Deep Learning I

8. 计算机视觉II

目标检测算法、语义分割、风格迁移

WU Xiaokun 吴晓堃

xkun.wu [at] gmail

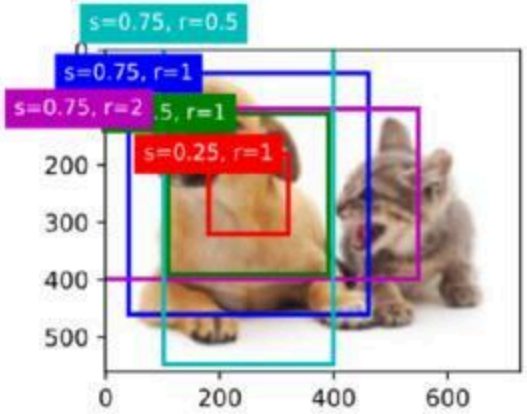
2021/04/18

单发多框检测（SSD）

生成锚框

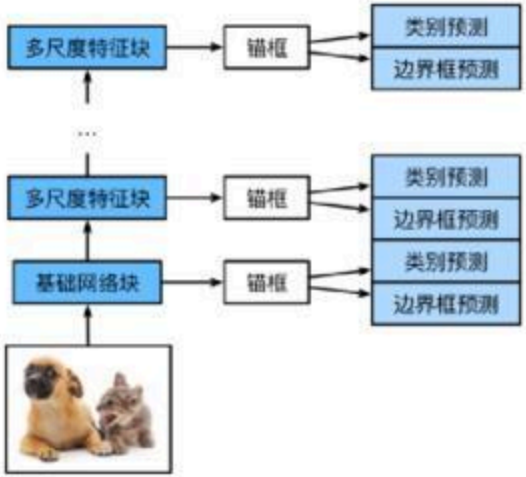
[Liu 2016] 以**每个像素**为锚点、多尺度检测

- 图宽w、高h，缩放比s，宽高比r
  - 锚框：宽$ws\sqrt{r}$、高$hs/\sqrt{r}$
- 实际计算中缩放比、宽高比取有限的一系列值
  - $s \in \{s_1, .., s_n\}, r \in \{r_1, .., r_m\}$
- 实践中只考虑包含s_1或r_1的$m+n-1$个锚框
  - $(s_1, r_1), .., (s_1, r_m), (s_2, r_1), .., (s_n, r_1)$



SSD 模型：架构

- 基础网络：**提取特征**
- 多尺度卷积模块：**减半特征图**

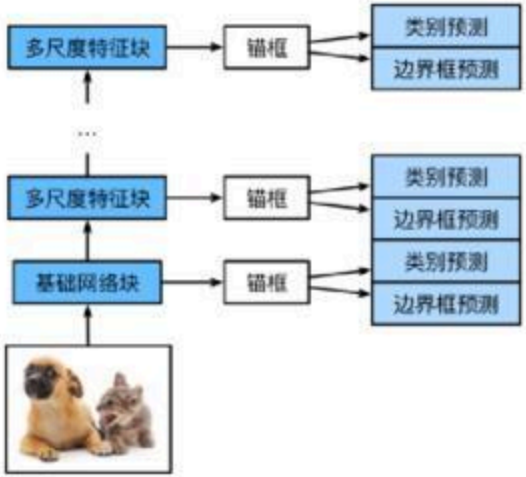


SSD 模型：锚框

- 基础网络：**提取特征**
- 多尺度卷积模块：**减半特征图**

每段都生成锚框：预测类别、边界框

- 顶部：感受野大，拟合大物体
- 底部：感受野小，拟合小物体



SSD 模型：类别预测层

假设有q个目标类别：总共$q+1$个**锚框类别**，0类是背景

- 每个像素生成a个**锚框**：总共hwa个
- 全连接参数过多？使用卷积层

SSD 模型：类别预测层

假设有q个目标类别：总共$q+1$个**锚框类别**，0类是背景

- 每个像素生成a个**锚框**：总共hwa个
- 全连接参数过多？使用卷积层

类别预测层：**保持输入宽、高**的卷积层

- 二维空间坐标保持一致
- 每个通道：锚框、类别预测的**组合**，总共$a(q+1)$个
  - 通道$i(q+1)+j$：锚框索引i、类别索引j

SSD 模型：类别预测层

假设有q个目标类别：总共$q+1$个**锚框类别**，0类是背景

- 每个像素生成a个**锚框**：总共hwa个
- 全连接参数过多？使用卷积层

类别预测层：**保持输入宽、高**的卷积层

- 二维空间坐标保持一致
- 每个通道：锚框、类别预测的**组合**，总共$a(q+1)$个
  - 通道$i(q+1)+j$：锚框索引i、类别索引j
- 具体实现：填充1的3x3卷积

SSD 模型：连结多尺度预测

边界框预测层：$4a$个通道，即4个偏移量

SSD 模型：连结多尺度预测

边界框预测层：$4a$个通道，即4个偏移量

多尺度预测：锚框数量、形状不同

- 回顾张量形状：（批量大小，通道数，高度，宽度）
  - 不同层：通道数、高度、宽度都不同

SSD 模型：连结多尺度预测

边界框预测层：$4a$个通道，即4个偏移量

多尺度预测：锚框数量、形状不同

- 回顾张量形状：（批量大小，通道数，高度，宽度）
  - 不同层：通道数、高度、宽度都不同
- 转换成**一致格式**：（批量大小，高 x 宽 x 通道数）
  - 批量大小相同，在维度1上拼接

SSD 模型：高和宽减半块

多尺度预测实现：宽、高减半

- 模块与 VGG 类似：填充1的3x3卷积层 + 步幅2的最大池化
- **特征图减半**：感受野边长加倍

SSD 模型：高和宽减半块

多尺度预测实现：宽、高减半

- 模块与 VGG 类似：填充1的3x3卷积层 + 步幅2的最大池化
- **特征图减半**：感受野边长加倍

基础网络实现：提取特征

- 串联3个高和宽减半块
  - 例如256x256图像输出：32x32（$256/2^3 = 32$）

SSD 模型：高和宽减半块

多尺度预测实现：宽、高减半

- 模块与 VGG 类似：填充1的3x3卷积层 + 步幅2的最大池化
- **特征图减半**：感受野边长加倍

基础网络实现：提取特征

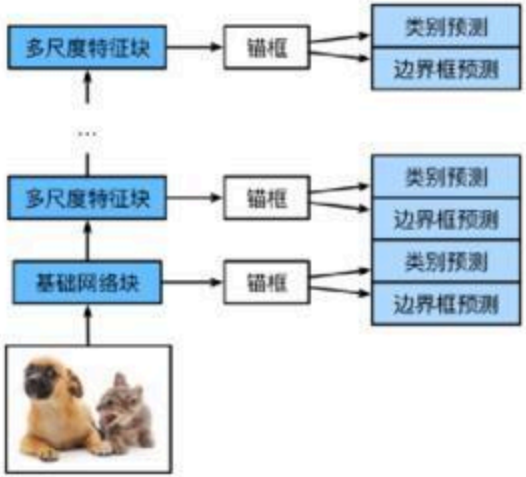
- 串联3个高和宽减半块
  - 例如256x256图像输出：32x32（$256/2^3 = 32$）
- 通道数逐步翻倍：$[3, 16, 32, 64]$

SSD 完整模型

五个模块组成

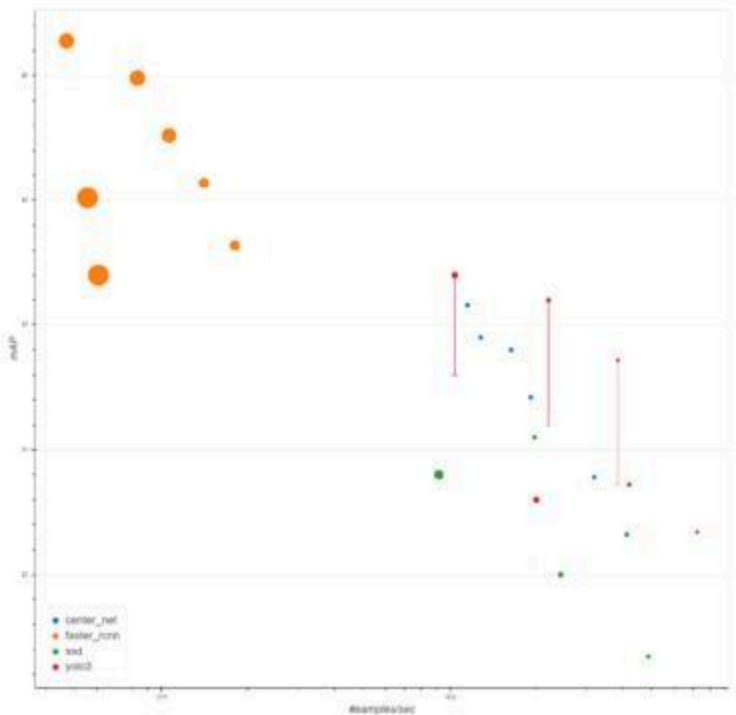
- 基础网络：**提取特征**
- 多尺度卷积模块：**减半特征图**

每段都生成锚框：预测类别、边界框



GluonCV Model Zoo：目标检测

cv.gluon.ai/model_zoo/detection



实验：SSD

小结：SSD

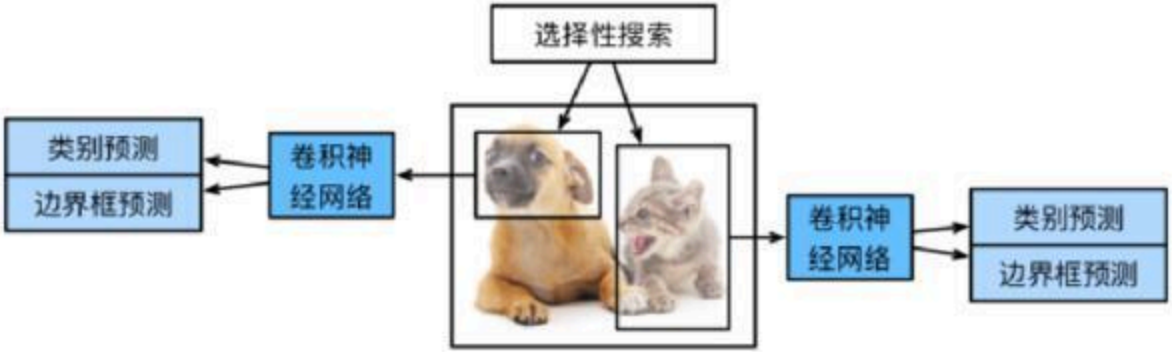
- SSD：以每个像素为锚点、多尺度检测
  - 单个神经网络
  - 多个不同尺度的输出阶段
- SSD 可以认为是集成了之前的数据结构、算法（没有按历史顺序安排章节）

区域卷积神经网络（R-CNN）系列

R-CNN

[Girshick 2014] 启发式搜索：选择出**提议区域**（如锚框）

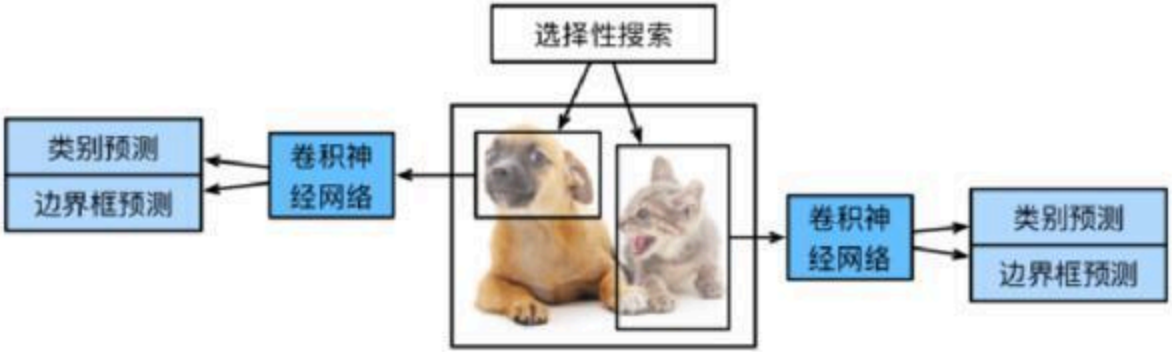
- 使用预训练模型前向传播、提取特征



R-CNN

[Girshick 2014] 启发式搜索：选择出**提议区域**（如锚框）

- 使用预训练模型前向传播、提取特征

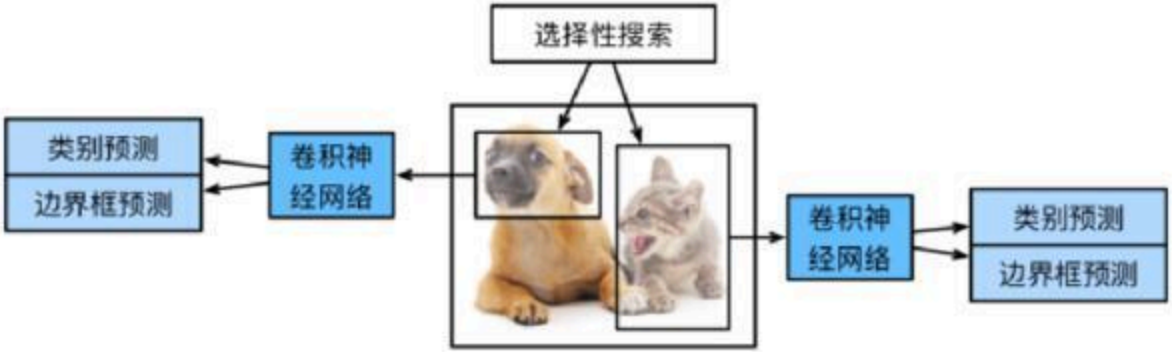


1. 选取**多尺度提议区域**：标注类别、真实边界框
2. 预训练模型：提议区域尺寸变换，前向传播、**提取特征**
3. 训练多个**SVM分类器**：将提议区域作为标注样本，判定是否属于目标类别
4. 训练**回归模型**：预测真实边界框

R-CNN：与SSD比较

[Girshick 2014] 启发式搜索：选择出**提议区域**（如边界框）

- 使用预训练模型前向传播、提取特征



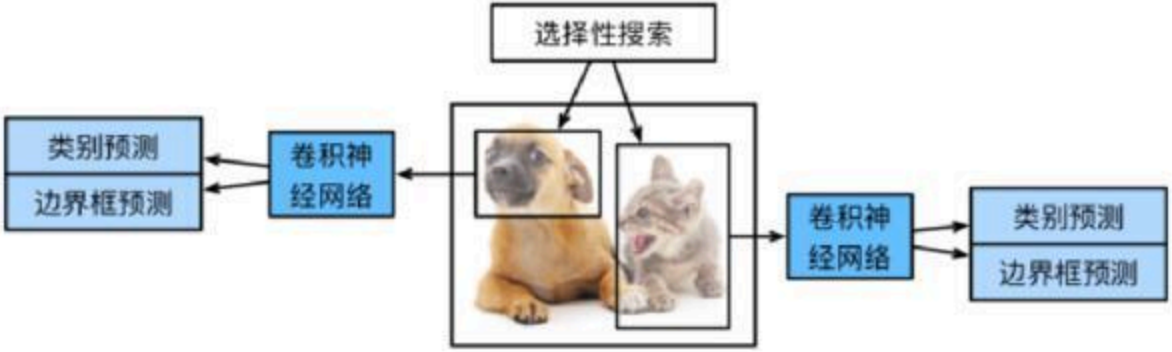
R-CNN (2014) 早于 SSD (2016)

- 锚框：边界框选取、预测的**系统化方法**

R-CNN：与SSD比较

[Girshick 2014] 启发式搜索：选择出**提议区域**（如边界框）

- 使用预训练模型前向传播、提取特征



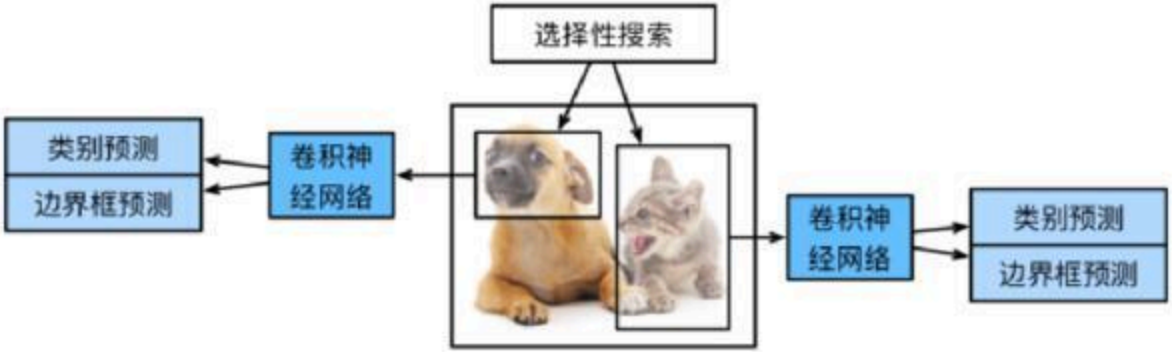
R-CNN (2014) 早于 SSD (2016)

- 锚框：边界框选取、预测的**系统化方法**
- [Uijlings 2013] 选择性搜索：选取规则并不完善
  - 单张图片随机选出上千个提议区域：覆盖率低、计算缓慢

R-CNN：计算性能

[Girshick 2014] 启发式搜索：选择出**提议区域**（如锚框）

- 使用预训练模型前向传播、提取特征



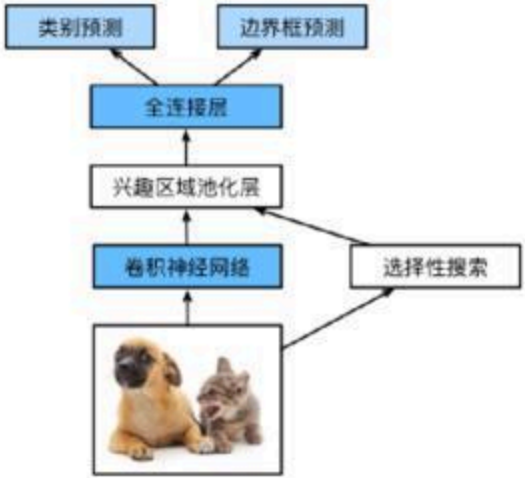
R-CNN主要性能瓶颈：每个提议区域上网络**独立传播**

- 但通常这些区域有大量重叠：本可以共享特征，却导致重复计算

Fast R-CNN

[Girshick 2015] Fast R-CNN：仅在**整张图像**上执行前向传播

- 卷积的输入是整个图像，且**参与训练**
- 从提议区域中提取**形状相同**的特征映射
  - 将特征拼接到一起输出
  - **备选区域池化层 RoI pooling**



Fast R-CNN：RoI pooling

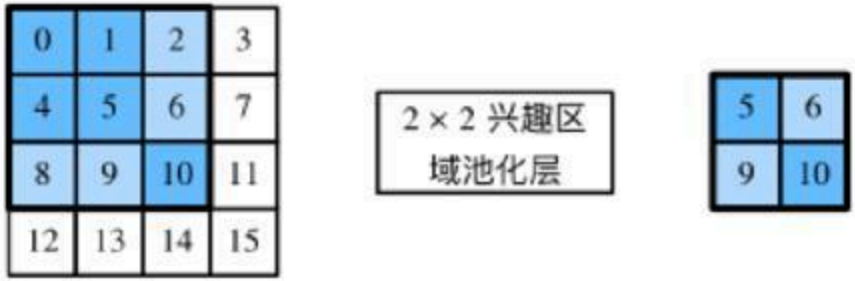
RoI pooling：指定相同的输出形状

- 给定备选区域：均匀分割
- 输出窗口中每个子区域的**最大值**
  - 注意：子区域不重叠、形状不规则，如图中2x1、1x2、1x1

Fast R-CNN：RoI pooling

RoI pooling：指定相同的输出形状

- 给定备选区域：均匀分割
- 输出窗口中每个子区域的**最大值**
  - 注意：子区域不重叠、形状不规则，如图中2x1、1x2、1x1



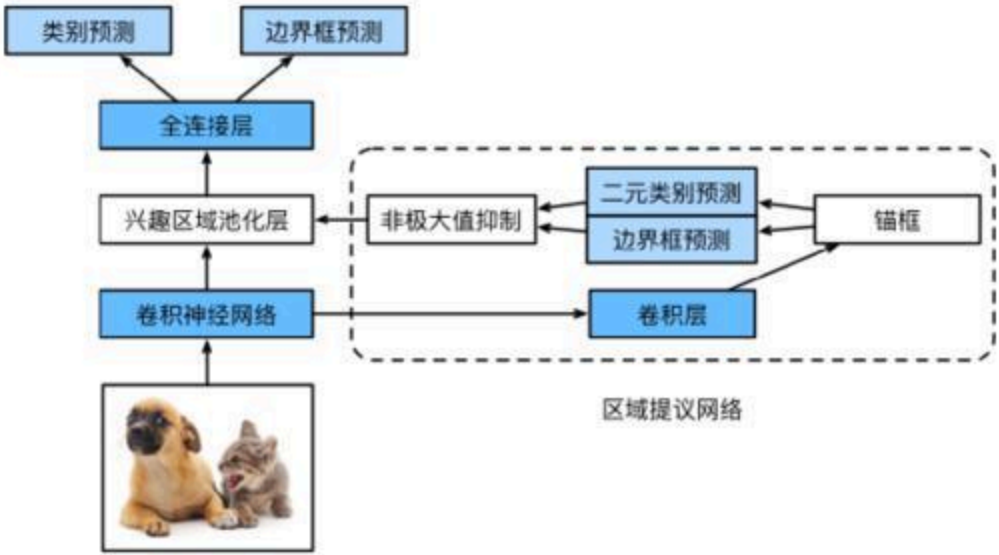
问题：还是不能系统地确定备选区域

- 随机选取的检测精度取决于**提议区域的数量**

Faster R-CNN

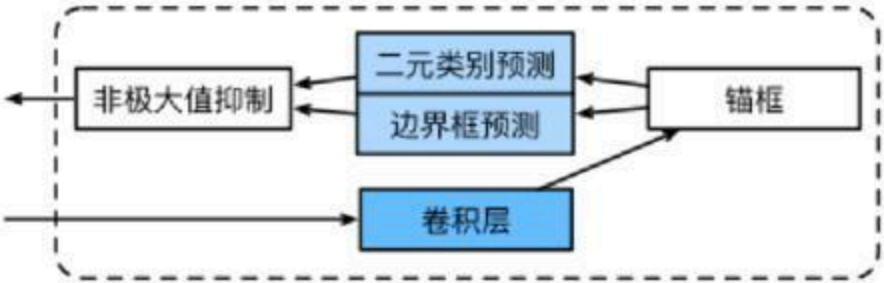
[Ren 2015] Faster R-CNN：将选择性搜索替换为网络学习

- **区域提议 region proposal 网络 (RPN)**：从数据中学习选取规律
  - 输出数量可控，同时可以保证检测精度
  - “深度学习算法很靠谱”已经成为此时（2015年）的主流观点



Faster R-CNN：RPN

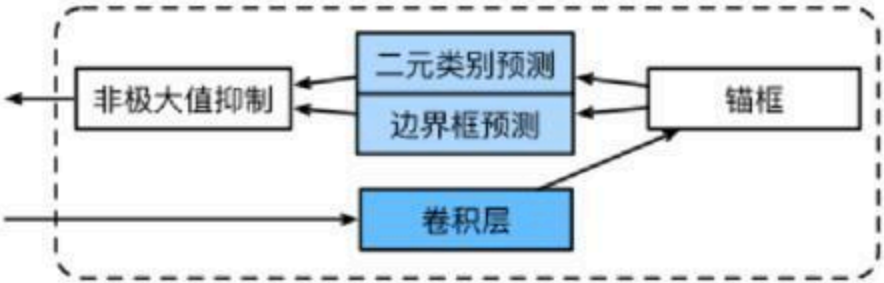
RPN：从数据中学习选取规律



1. 特征图**形状保持不变**：填充1的3x3卷积
2. **生成锚框**：每个像素作为锚点（不同大小、宽高比），并标注
3. **剪裁特征**（锚框范围内）：分别预测类别（目标还是背景）、边界框
4. **NMS**：移除相似的结果，输出提议区域

Faster R-CNN：RPN

RPN：从数据中学习选取规律



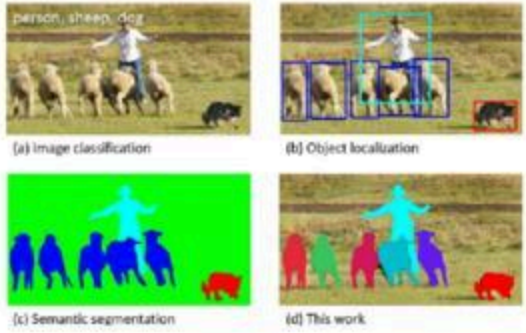
1. 特征图**形状保持不变**：填充1的3x3卷积
2. **生成锚框**：每个像素作为锚点（不同大小、宽高比），并标注
3. **剪裁特征**（锚框范围内）：分别预测类别（目标还是背景）、边界框
4. **NMS**：移除相似的结果，输出提议区域

注意：RPN 随整个模型一起训练

Mask R-CNN

[He 2017] Mask R-CNN：像素级预测

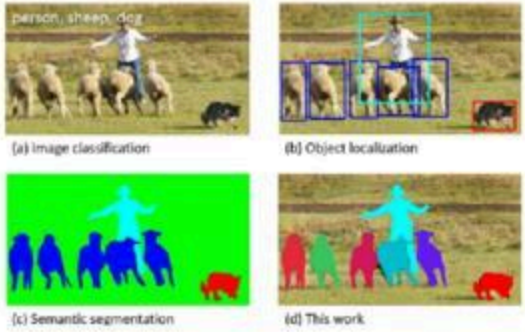
- 要求训练集标注像素级位置信息



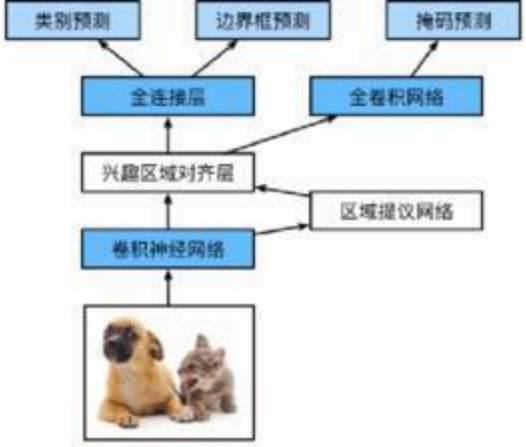
Mask R-CNN

[He 2017] Mask R-CNN：像素级预测

- 要求训练集标注像素级位置信息

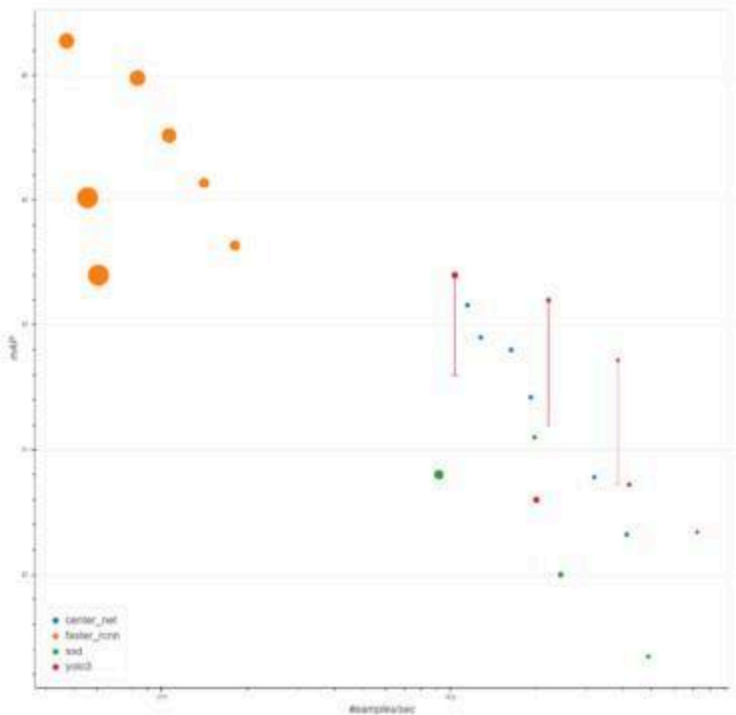


1. 备选区域对齐层：保留特征图的空间信息
2. 额外的**全卷积网络**：像素级预测


类别预测
边界框预测
掩码预测
全连接层
全卷积网络
兴趣区域对齐层
区域提议网络
卷积神经网络

GluonCV Model Zoo：目标检测

cv.gluon.ai/model_zoo/detection



小结：R-CNN

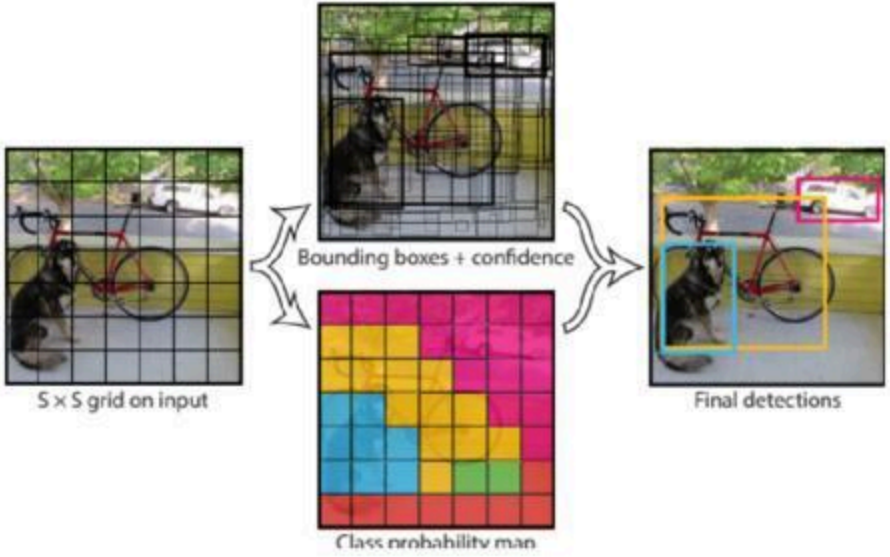
- R-CNN：选取若干提议区域，分别执行前向传播以抽取其特征，然后再预测类别、边界框。
- Fast R-CNN：只对整个图像做前向传播。
  - 备选区域池化层：为不同形状的备选区域抽取相同形状的特征。
- Faster R-CNN：使用参与训练的区域提议网络
  - 可控提议区域数量，仍保证目标检测精度。
  - “深度学习算法很靠谱”已经成为此时（2015年）的主流观点
- Mask R-CNN：引入全卷积网络，像素级预测精度。

你只看一次（YOLO）

YOLO

SSD中锚框大量重复：导致重复计算

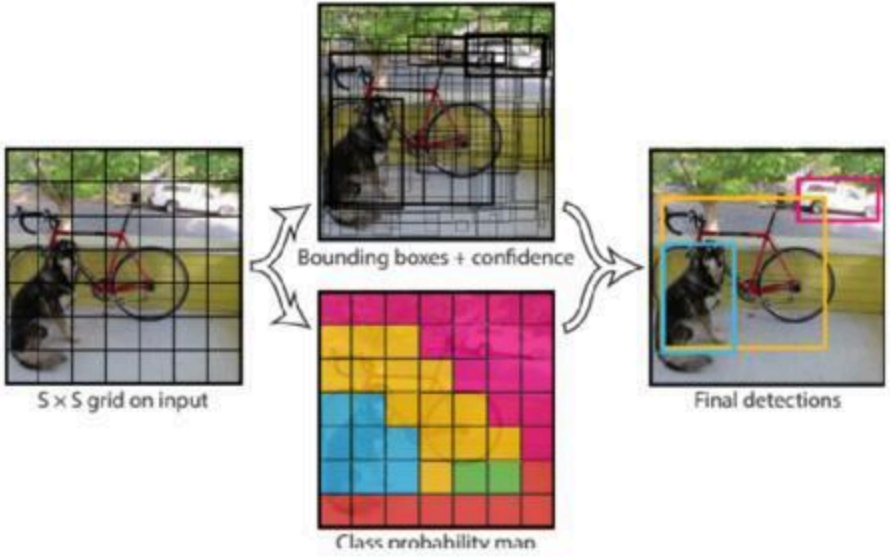
- [Redmon 2016] 将图片均分为锚框，每个锚框分别预测



YOLO

SSD中锚框大量重复：导致重复计算

- [Redmon 2016] 将图片均分为锚框，每个锚框分别预测



- 后续版本（V2，V3，V4）有持续改进

- 但原作者退圈了

语义分割和数据集

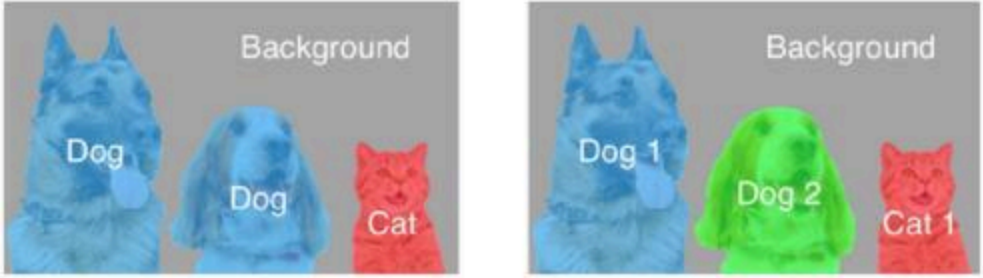
语义分割

将图像分割成**语义**区域：像素级标注、预测

语义分割

将图像分割成**语义**区域：像素级标注、预测

与**实例**分割的区别

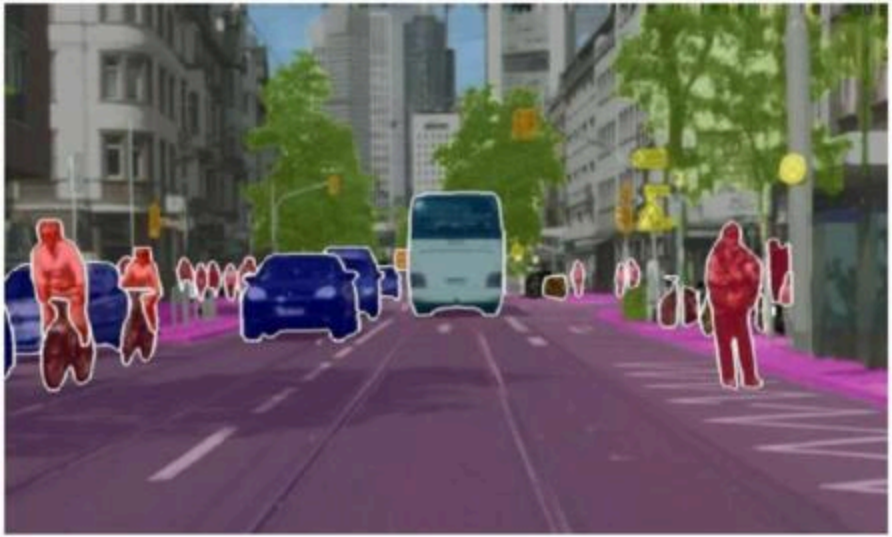


应用：背景虚化

模拟大光圈、小景深效果

应用：自动驾驶

实时像素级物体检测



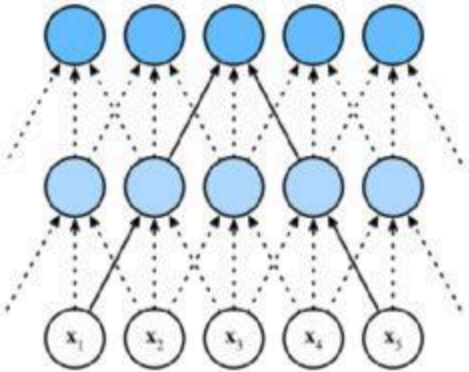
实验：语义分割数据集

转置卷积

特征提取、语义分割

神经网络：逐层提取不同尺寸的特征

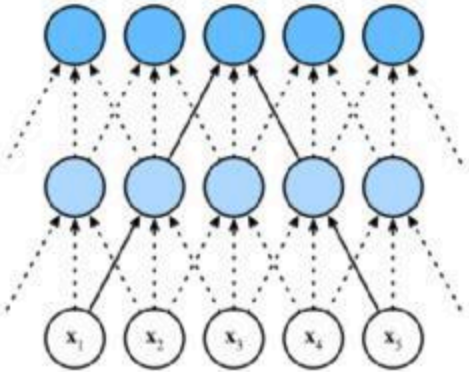
- 不断**对特征图下采样**：深层的“视野”相对变大，“站得高看得远”



特征提取、语义分割

神经网络：逐层提取不同尺寸的特征

- 不断**对特征图下采样**：深层的“视野”相对变大，“站得高看得远”



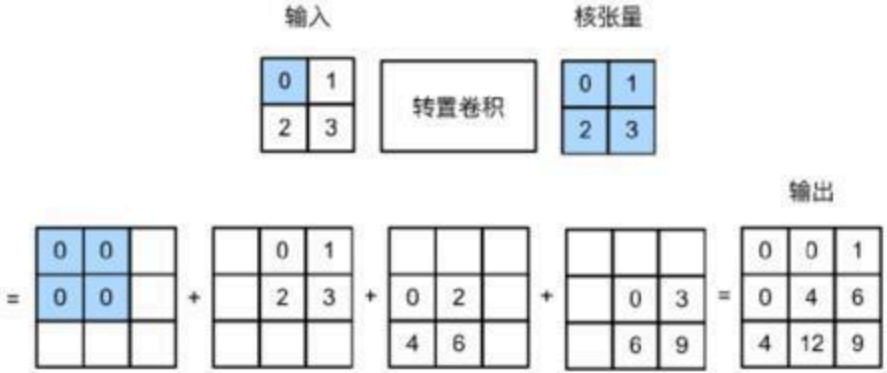
语义分割：输出像素级语义预测

- 如何建立小尺寸特征图到**图像空间**的映射?
  - 需要某种上采样算法：逆转特征图下采样过程

转置卷积操作

转置卷积：本质上是**上采样**

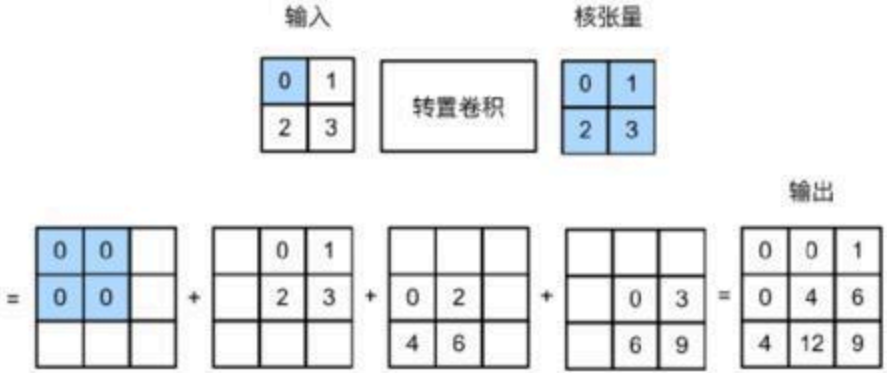
- 通过卷积核逐点“广播”输入元素：故产生更大的输出



转置卷积操作

转置卷积：本质上是**上采样**

- 通过卷积核逐点“广播”输入元素：故产生更大的输出



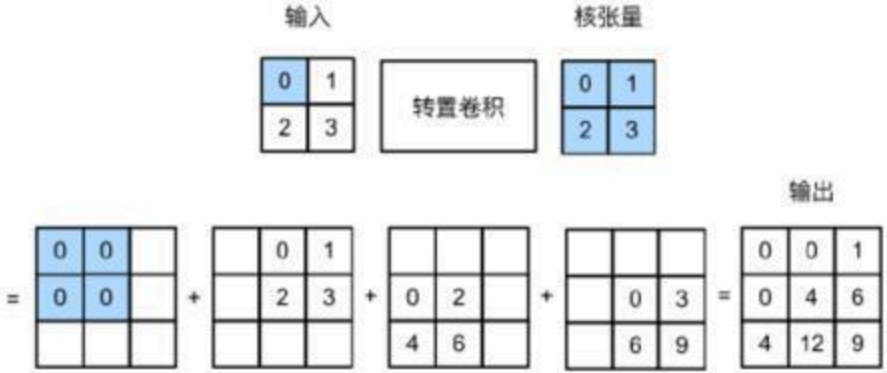
- 中间张量：替换一部分为卷积结果

$$Y[i:i+h, j:j+w] \mathrel{+}= X[i,j] * K$$

转置卷积操作

转置卷积：本质上是**上采样**

- 通过卷积核逐点“广播”输入元素：故产生更大的输出



- 中间张量：替换一部分为卷积结果

$$Y[i:i+h, j:j+w] \mathrel{+}= X[i,j] * K$$

转置卷积：填充、步幅和多通道

转置卷积中，**填充用于输出**（常规卷积中用于输入）

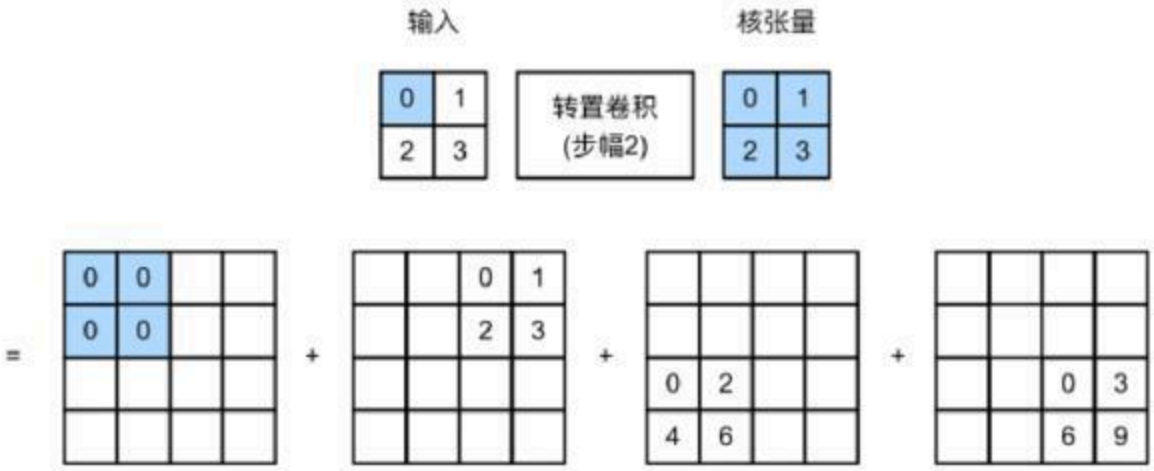
- 例如：填充1的输出将**删减**第一、最后的行、列

转置卷积：填充、步幅和多通道

转置卷积中，**填充用于输出**（常规卷积中用于输入）

- 例如：填充1的输出将**删减**第一、最后的行、列

步幅用于中间结果（输出），而非输入

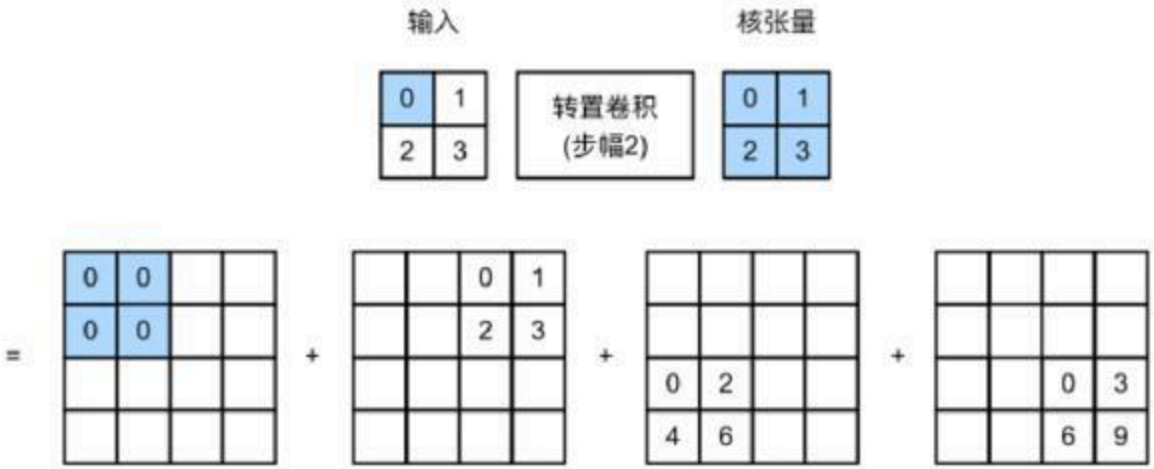


转置卷积：填充、步幅和多通道

转置卷积中，**填充用于输出**（常规卷积中用于输入）

- 例如：填充1的输出将**删减**第一、最后的行、列

步幅用于中间结果（输出），而非输入



多通道：与常规卷积相同

实验：转置卷积

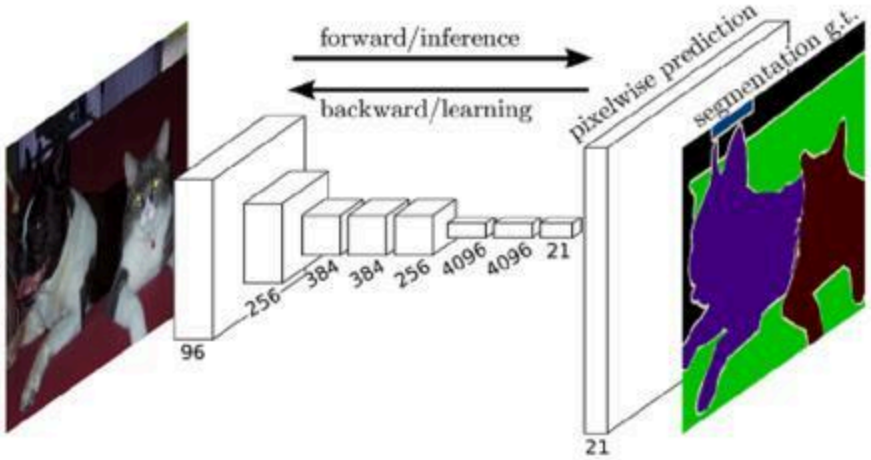
小结：转置卷积

- 转置卷积：本质是上采样
  - 命名：将卷积转换成矩阵乘积后，转置矩阵恰好对应转置卷积
- 填充用于输出，删减行、列；步幅用于中间结果；多通道与常规卷积相同

全卷积网络 (FCN)

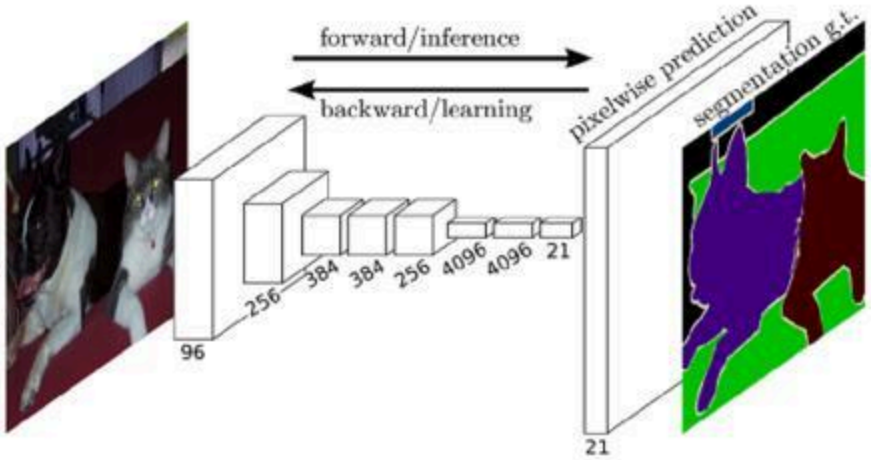
FCN：语义分割

[Long 2015] **全卷积网络 fully convolutional network (FCN)**：只有卷积层



FCN：语义分割

[Long 2015] **全卷积网络 fully convolutional network (FCN)**：只有卷积层



语义分割：对每个像素分类

- 将特征图尺寸**变换回输入尺寸**：转置卷积
- 每个通道对应一个类别：输出预测概率

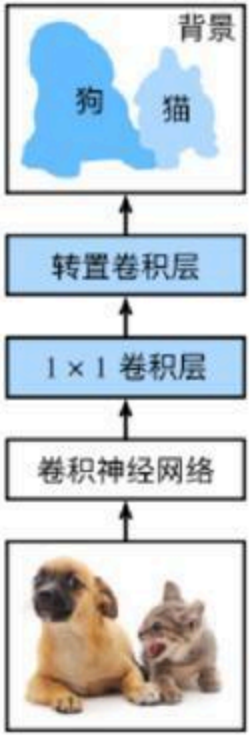
FCN：架构

三个主要阶段

1. 卷积网络：提取特征图
2. 1x1卷积：通道数调整为类别数
3. 转置卷积：变换回输入尺寸

转置卷积：假设需要放大s倍**恢复尺寸**

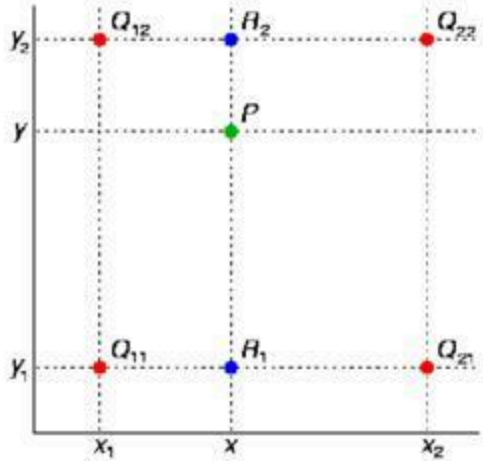
- 步幅s，填充$s/2$，卷积核$2s$



FCN：初始化转置卷积层

坐标按尺寸比例放大，映射回输入图像空间

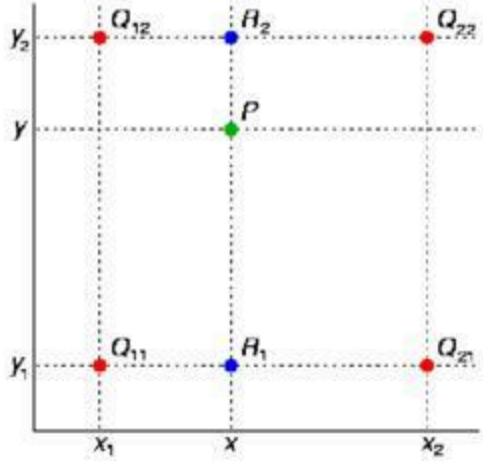
- 大概率落在相邻**四个像素之间**



FCN：初始化转置卷积层

坐标按尺寸比例放大，映射回输入图像空间

- 大概率落在相邻**四个像素之间**



借鉴图像处理中**上采样**方法：**双线性插值**

- 二维平面4个角点上数值的加权组合

- 权重：水平、垂直方向的相对距离

实验：FCN

小结：FCN

- 全卷积网络 (FCN)：利用转置卷积恢复图像尺寸
  - 用于像素级预测：使用1x1卷积调整通道数（与类别数相同）
- 初始化转置卷积：双线性插值

风格迁移

风格迁移

[Gatys 2016] 将样例图片风格迁移到内容图像上

Content image

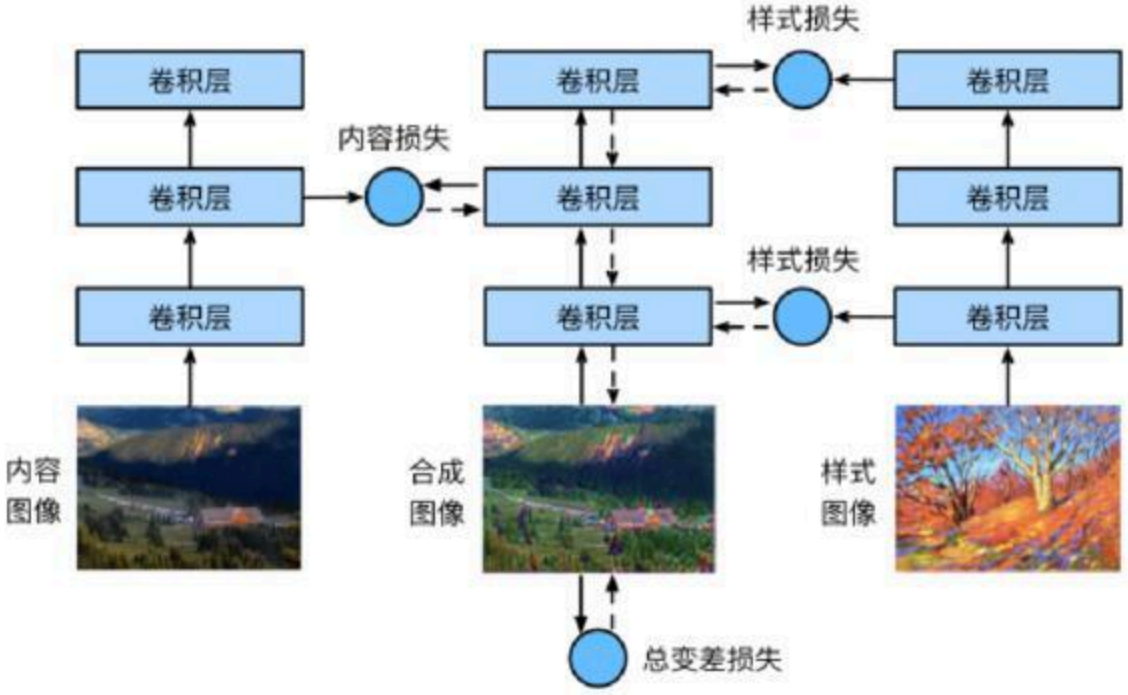
Style image

Synthesized image

风格迁移：模型

合成图像是唯一需要更新的**模型参数**

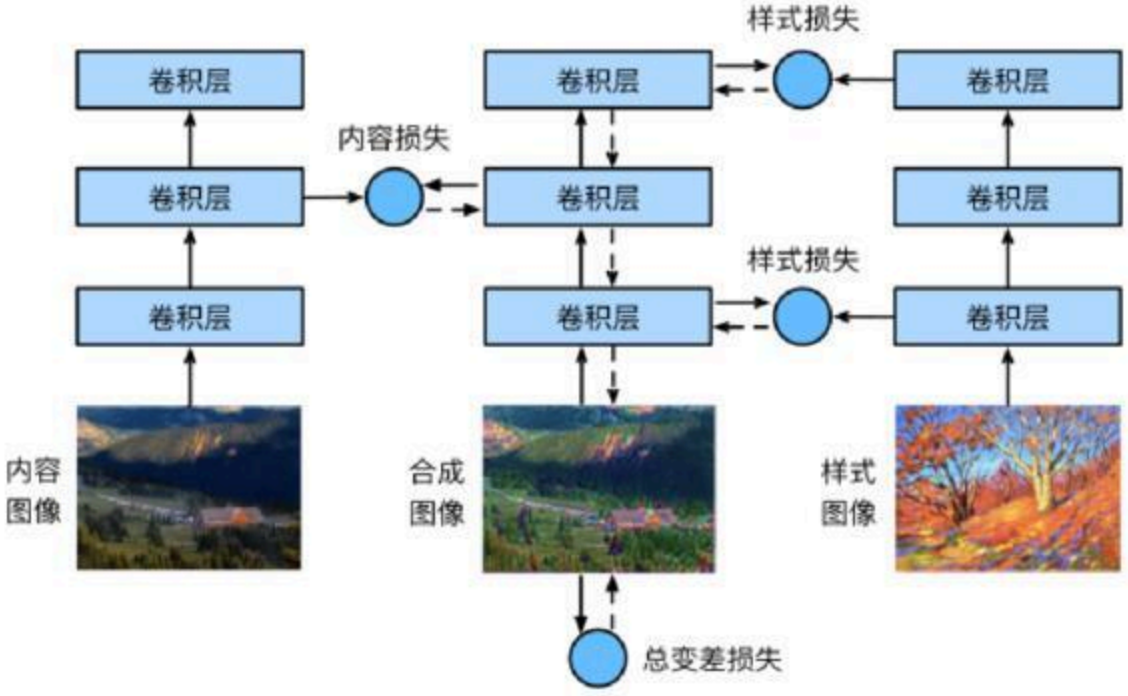
- 预训练模型只**提取图像特征**：**替换**内容或风格


样式损失
卷积层
卷积层
卷积层
内容损失
卷积层
卷积层
卷积层
样式损失
卷积层
卷积层
卷积层
内容
图像
合成
图像
样式
图像
总变差损失

风格迁移：原理

回顾：越靠近输入层，越容易抽取细节信息；反之，则越容易抽取全局概述

- 人工决定细节选用哪些**风格特征**


样式损失
卷积层
卷积层
卷积层
内容损失
卷积层
卷积层
卷积层
样式损失
卷积层
卷积层
卷积层
内容
图像
合成
图像
样式
图像
总变差损失

实验：风格迁移

小结：风格迁移

- 风格迁移：混合内容、风格图像
  - 类比微调：人工决定细节选用哪些风格特征

Review

本章内容

单发多框检测（SSD）。区域卷积神经网络（R-CNN）系列。你只看一次（YOLO）。转置卷积。全卷积网络 (FCN)。风格迁移。

重点：SSD；R-CNN、Fast R-CNN、Faster R-CNN、Mask R-CNN；YOLO；转置卷积、FCN；风格迁移。

难点：目标检测算法的实现。

学习目标

- 理解SSD的设计特点（以每个像素为锚点、多尺度检测）
- 理解R-CNN（多尺度提议区域）、Fast R-CNN（单个预训练网络、备选区域池化）、Faster R-CNN（区域提议网络）、Mask R-CNN（像素级预测）的设计特点
- 理解YOLO的设计特点（图片均分为锚框）
- 理解转置卷积的作用（上采样）、计算方法，及全卷积网络 (FCN)的架构
- 理解风格迁移的原理

问题

简述SSD的设计特点。

简述R-CNN、Fast R-CNN、Faster R-CNN、Mask R-CNN的设计特点。

简述YOLO的设计特点。

简述转置卷积的作用、计算方法，及全卷积网络 (FCN)的架构。

简述风格迁移的原理。