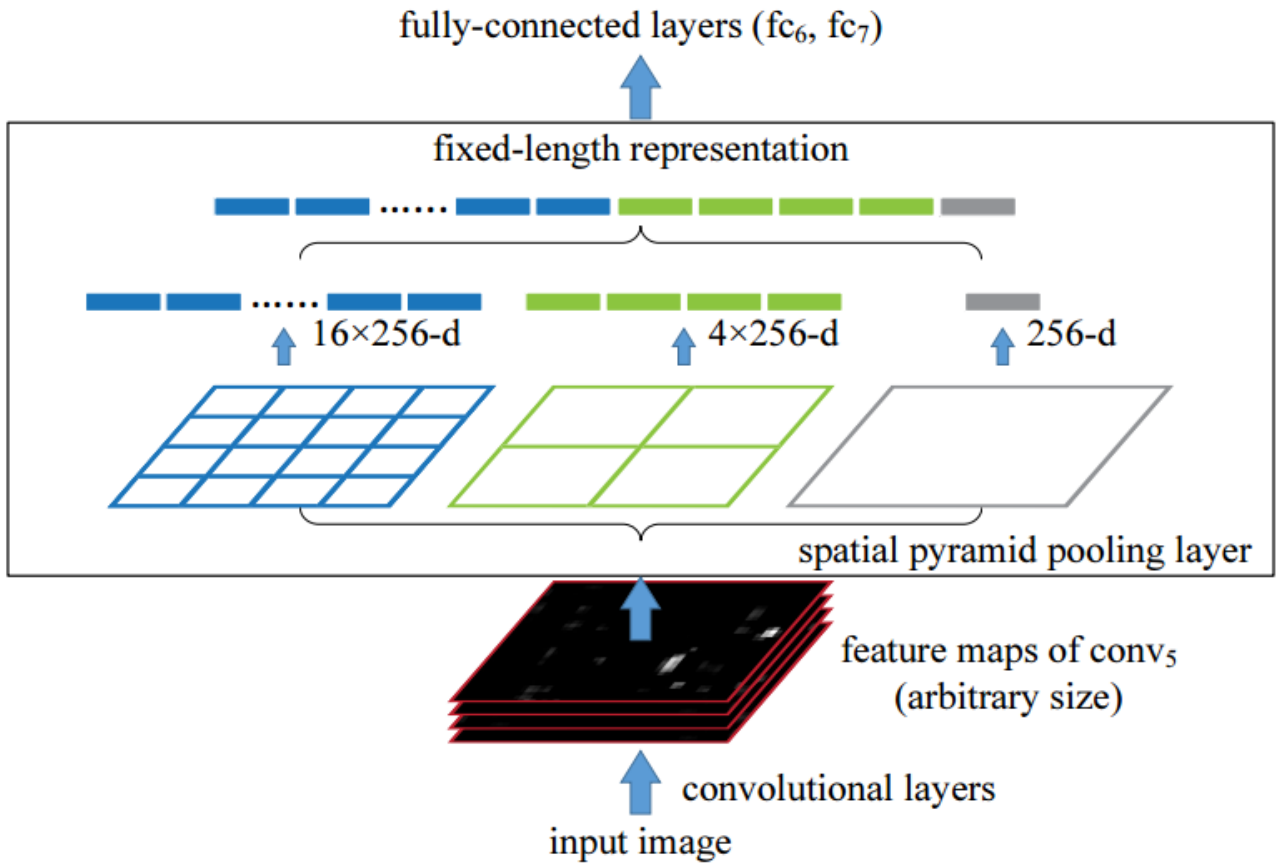
Two Stage Method

- Method:

1. RCNN
2. SPP-Net
3. Fast-RCNN
4. Faster-RCNN

Two Stage

- **Architecture(SPP Layer)**



Two Stage

- Pros.
  1. CNN layer提至网络**最前端**：彻底消除了RPs之间的overlap部分的重复卷积计算
  2. SPP Layer:
     1. (1) generate a **fixed-length** representation **regardless** of image size/scale.
     2. (2) robust to object deformations
- Cons.
  1. Slow RP selection: RP selective的时间比Conv ops的时间要长得多
  2. No end-to-end
  3. SPP Layer无法BP，所以Conv Layer无法在VOC Dataset上做fine-tune

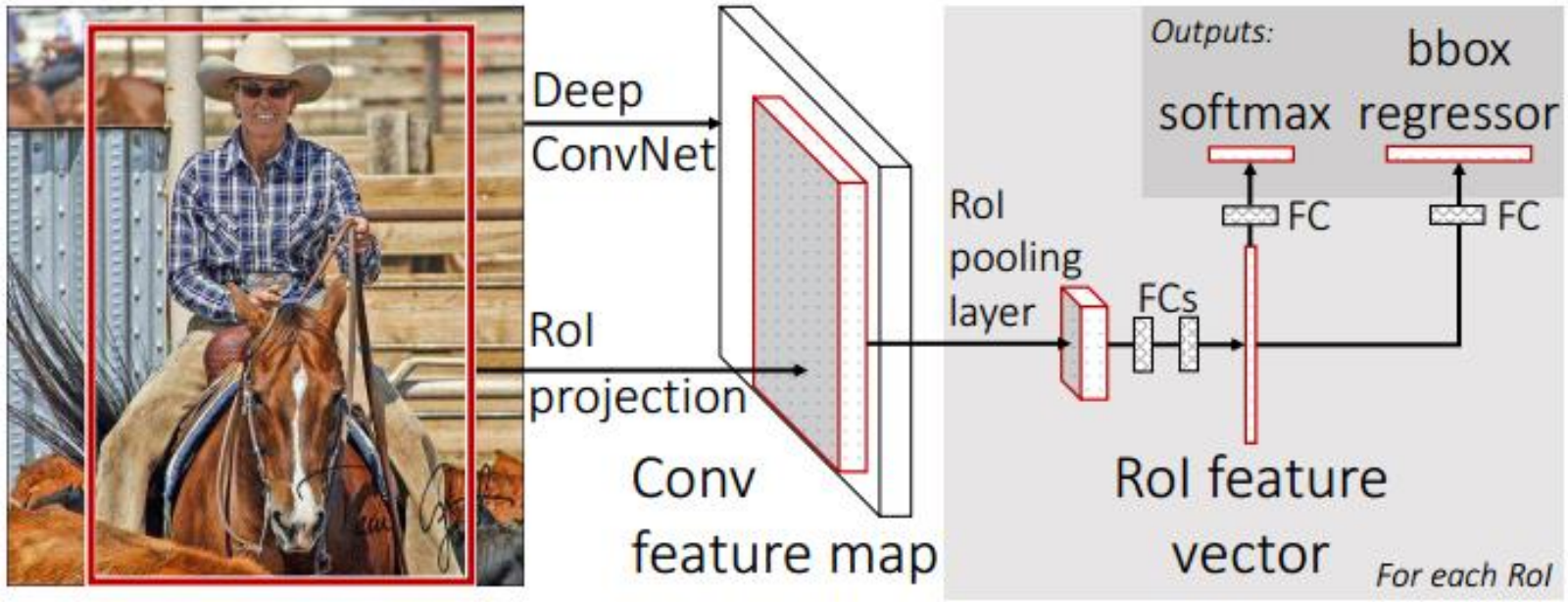
Two Stage Method

- Method:

1. RCNN
2. SPP-Net
3. Fast-RCNN
4. Faster-RCNN

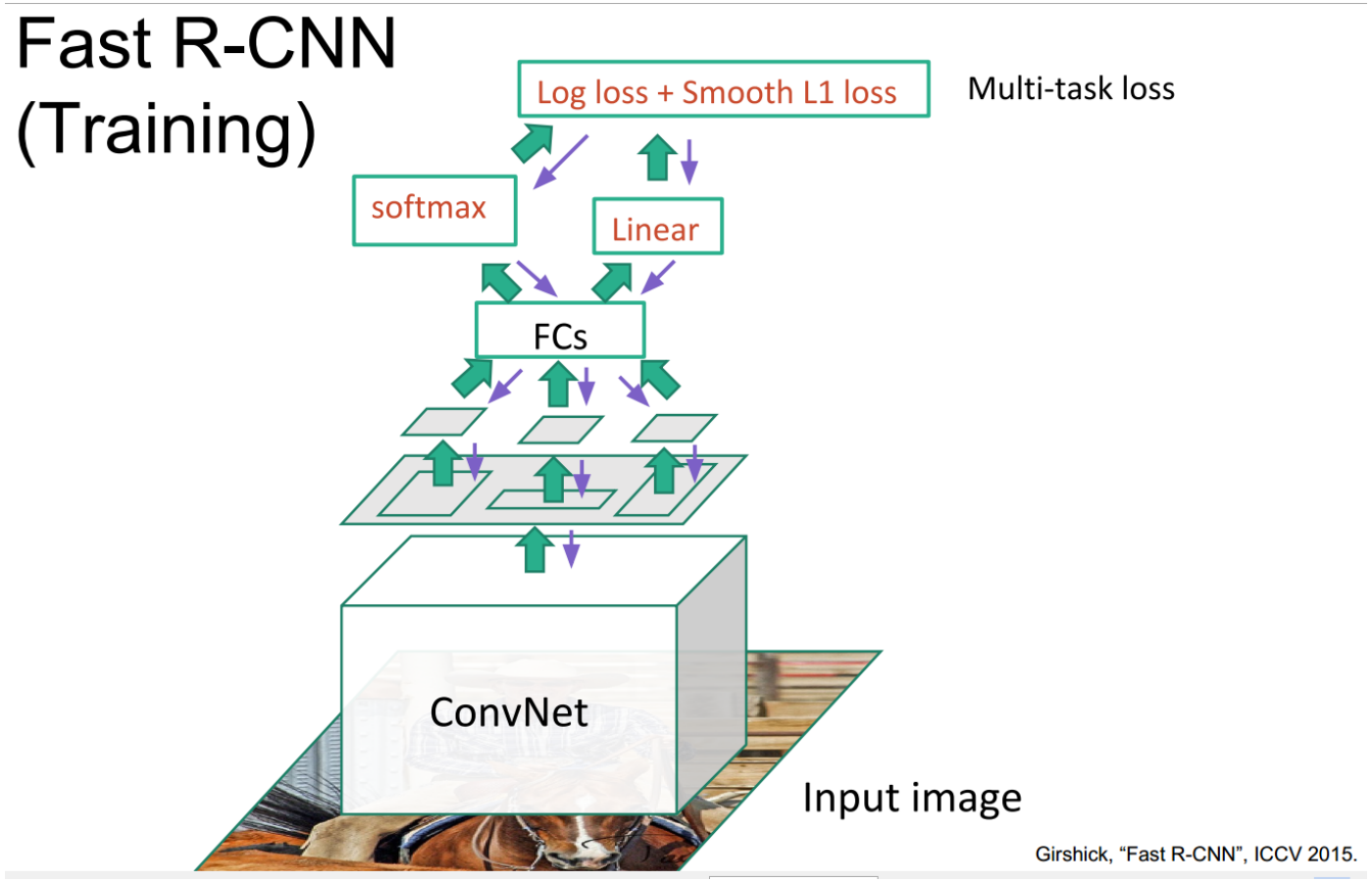
Two Stage

- **Architecture**



Two Stage

- **Architecture**



Two Stage

- Pros.
  1. CNN layer提至网络**最前端**：彻底消除了RPs之间的overlap部分的重复卷积计算
  2. RoI Pooling: 相当于只要一层的SPP Layer，效率接近但可以方便做BP，整个网络都可以在VOC Dataset上做fine-tune
  3. 实现了end-to-end

- Cons.
  1. **Slow** RP selection: RP selective的时间比Conv ops的时间要长得多

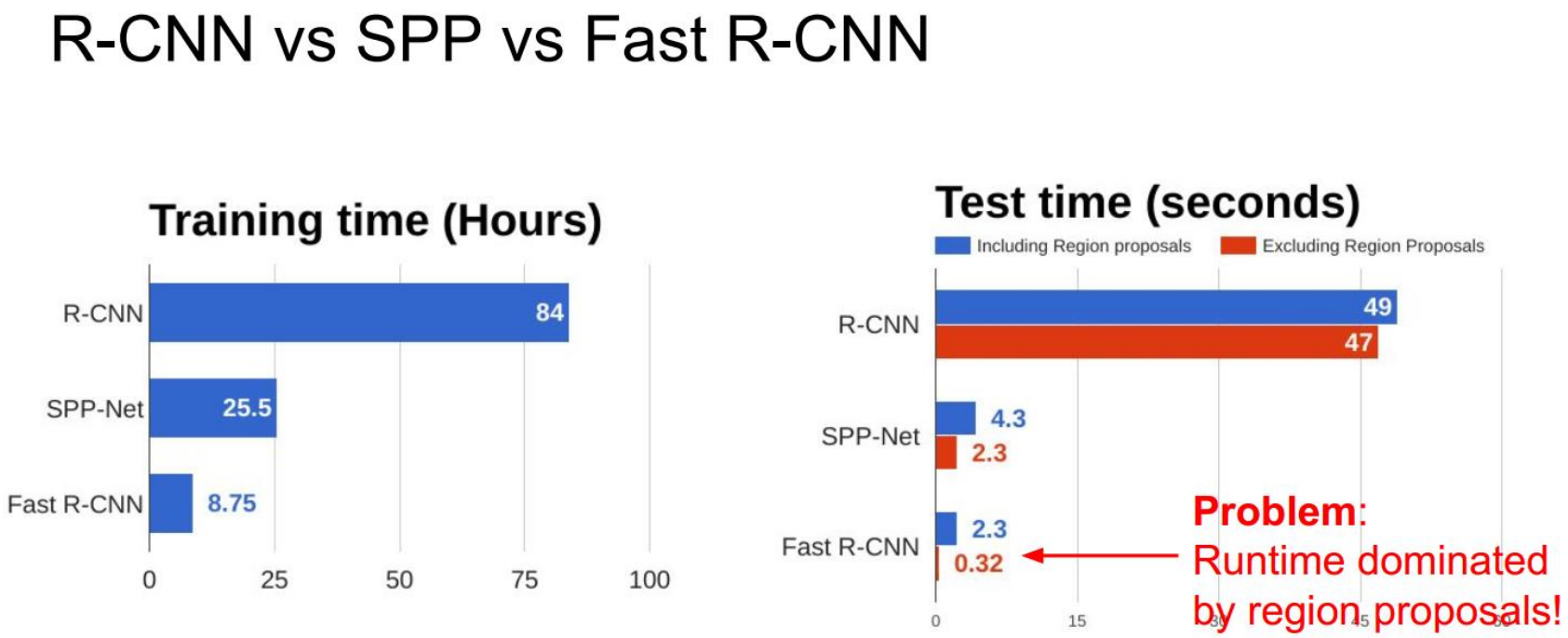
Two Stage Method

- Method:

1. RCNN
2. SPP-Net
3. Fast-RCNN
4. Faster-RCNN

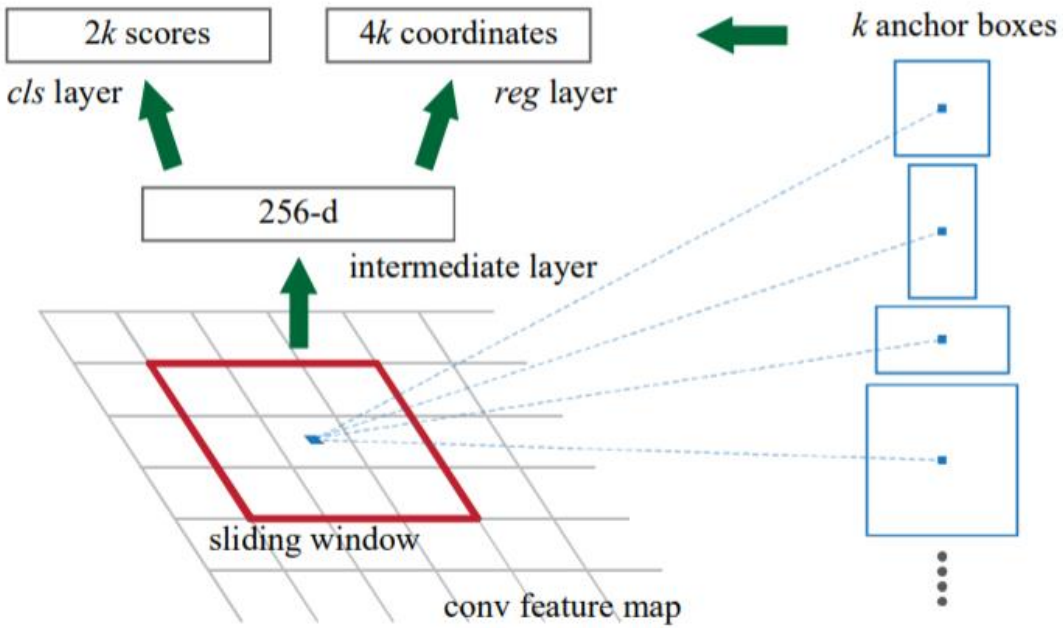
Two Stage

- Bottleneck: selective search



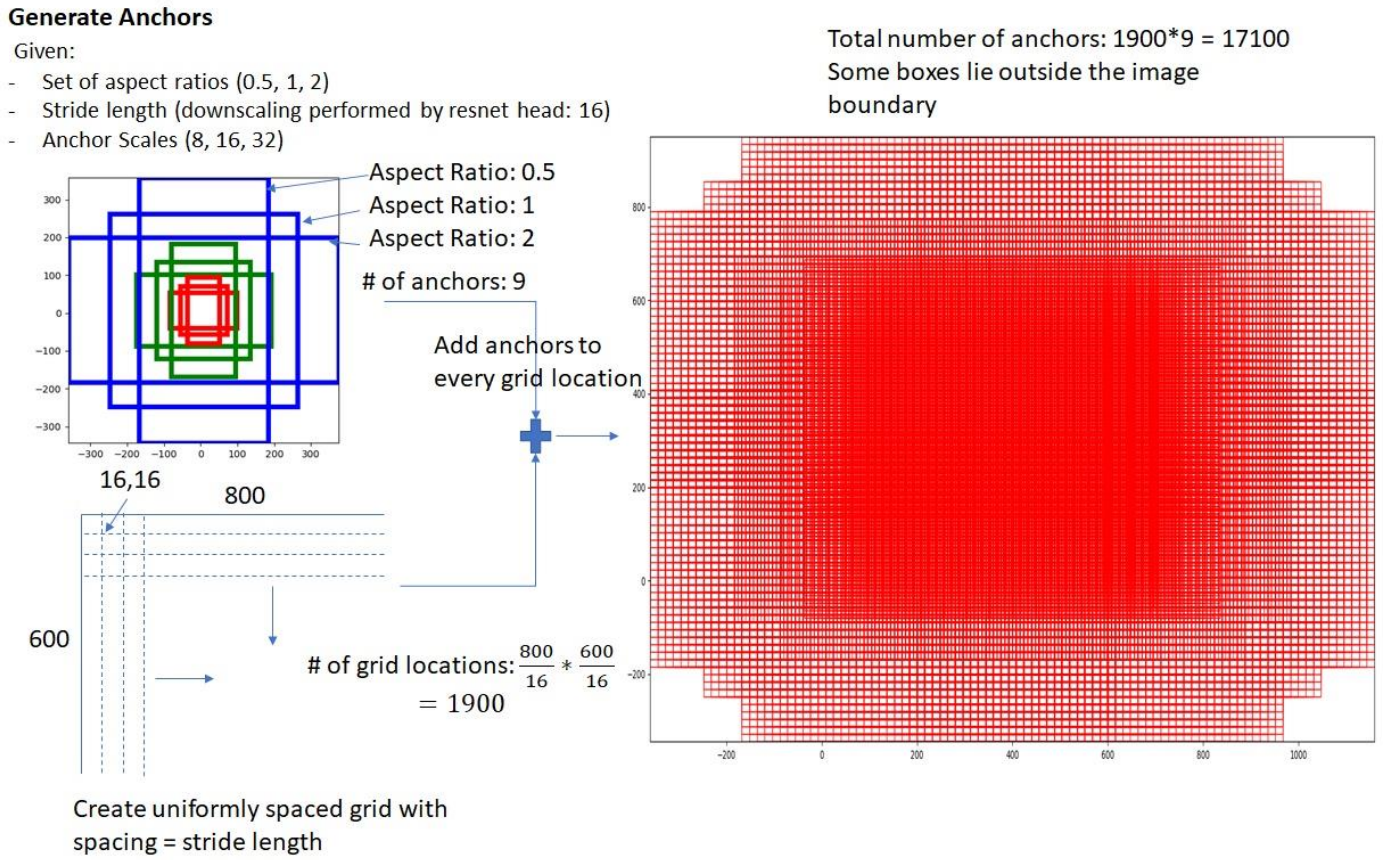
Two Stage

- **Architecture: RPN**



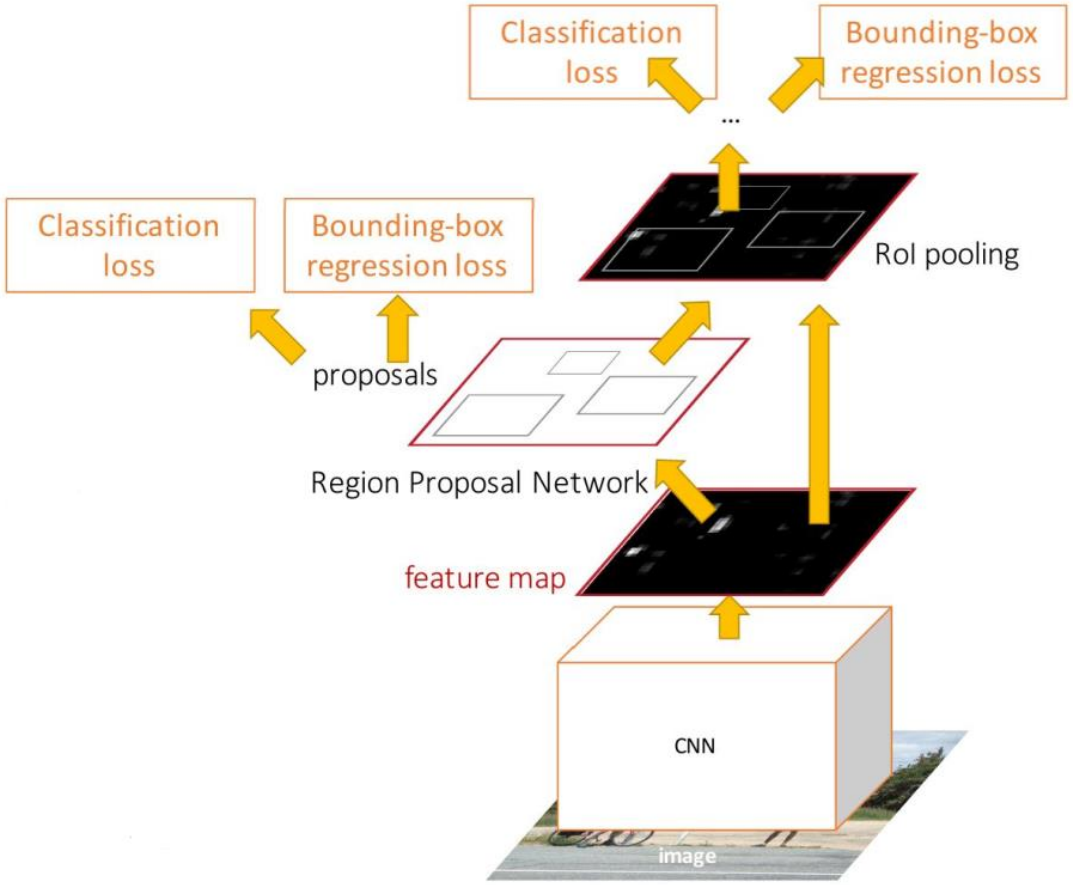
Two Stage

- **Architecture: RPN**



Two Stage

- **Architecture: Faster-RCNN**

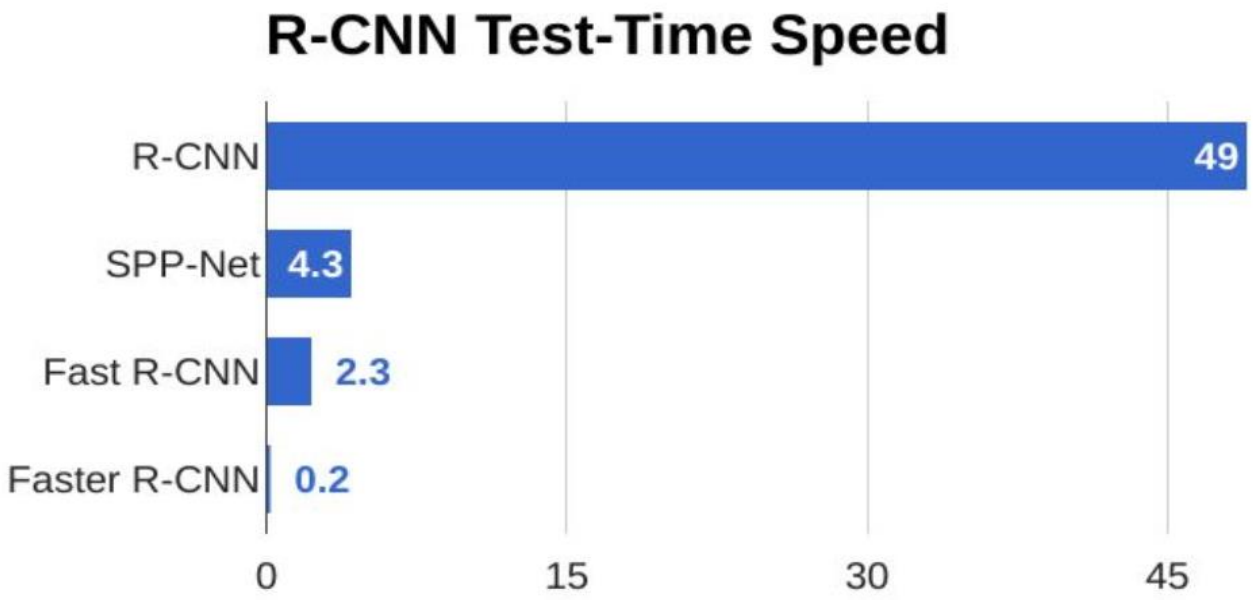


Two Stage

- Pros.
  1. **RPN**: 使用region proposal **network**来代替region proposal method，取得了同等精度下的更高处理速度
  2. **One network**: RPN与后面的CNN layers结合，two stage变成一个network来处理，仅训练一个network极大提高了检测速度
  3. 就**通用**目标检测需求和ImageSet数据集而言，速度和精度已经是**achieve state of the art**
- Cons.
  1. 速度还不能满足高速的实时场景(不如One Stage Method系列快速)
  2. 小目标容易漏检

Two Stage

- 性能对比:



One Stage Method

- Key idea
  1. **Procedure(**扫描图片仅一次，滑动处理窗口，依次做如下处理：)
     - 划分图片为网格
     - 每个网格cell为中心，设置若干个ancor box
     - 实时对每个ancor box做multi_class classification &localisation fine-tune regression

One Stage Method

- Method:

1. YOLO-v1
2. SSD-v1

One Stage Method

- Method:

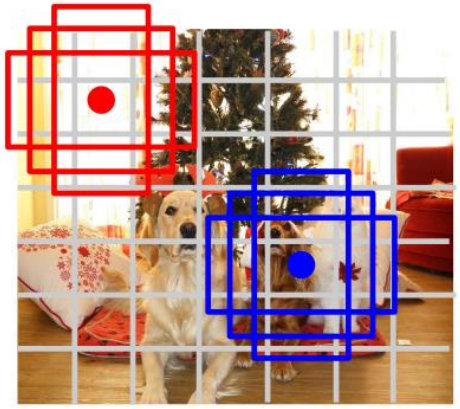
1. YOLO-v1

2. SSD-v1

One Stage

- YOLO-v1

Input image
3 x H x W



Divide image into grid
7 x 7

Image a set of **base boxes** centered at each grid cell
Here B = 3

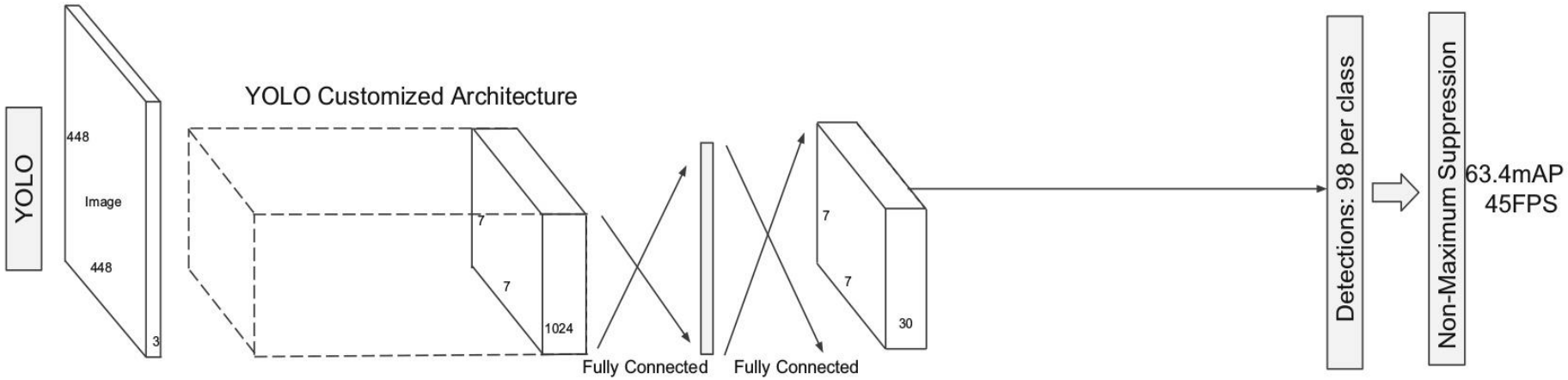
Within each grid cell:
- Regress from each of the B base boxes to a final box with 5 numbers: (dx, dy, dh, dw, confidence)
- Predict scores for each of C classes (including background as a class)

Output:
7 x 7 x (5 * B + C)

Redmon et al, "You Only Look Once: Unified, Real-Time Object Detection", CVPR 2016
Liu et al, "SSD: Single-Shot MultiBox Detector", ECCV 2016

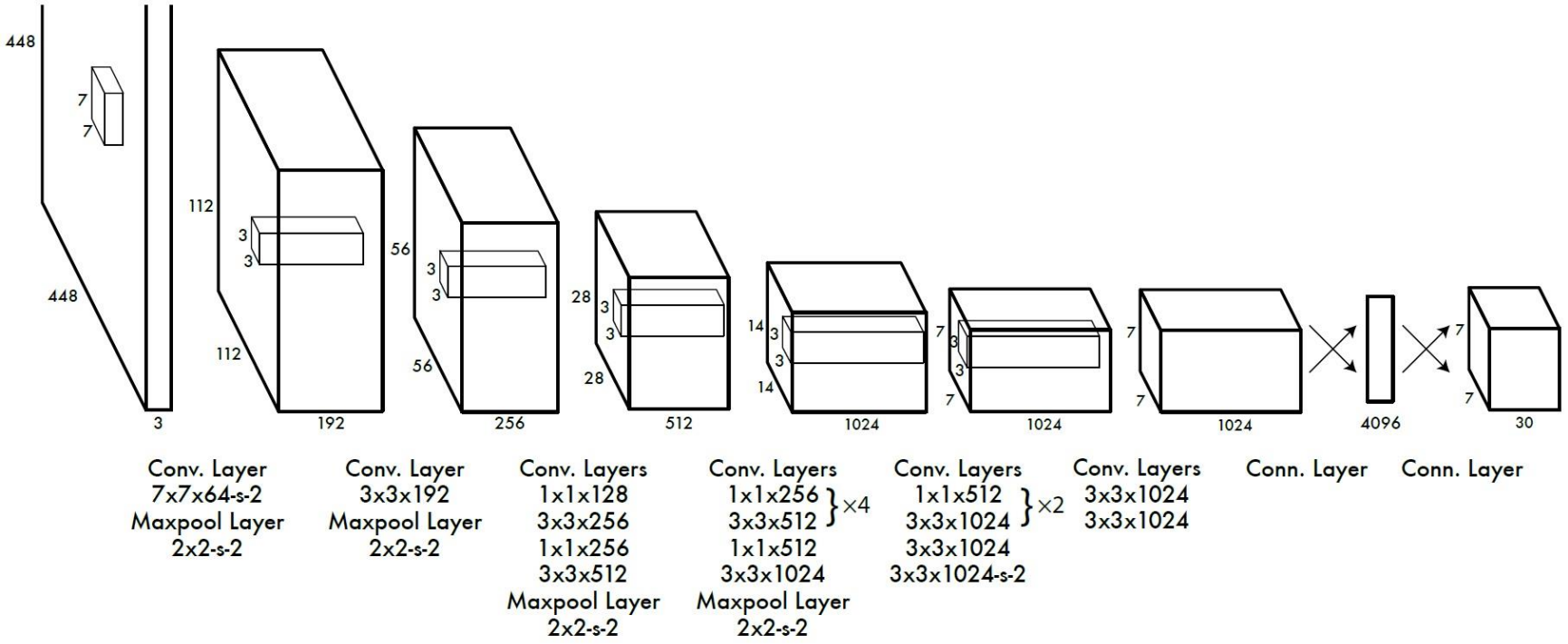
One Stage

- YOLO-v1



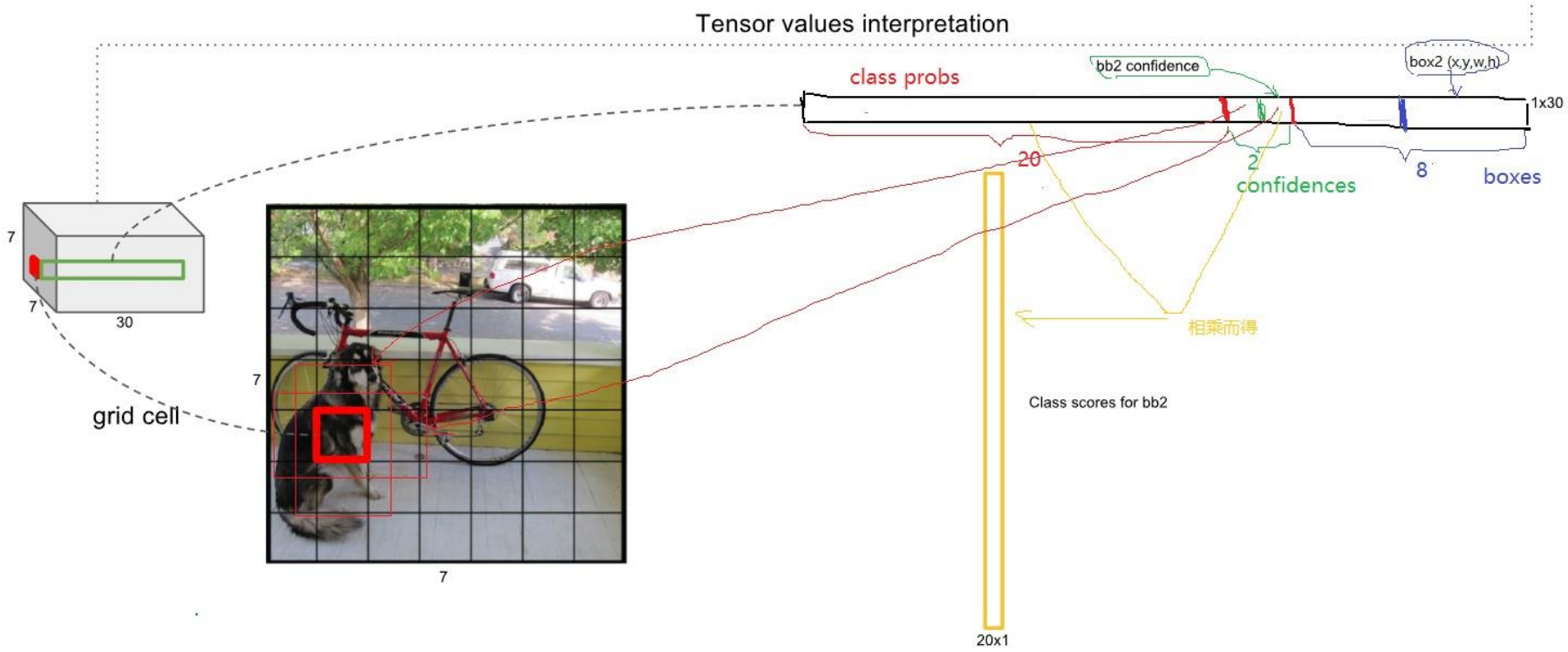
One Stage

- YOLO-v1



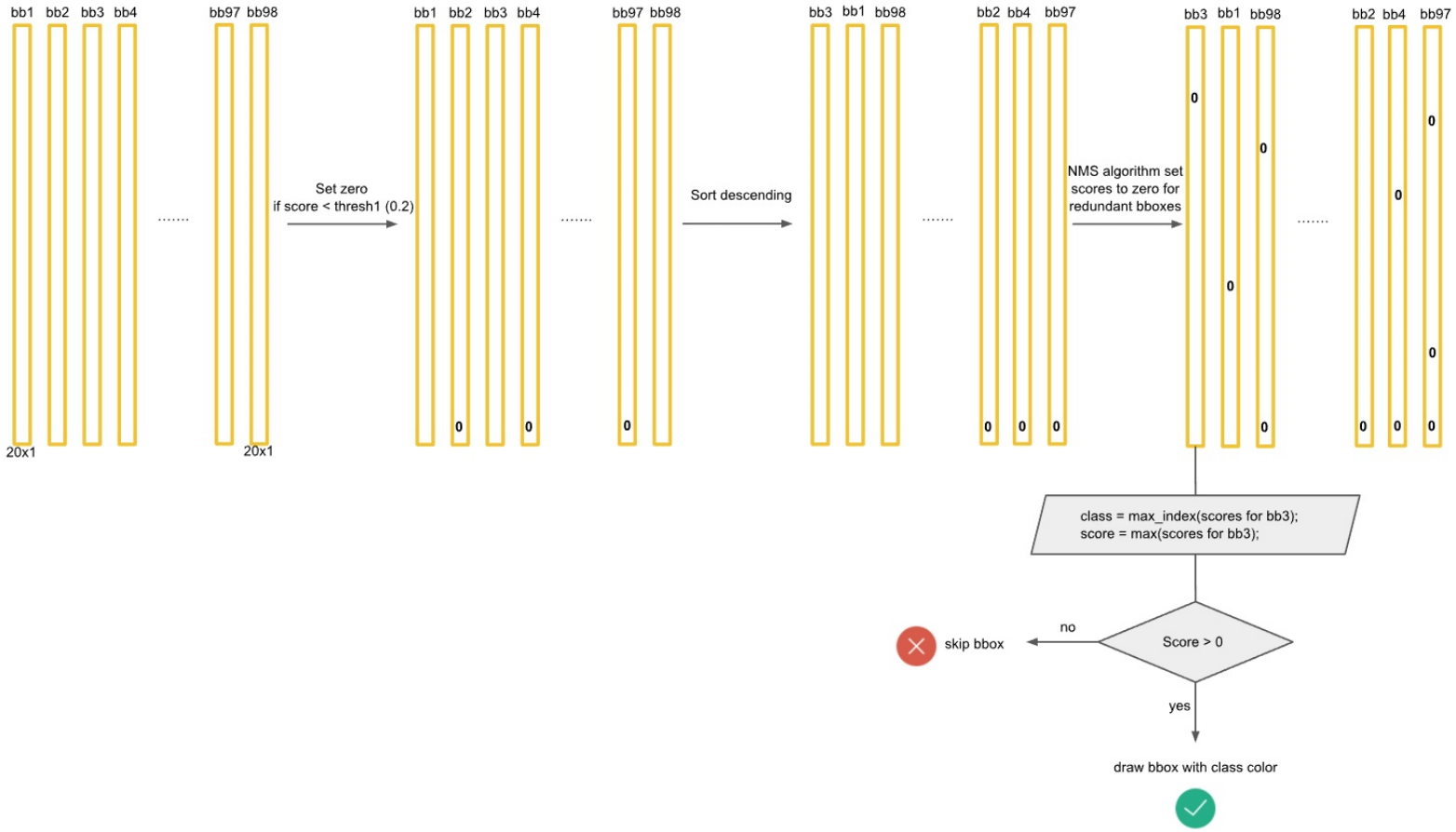
One Stage

- YOLO-v1



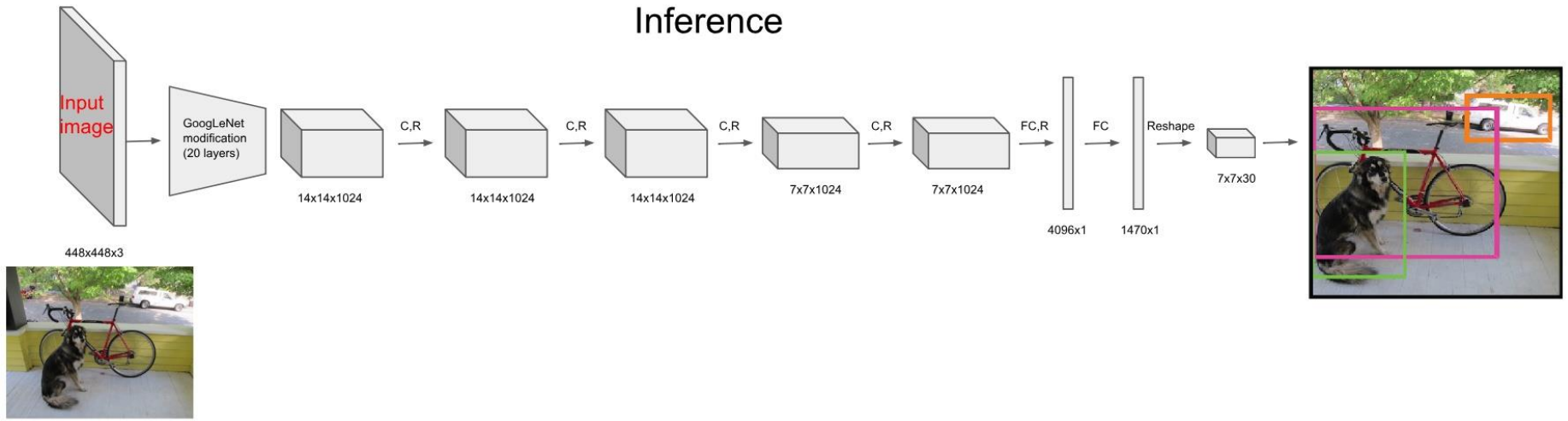
One Stage

- YOLO-v1



One Stage

- YOLO-v1



One Stage

- YOLO-v1
  1. 在YOLO网络中，首先通过一组CNN提取feature maps
  2. 然后通过最后一个全连接FC层生成SxSx(5*B+C)=7x7x(5*2+20)=1470长的向量
  3. 再把1470向量reshape成SxSx(5*B+C)=7x7x30形状的多维矩阵
  4. 通过解析多维矩阵获得Detection bounding box + Confidence
  5. 最后对Detection bounding box + Confidence进行Non maximum suppression获得输出

One Stage

- Pros.
  1. 首次用回归模型解决了object detection
  2. 相比faster-rcnn，实现了实时检测，mAP却没有太大损失
- Cons.
  1. 分类准确率较高，但是定位准确率很低(相比faster-RCNN)

One Stage

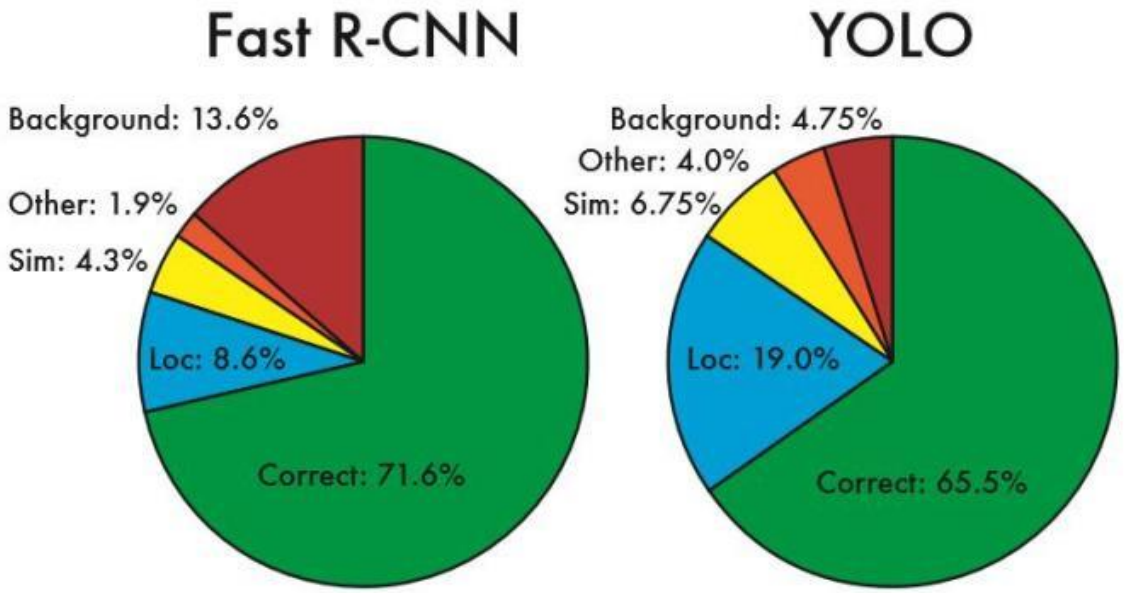


Figure 4: Error Analysis: Fast R-CNN vs. YOLO These charts show the percentage of localization and background errors in the top N detections for various categories (N = # objects in that category).

One Stage Method

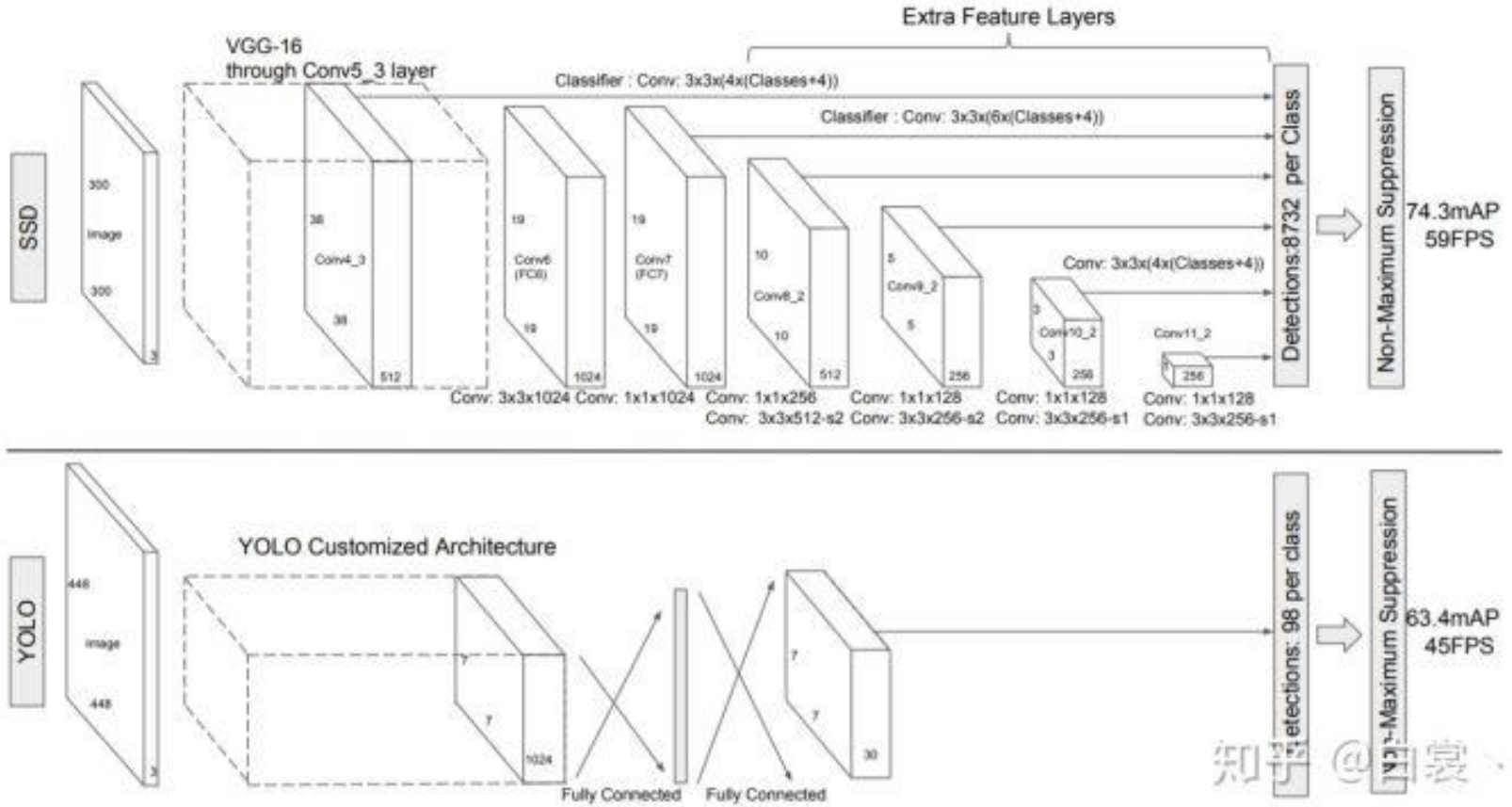
- Method:

1. YOLO-v1

2. SSD-v1

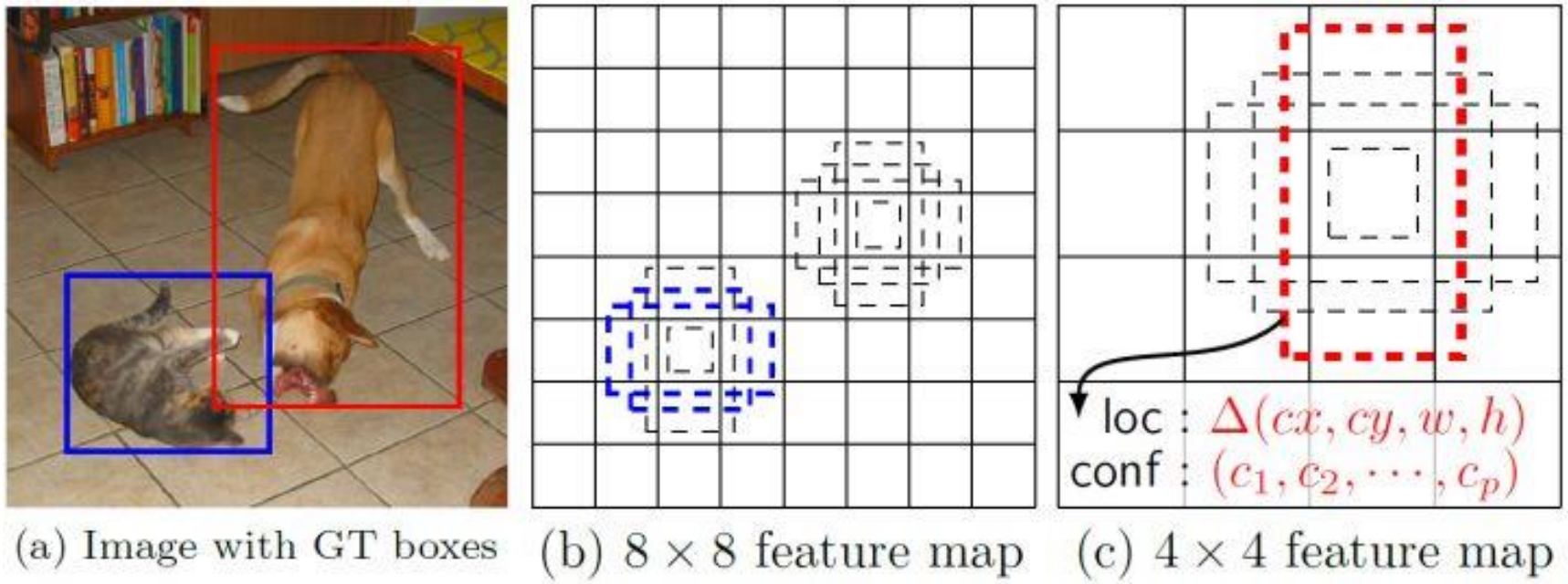
One Stage

- SSD-v1



One Stage

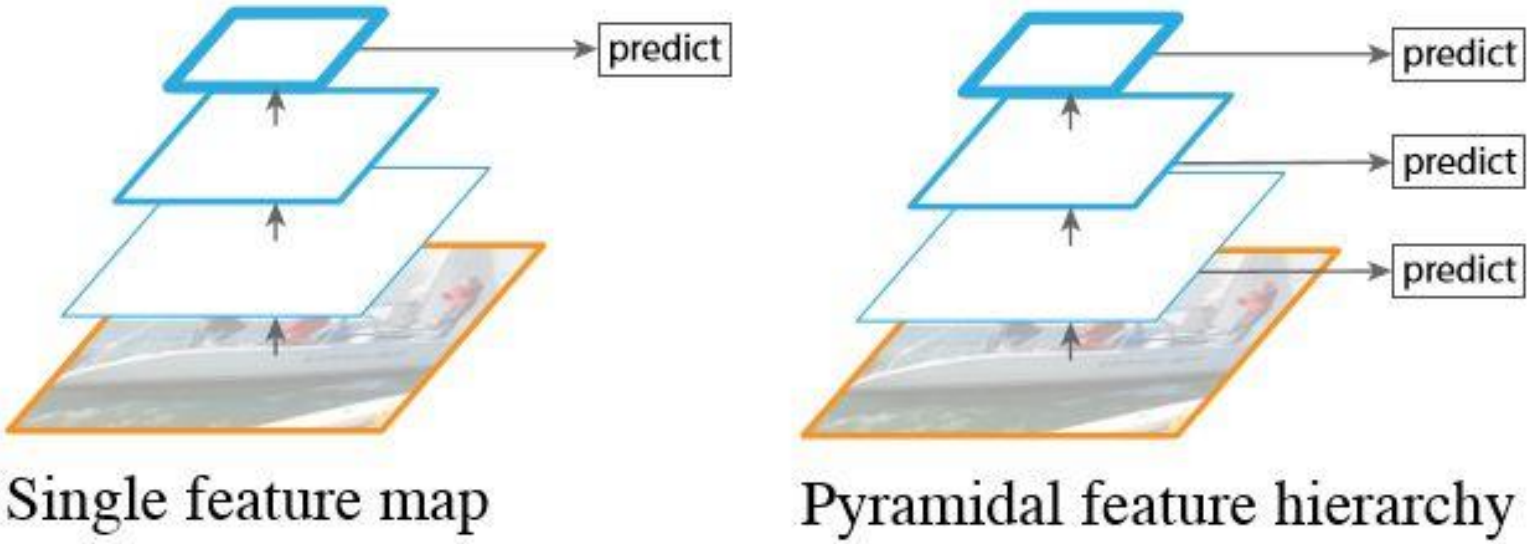
- SSD-v1



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

One Stage

- SSD-v1



Single feature map

Pyramidal feature hierarchy

One Stage

- Pros.
  1. Feature Pyramid embedding: friendly to small object
  2. More effective Ancbox: lower localisation error

  (compared with YOLO-v1)

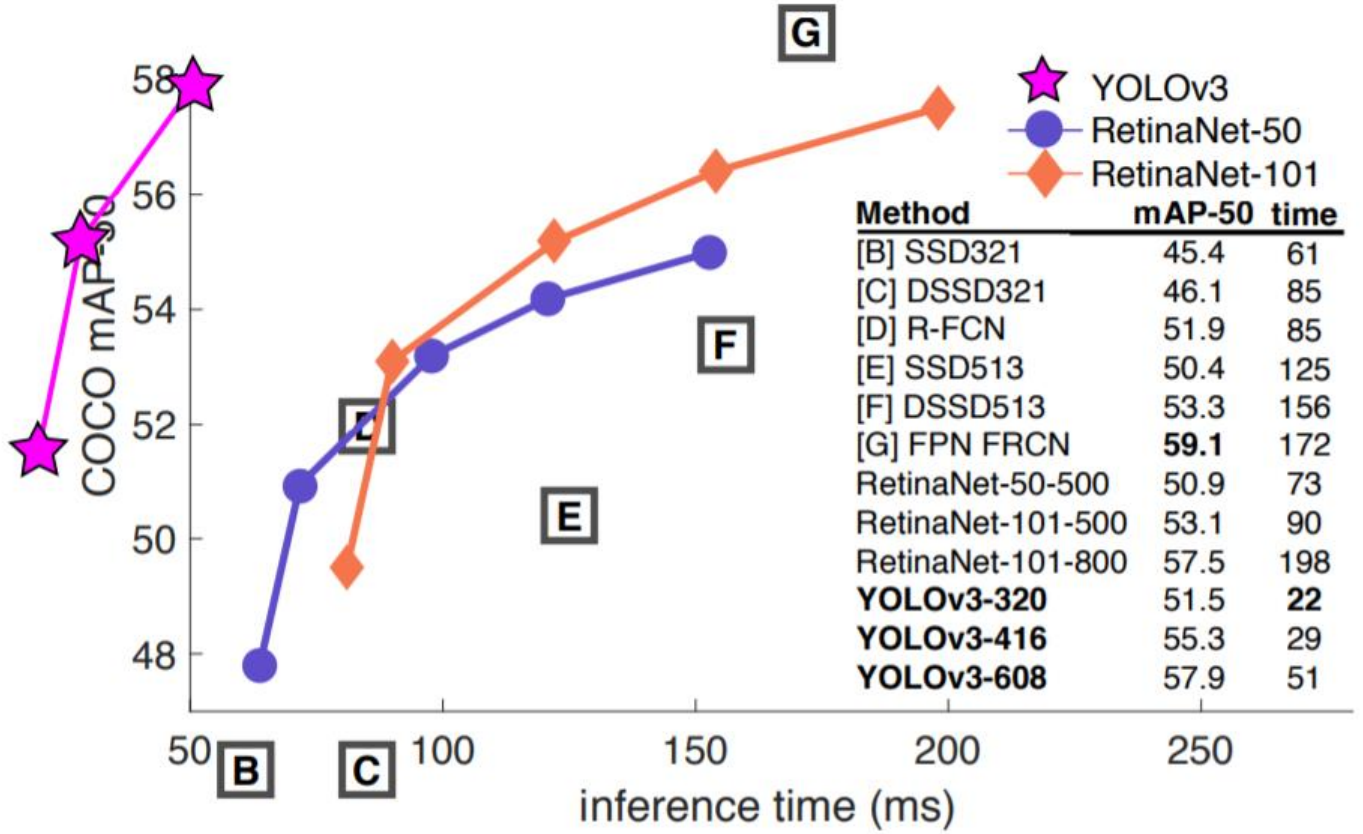
- Cons.
  1. Speed & precison

Sumary

- Two stage vs. One stage:
  - 一般而言，two stage方法由于**第二次精修**检测框的位置而具有**更高的检测精度**；one stage方法由于选取候选框与定位**同时**进行而具有**更高的检测速度**
  - 目前还一个趋势：两类方法**混合互补,** e.g. YOLO-v3, FPN, etc

Sumary

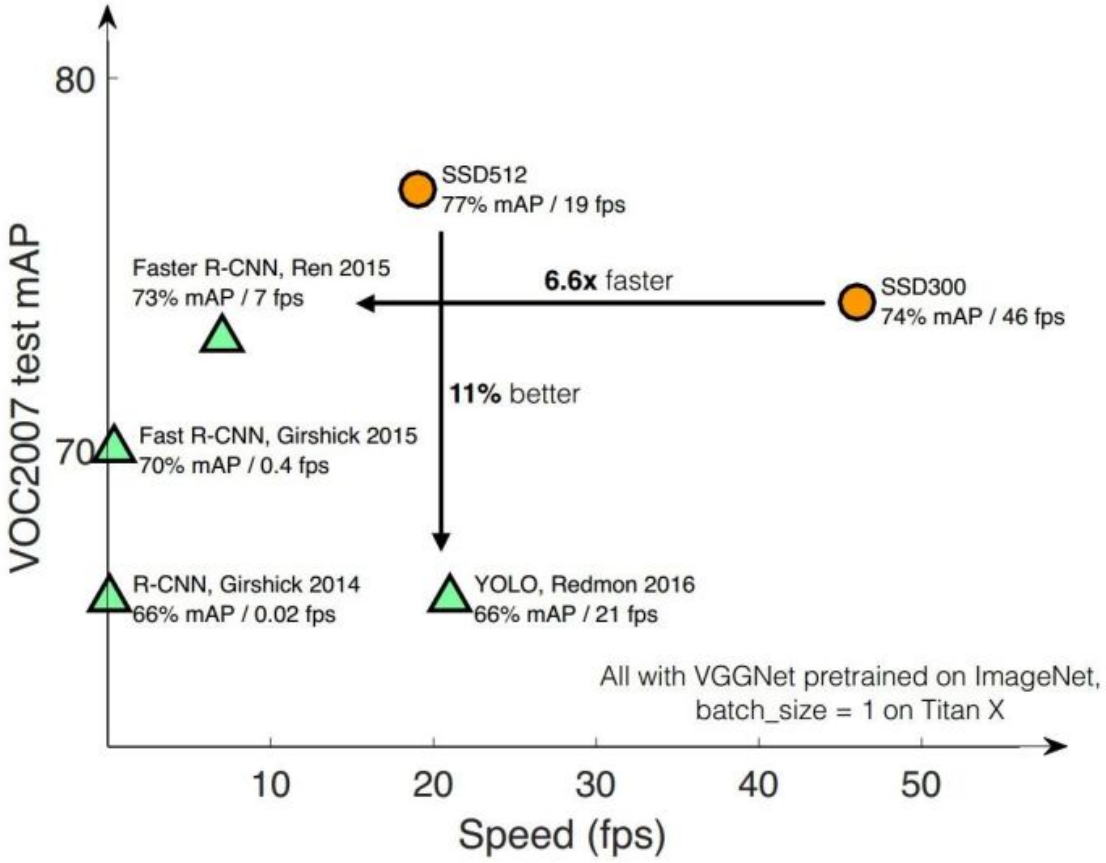
- Two stage vs. One stage:



Method	mAP-50	time
[B] SSD321	45.4	61
[C] DSSD321	46.1	85
[D] R-FCN	51.9	85
[E] SSD513	50.4	125
[F] DSSD513	53.3	156
[G] FPN FRCN	**59.1**	172
RetinaNet-50-500	50.9	73
RetinaNet-101-500	53.1	90
RetinaNet-101-800	57.5	198
YOLOv3-320	51.5	**22**
YOLOv3-416	55.3	29
YOLOv3-608	57.9	51

Sumary

- Two stage vs. One stage:



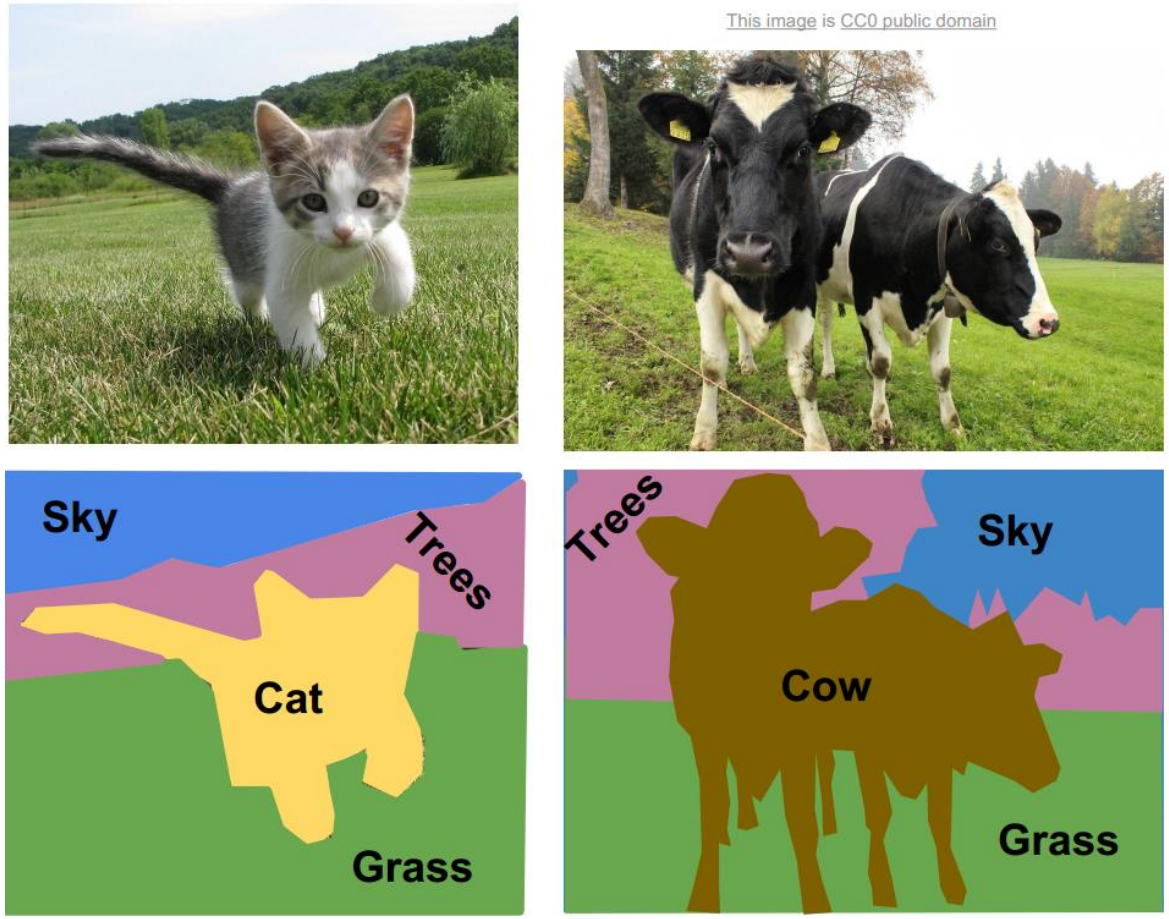
Outline

- Brief to CV
- Classification
- Localisation
- Object Detection
- Segmentation

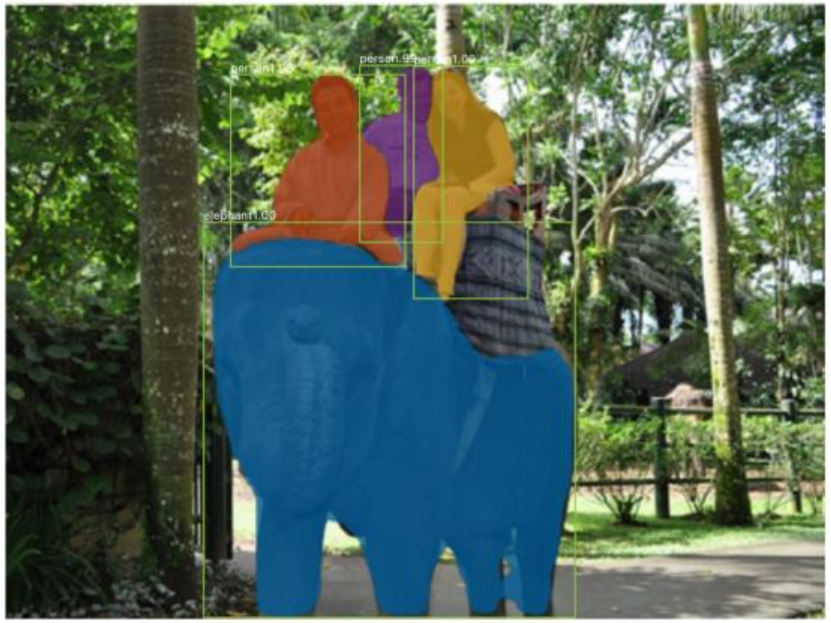
Task description

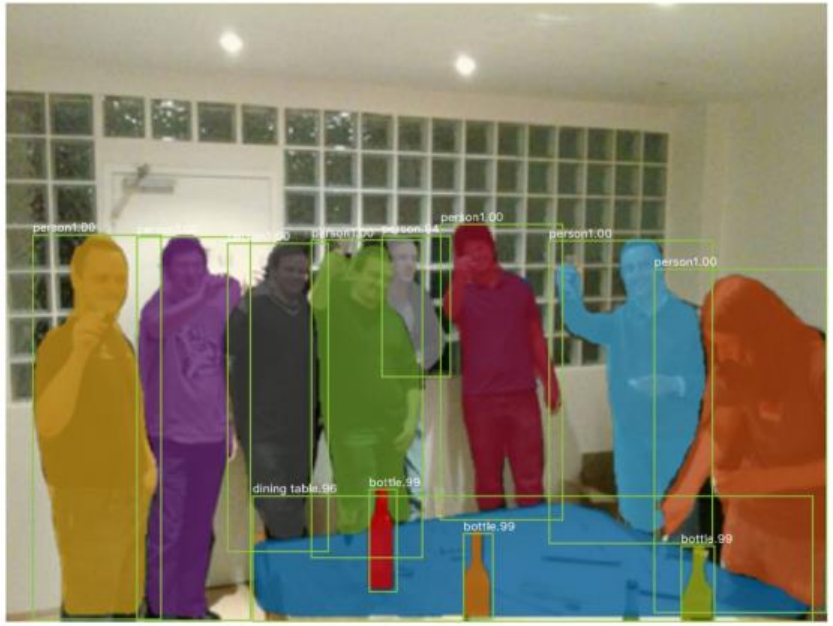
- **Definition：** 给定一张图片，对这张图片上你感兴趣的每一个pixel预测一个category,可分为两类:
  1. **Semantic Segmentation**: 给每个pixel预测category，可以粗糙地理解为 Classication on each pixel of a image
  2. **Instance Segmentation**: 可以粗糙地理解为：Objecet Detection & Semantic Segementation over each detected object

Task description



Task description





Methods

- 几个核心思想：
  - **Slide window**： Farabet et al, TPAMI 2013 & Pinheiro , ICML 2014
  - **Fully Convolutional**： Fully Convolutional Networks for Semantic Segmentation, CVPR 2015
  - **Deconvolution**： Learning Deconvolution Network for Semantic Segmentation, ICCV 2015

Thank you !

Q&A