图像处理与分析

第十九讲：图像表示与描述

胡晰远

中国科学院自动化研究所

xiyuan.hu@ia.ac.cn

内容大纲

- 基本概念
- 表示方法
- 图像描述
  - 边界描述子
  - 区域描述子
  - 关系描述子
  - 局部特征描述子
- 图像目标检测介绍

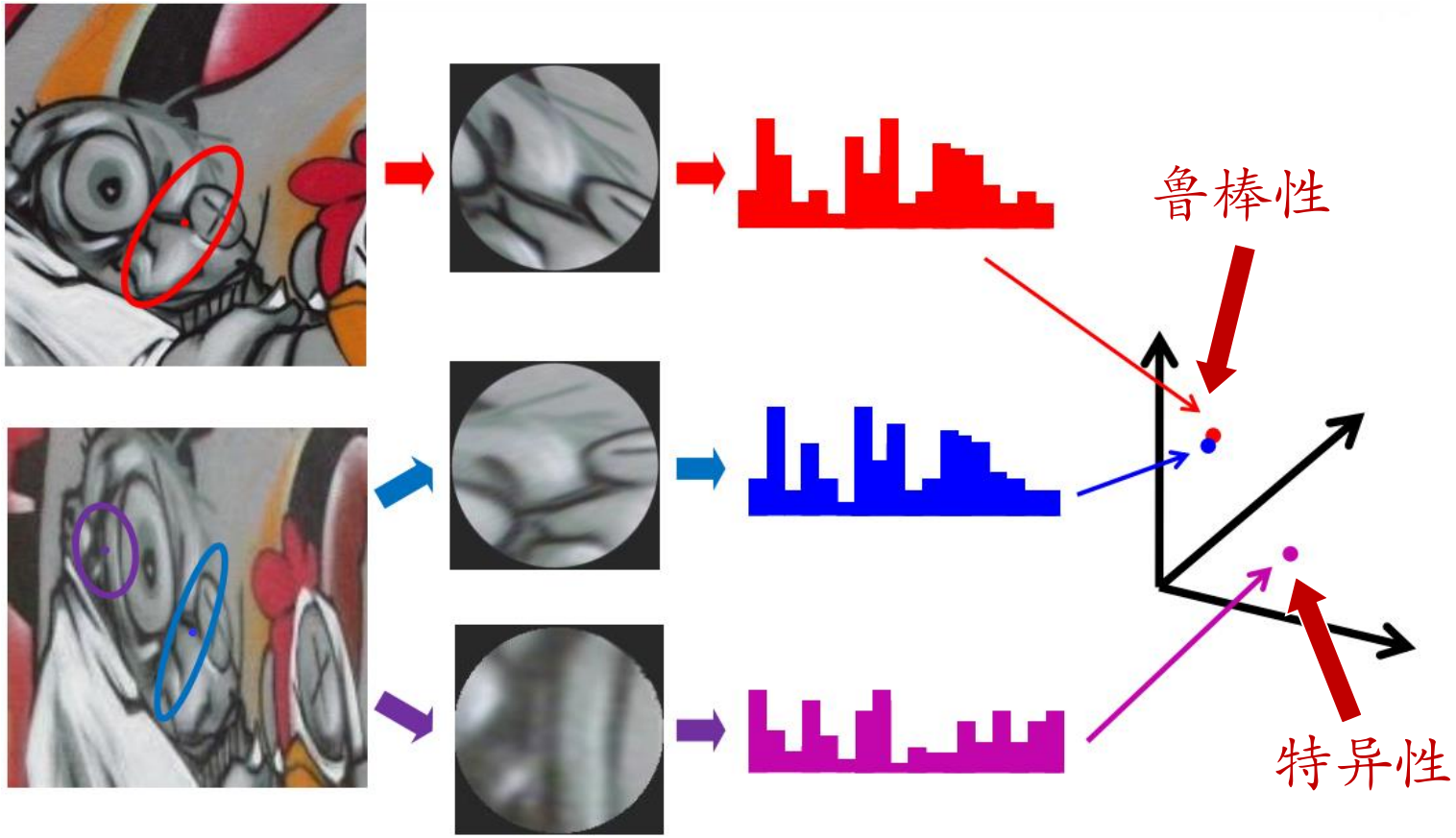
特征描述子

- 基本概念
- 应用领域
- 经典的手工特征描述子介绍
  - Gabor，HOG，SIFT，……
- 基于深度学习的特征描述

特征描述子

- 基本概念
  - 将原始的输入图像，通过某些变换（算子），映射到一个（一般维数较高的）特征空间。
  - 特征空间通常能够克服一些图像的几何形变，实现对图像相似内容或结构的有效匹配。
  - 并且对不同内容或结构能够实现有效地区分。

特征描述子

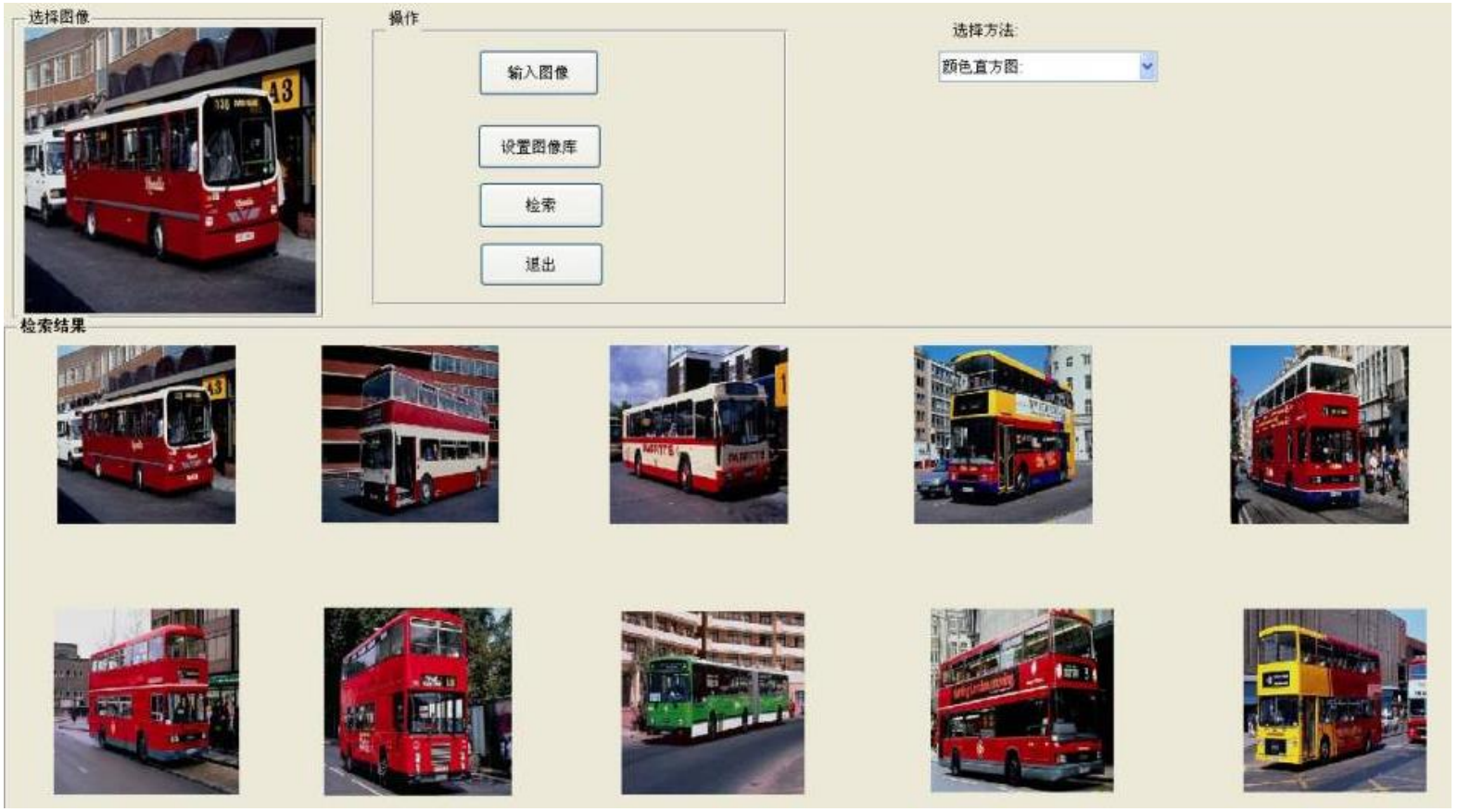


鲁棒性：相同物理点的特征描述子距离近

特异性：不同物理点的特征描述子距离远

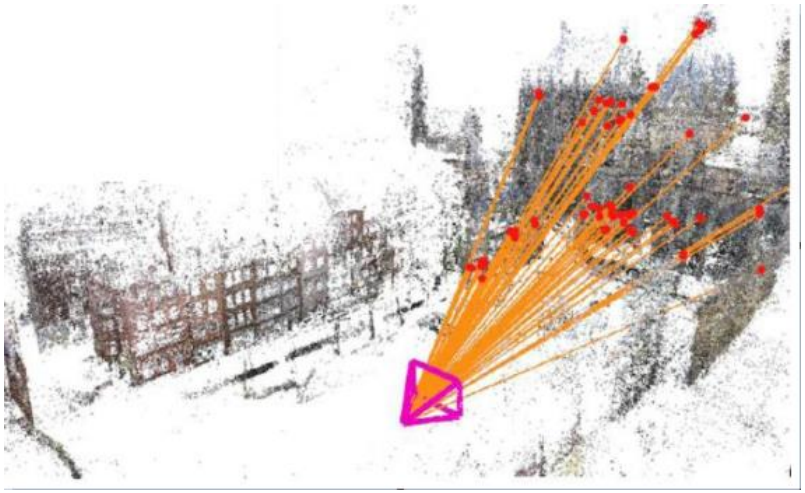
特征描述子

- 应用领域——图像内容检索



特征描述子

- 应用领域——图像拼接、三维重建



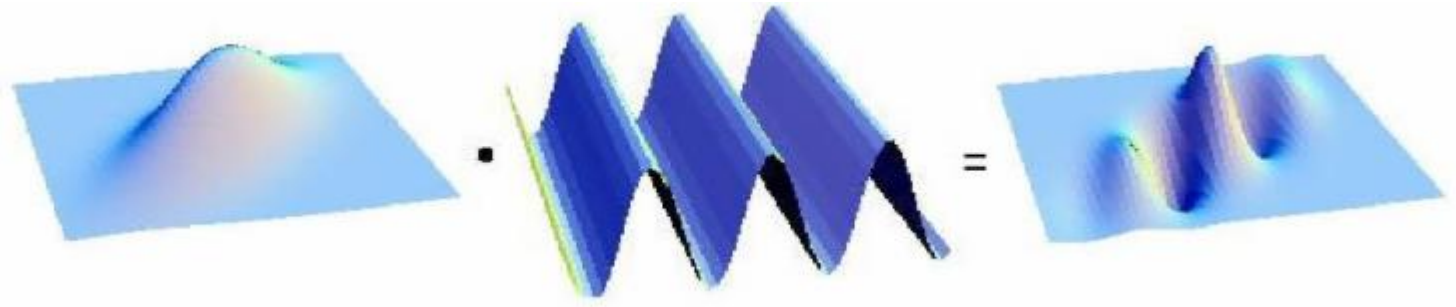
特征描述子

- 应用领域——目标跟踪、超分辨率重建

特征描述子

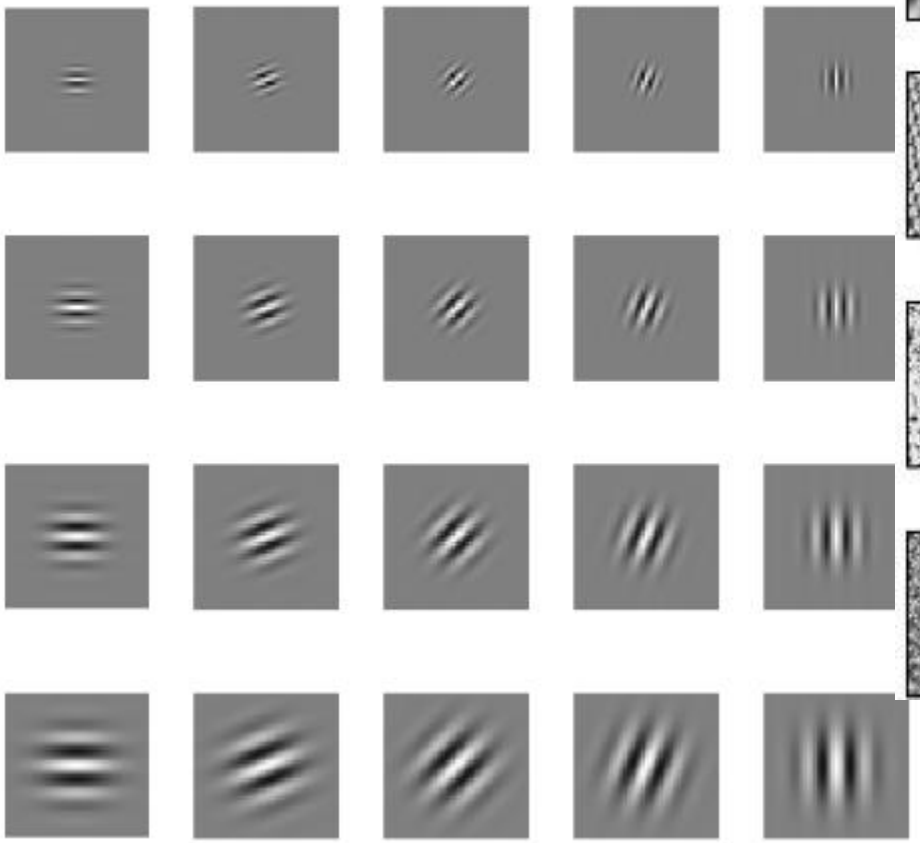
- 傅里叶描述子的改进——Gabor变换
  - 定义：$\phi(x,y)=\frac{1}{c}\exp\left\{-\left(\frac{f^2}{\gamma^2}x_r^2+\frac{f^2}{\eta^2}y_r^2\right)\right\}\exp\{j2\pi f x_r\}$

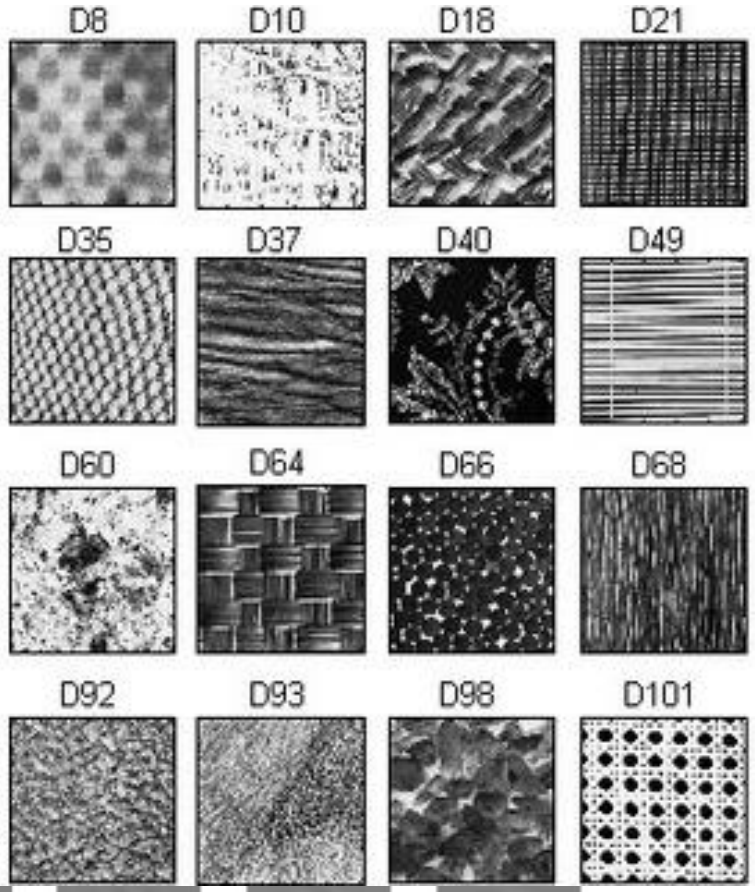
    $x_r=x\cos\theta+y\sin\theta$
  - 示例：



特征描述子——Gabor变换

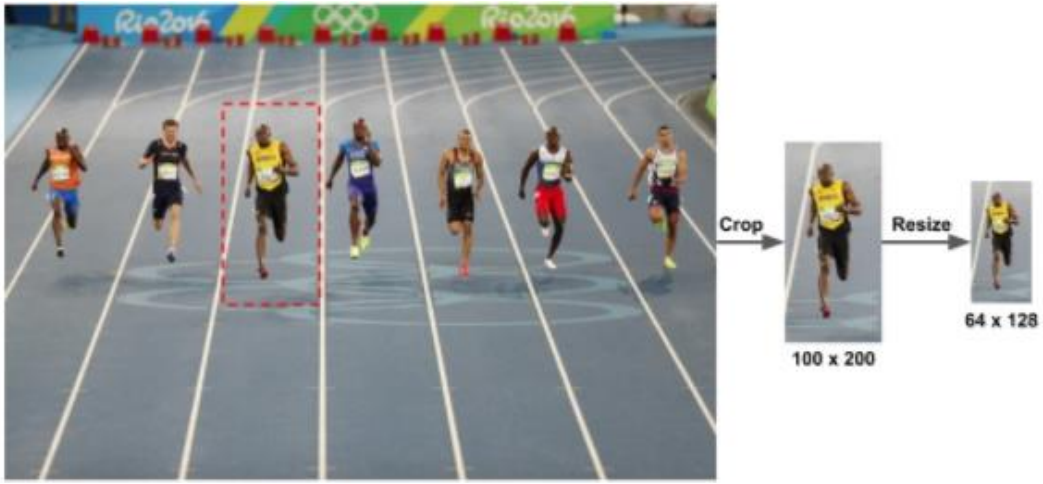
- 可以很好地描述纹理图像特





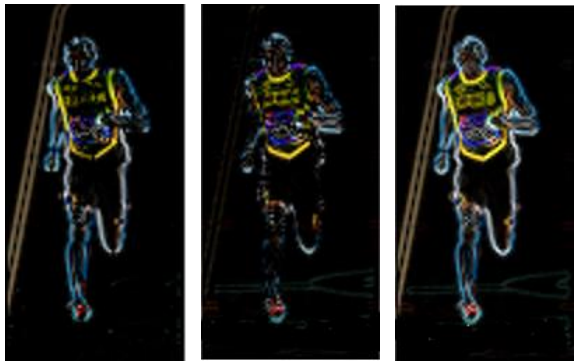
特征描述子——HOG

- 梯度方向直方图（Histogram of Oriented Gradients）
  - 利用梯度信息刻画图像中目标的边缘特征
  - 可以认为是对图像边缘描述子的扩展
  - 最初用于行人检测，但也是较为通用的特征描述子

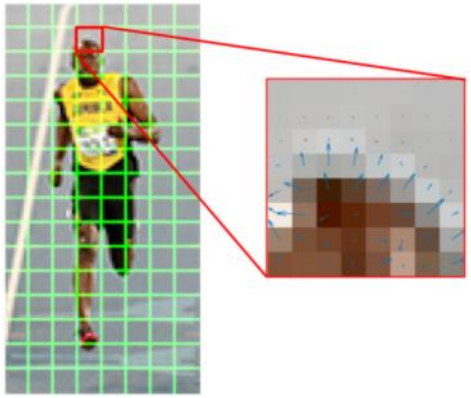


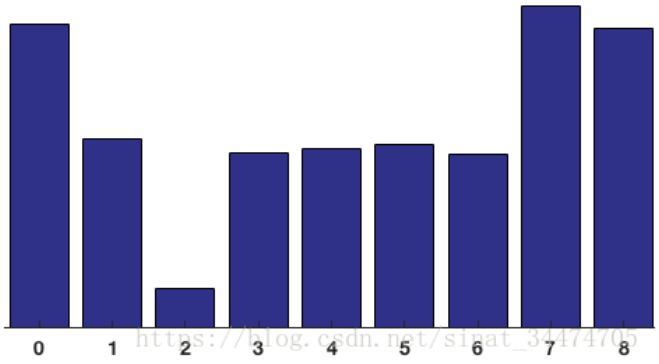
Navneet Dalal and Bill Triggs, Histogram of Oriented Gradients for Human Detection. *IEEE CVPR*, 2005.

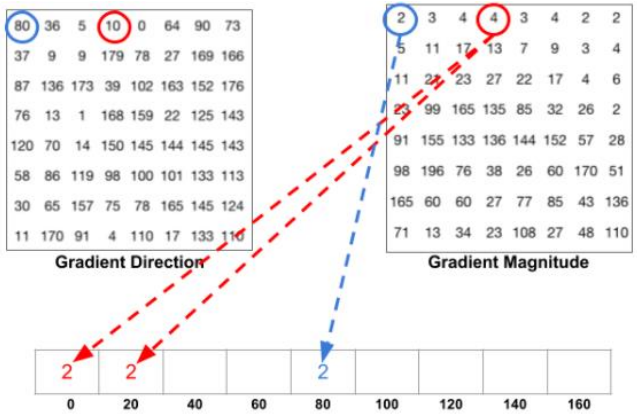
特征描述子——HOG



- 具体步骤
  - 预处理（尺寸归一化）
  - 计算梯度图（包含幅值和方向）
  - 梯度图分块（8×8）统计直方图
  - 直方图特征归一化（16×16）







特征描述子——SIFT

- 尺度不变特征（Scale Invariant Feature Transform）
  - 提取角点、边缘等特征容易受到环境（姿态，光照等）的影响
  - 适应能力差，特征不够鲁棒
  - 基于尺度空间，对缩放选择等变换具有一定的不变性
  - 一种基于区域的特征

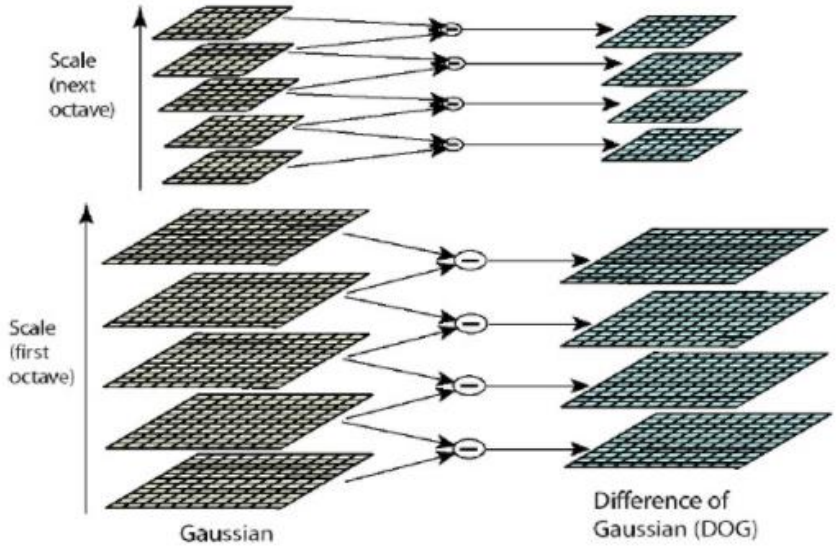
D.G. Lowe, Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004. (citations: 54000+)

特征描述子——SIFT

- SIFT特征的一些优点
  - 一定程度上保持平移、旋转和缩放不变
  - 受光照（Illumination）变化的影响较小
  - 能够解决一部分目标遮挡的情况
  - 对噪声和不同场景较为鲁棒
- 主要步骤
  - 关键点检测
  - 关键点邻域特征提取

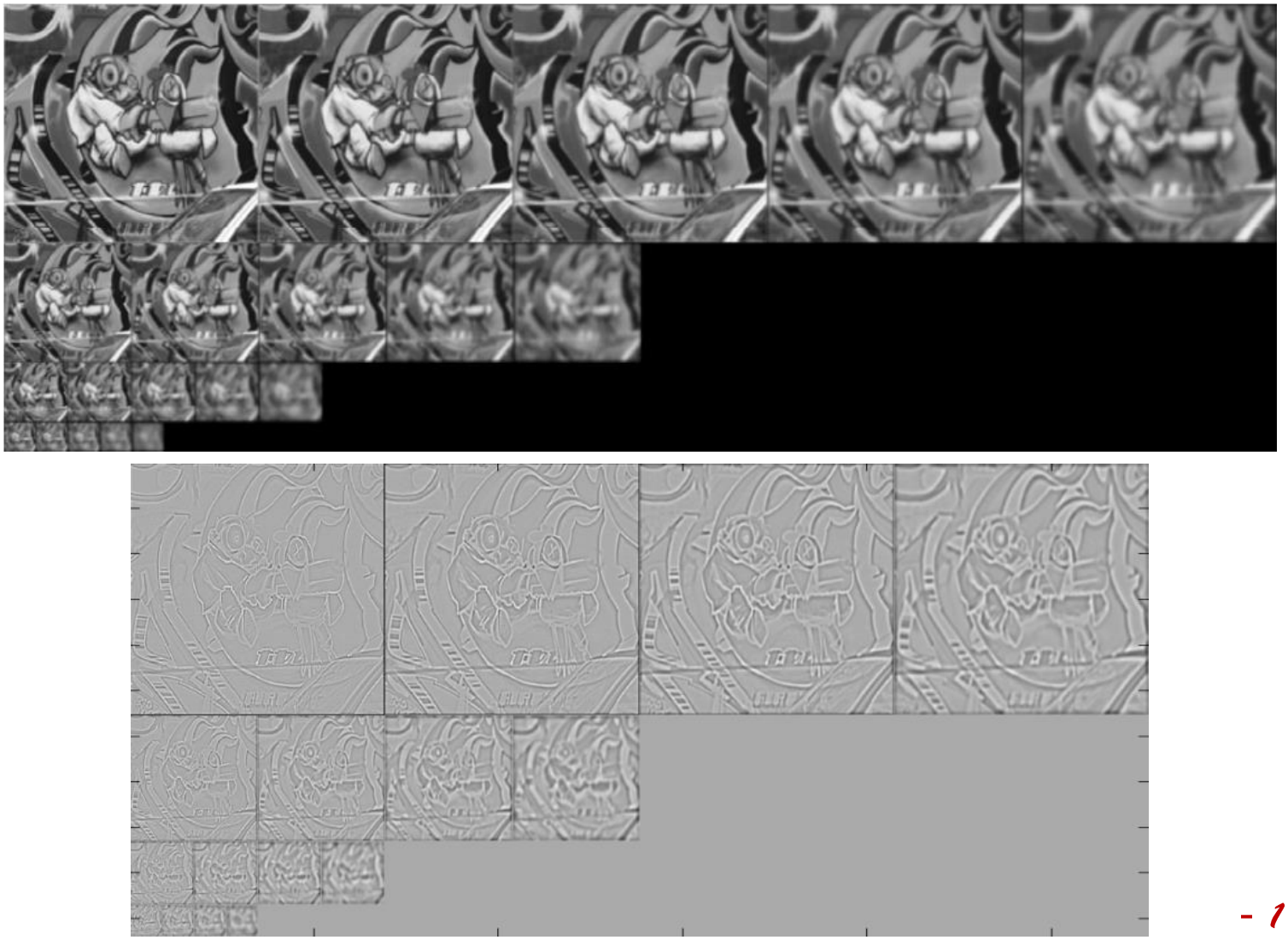
特征描述子——SIFT

- 关键点检测
  - 在不同尺度空间，一定仿射变换下，都比较稳定，并且局部特征最显著的点
  - 高斯差分金字塔（DoG）



$$DoG(x, y; \sigma) = [G_{k\sigma}(x, y) - G_{\sigma}(x, y)] * I(x, y)$$

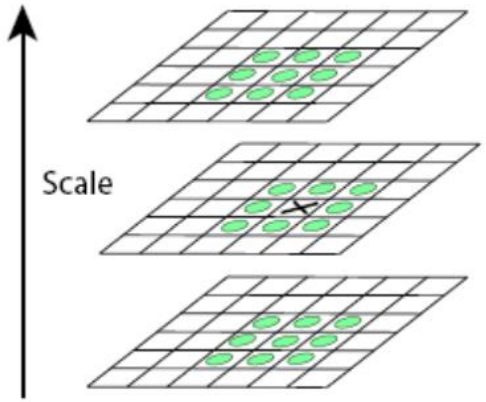
高斯差分金字塔示例



特征描述子——SIFT

- 关键点检测
  - 选取极值点，定位到亚像素精度
  - 选取稳定的极值点

$$\begin{cases} \text{对比度太低}\ (I < 0.04) \\ \text{边缘处的点}\ (r < 10) \end{cases}$$



$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

$$\mathrm{Tr}(H) = \alpha + \beta$$

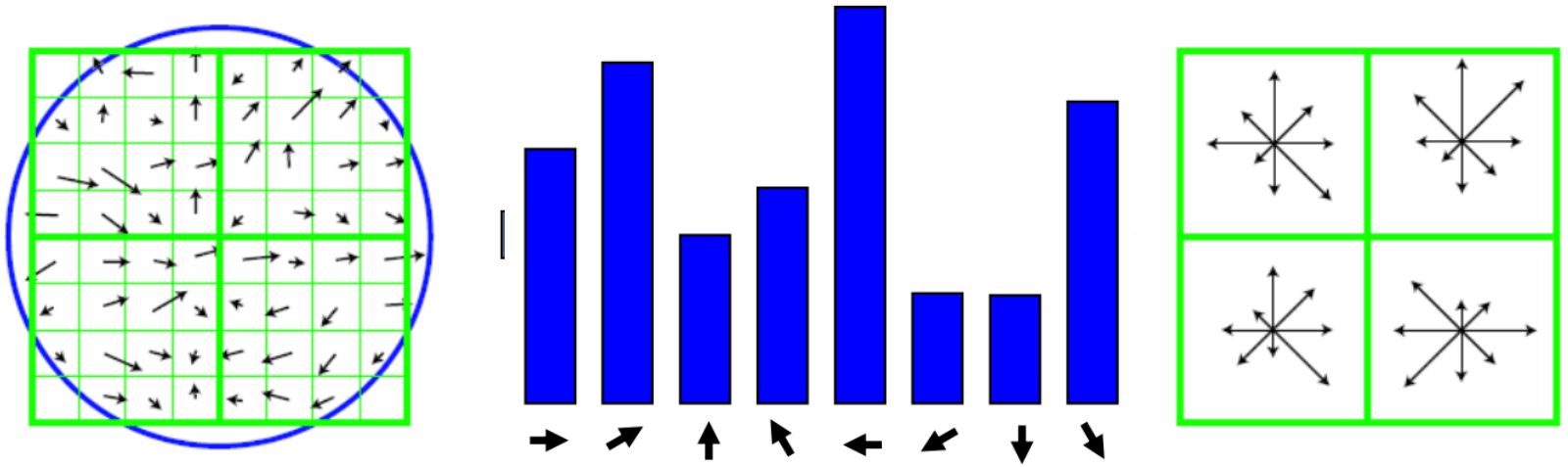
$$\mathrm{Det}(H) = \alpha\beta$$

$$\frac{\mathrm{Tr}(H)^2}{\mathrm{Det}(H)} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r+1)^2}{r}$$

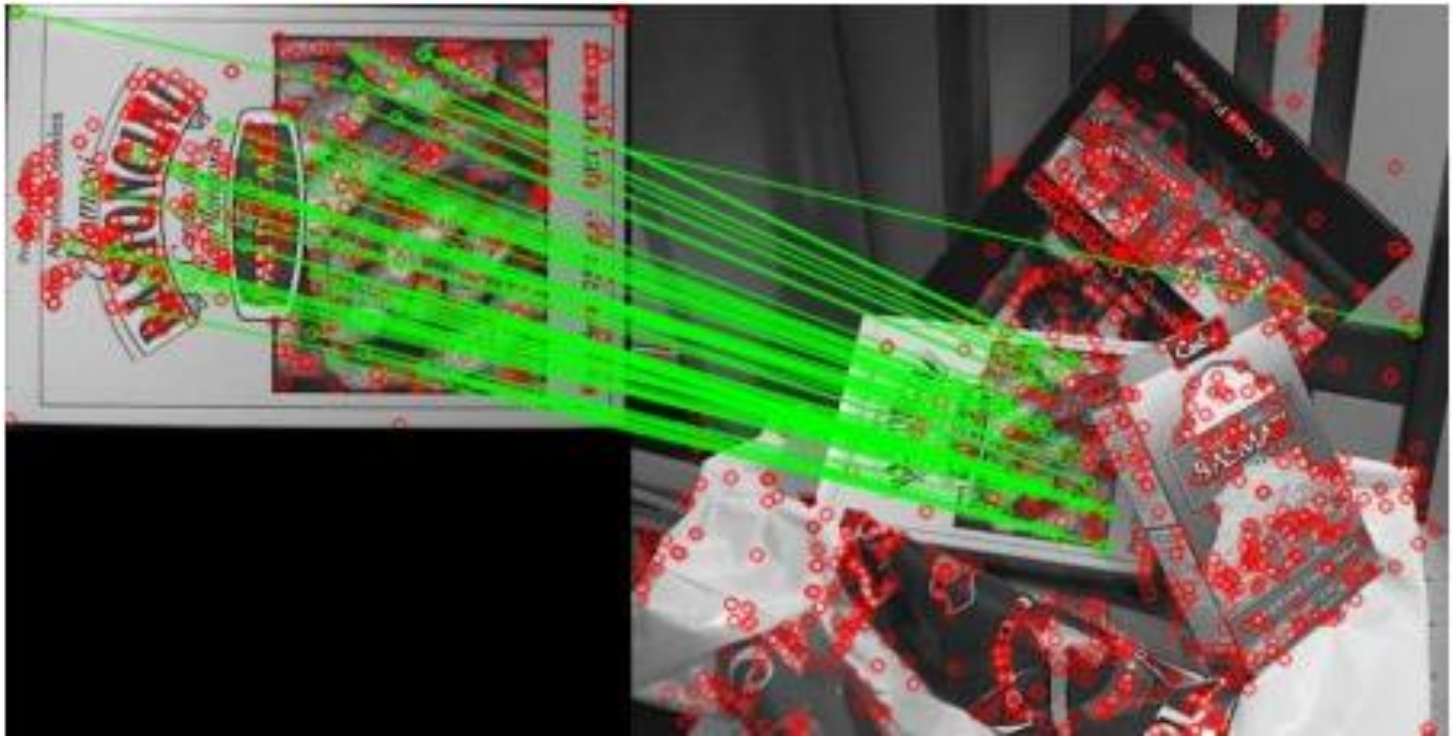
特征描述子——SIFT

- 关键点特征提取
  - 类似于HOG，在关键点邻域内计算梯度直方图
  - 采用加权方式实现：

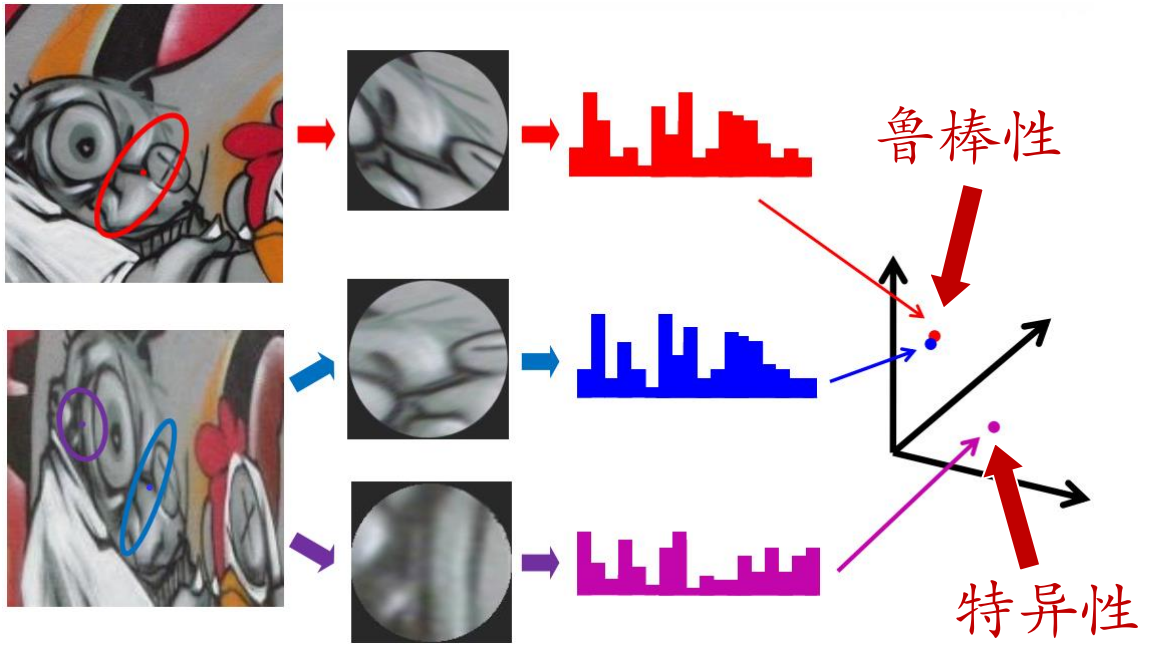
$$G_w = \sum |\nabla I_\sigma(x_i, y_i)| \exp\left\{-\frac{d_{x_i}^2 + d_{y_i}^2}{2\sigma_w^2}\right\}$$



SIFT特征点匹配示例

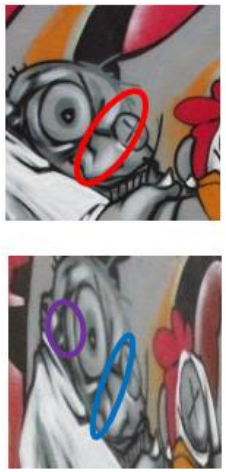


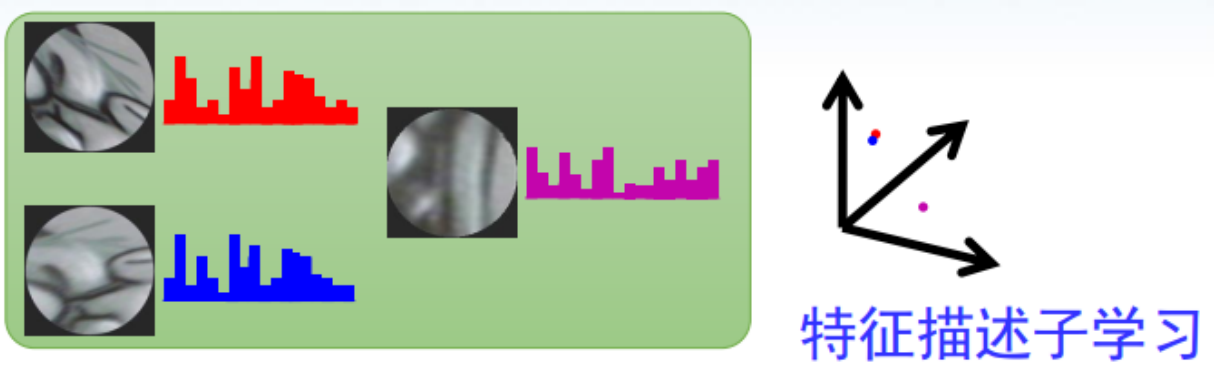
基于深度学习的特征描述子



- 传统手工特征：底层特征+特征汇聚+归一化→描述子
  - 优点：通用性强，可解释性好
  - 缺点：依赖专家知识，难度大，设计周期长

基于深度学习的特征描述子

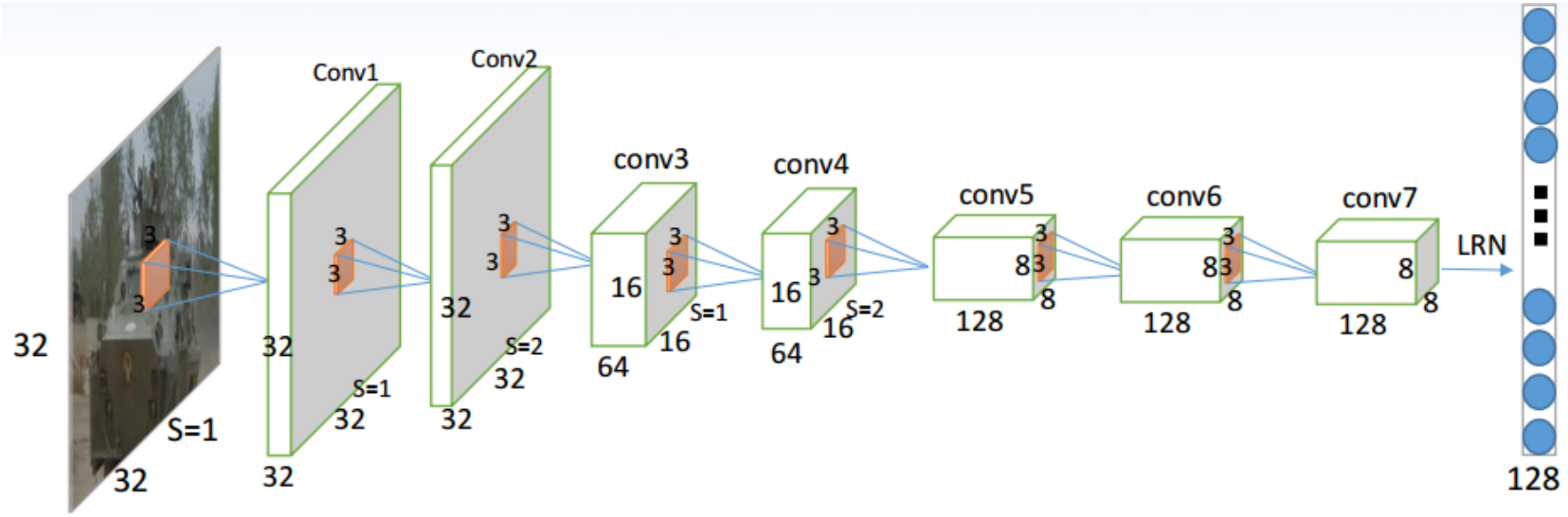




- 基于CNN的局部图像特征描述子学习：
  - 网络输出即特征描述子，在欧式空间使用，直接替代传统方法，应用广泛，适合于最近邻匹配

基于深度学习的特征描述子

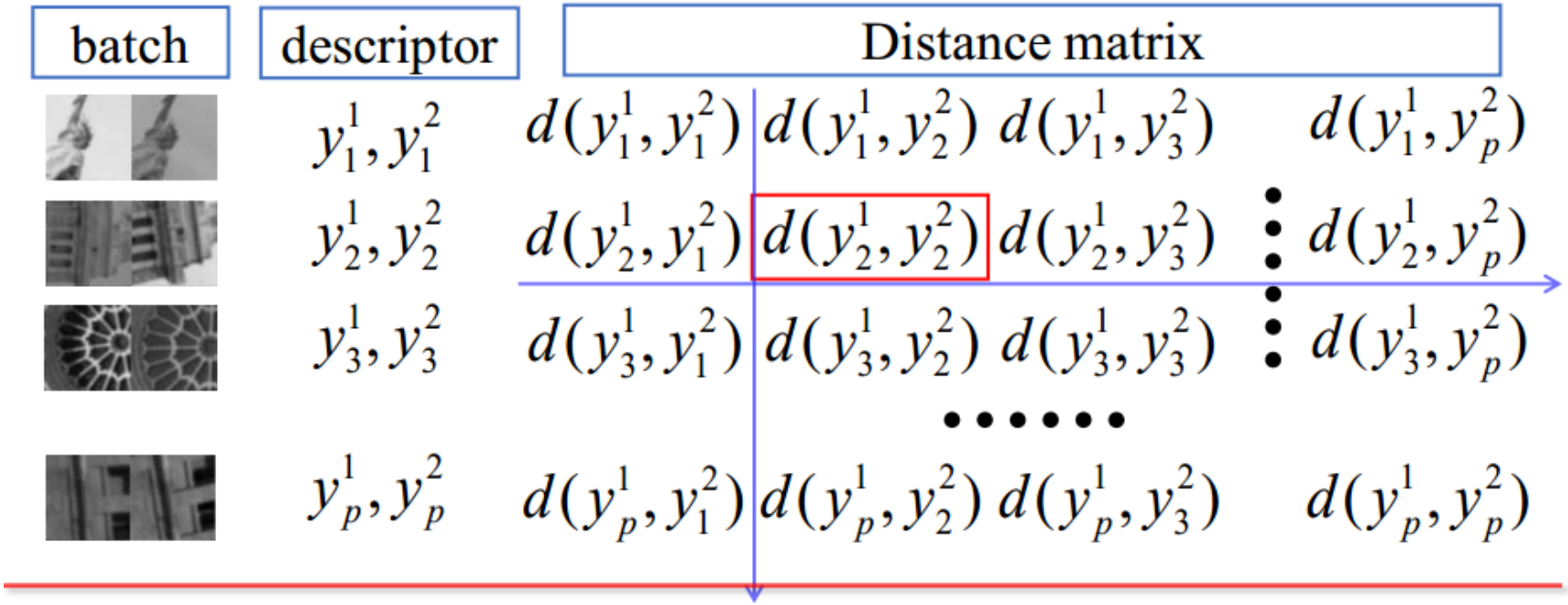
- 特征描述子提取网络结构：
  - 全卷积结构，卷积后接BN层，最后进行归一化
  - 输入32×32，输出128×1



Y. Tian, B. Fan, F. Wu, L2-Net: Deep Learning of Discriminative Patch Descriptor in Euclidean Space. *IEEE CVPR*, 2017.

基于深度学习的特征描述子

- 结构化损失：一个样本y_i^1与它匹配的样本y_i^2之间的距离小于所有与它不匹配样本之间的距离



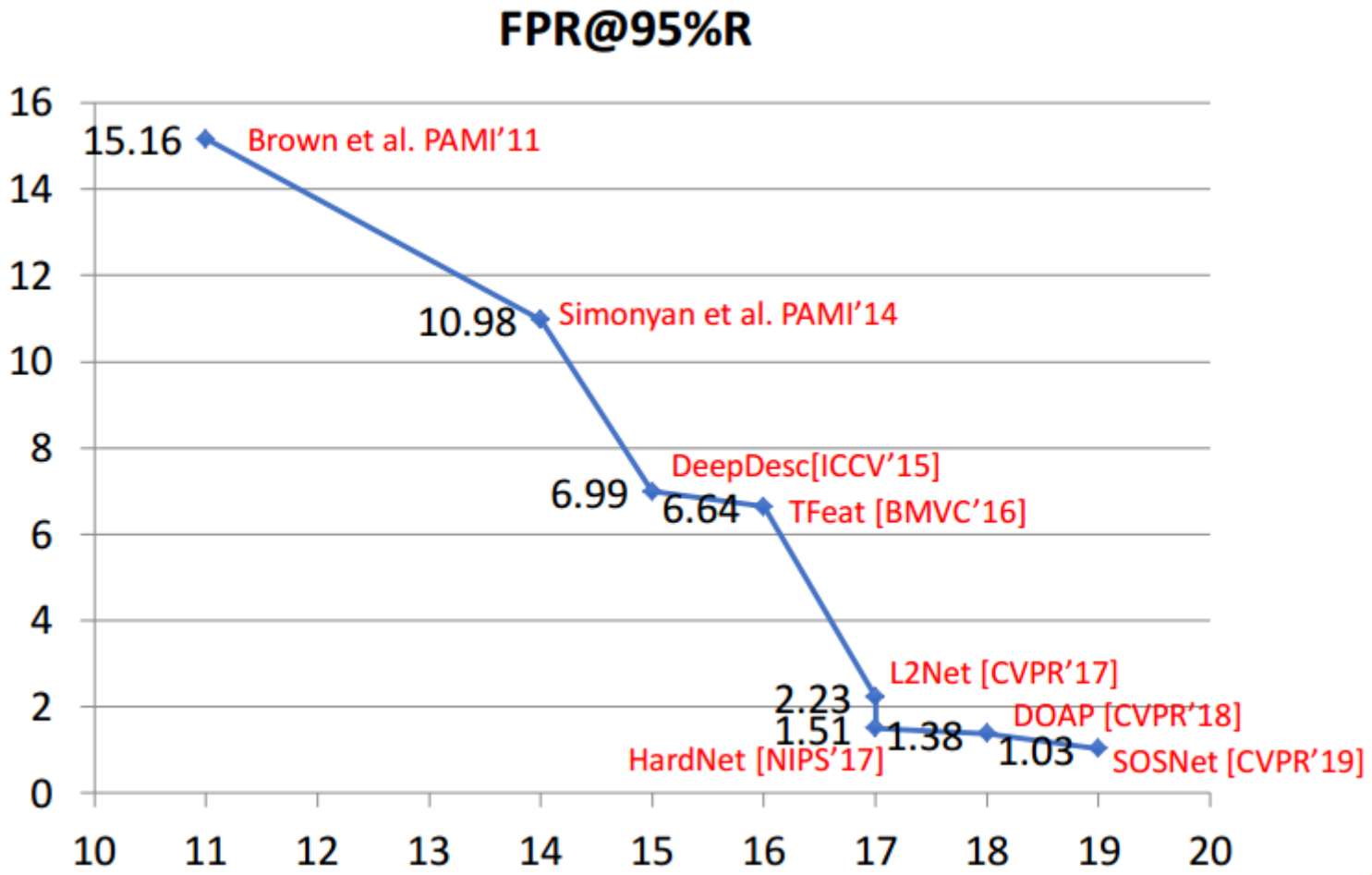
$$s_i^r = \exp(2-d_{ii}) / \sum_m \exp(2-d_{im})$$

$$s_i^c = \exp(2-d_{ii}) / \sum_m \exp(2-d_{mi})$$

$$E_1 = -\frac{1}{2}\left(\sum_i \log s_i^r + \sum_i \log s_i^c\right)$$

基于深度学习的特征描述子

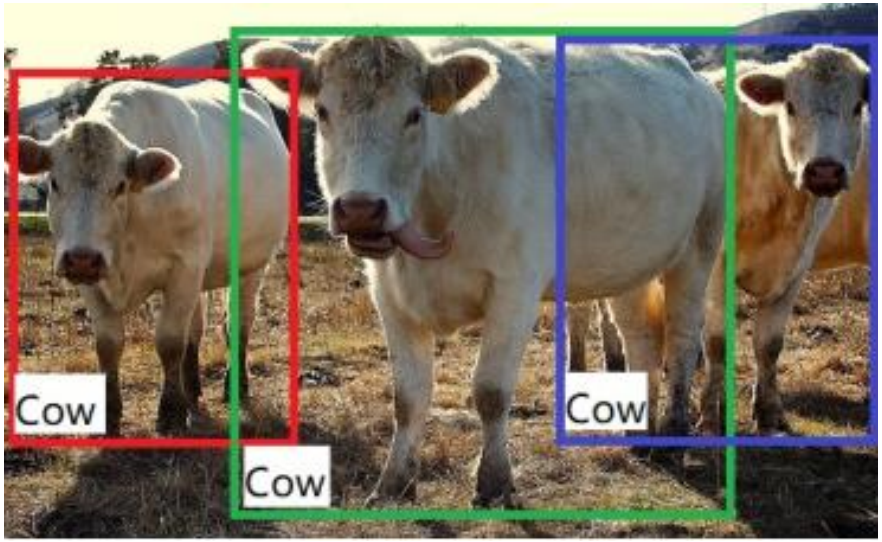
- 基于深度学习的特征描述子研究进展



内容大纲

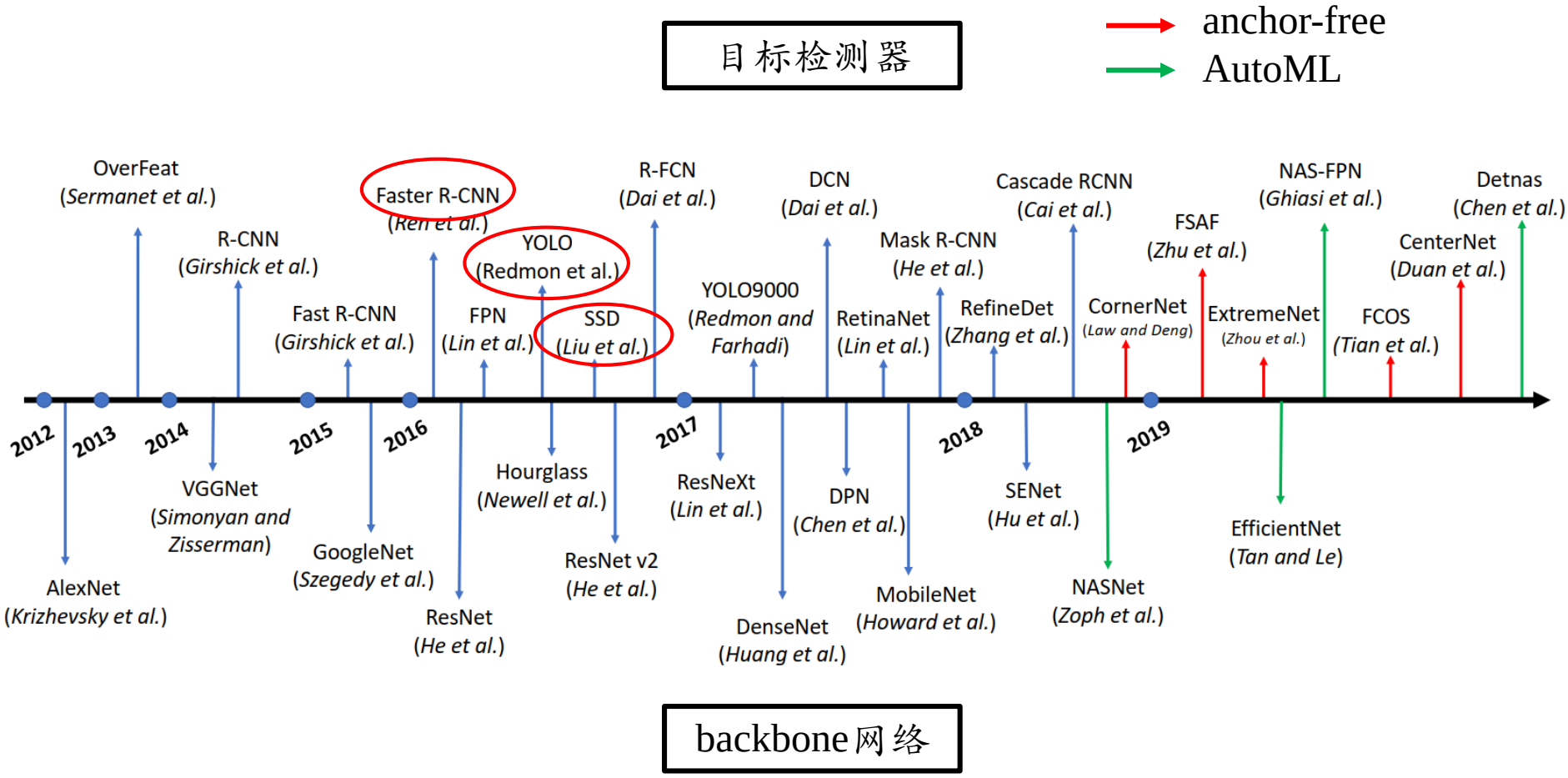
- 基本概念
- 表示方法
- 图像描述
  - 边界描述子
  - 区域描述子
  - 关系描述子
  - 局部特征描述子
- 图像目标检测介绍

Object Detection in Deep Learning

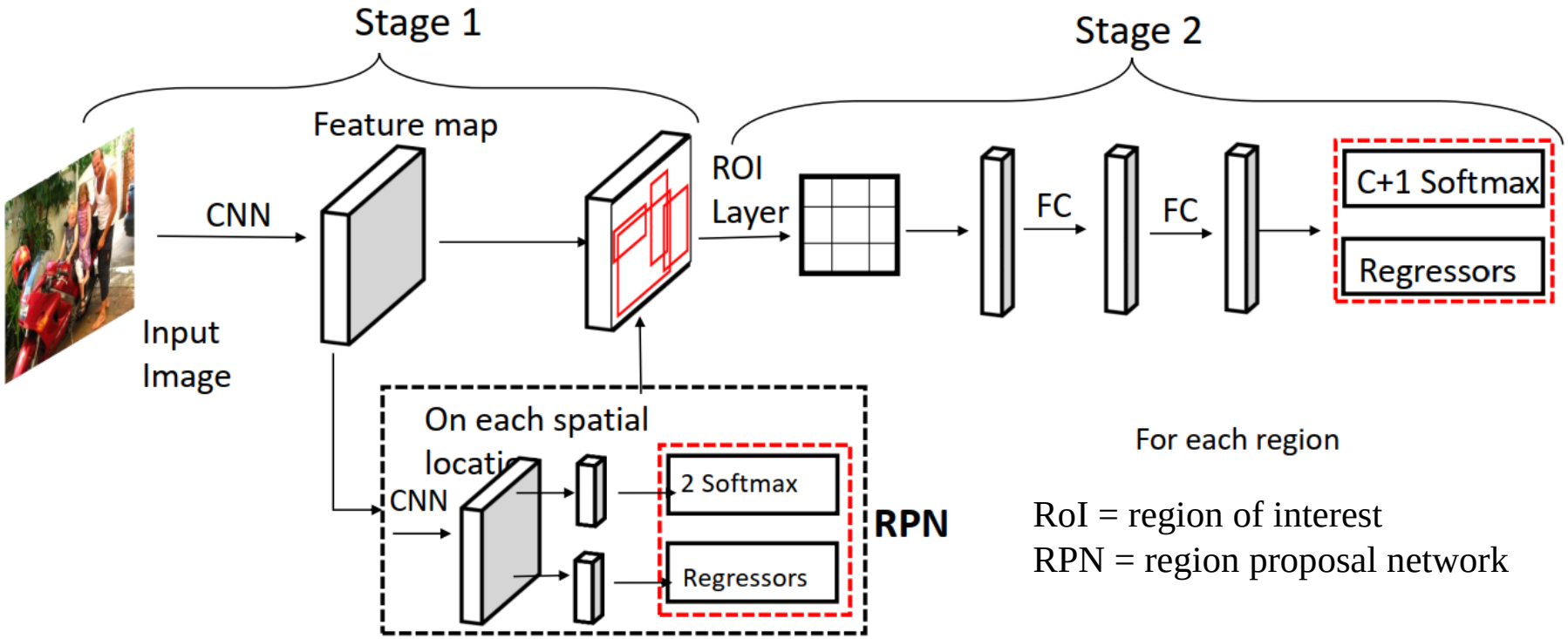


object detection =
recognition + localization

发展历程


目标检测器
anchor-free
AutoML
OverFeat (Sermanet et al.)
R-CNN (Girshick et al.)
Faster R-CNN (Ren et al.)
YOLO (Redmon et al.)
Fast R-CNN (Girshick et al.)
FPN (Lin et al.)
SSD (Liu et al.)
R-FCN (Dai et al.)
YOLO9000 (Redmon and Farhadi)
DCN (Dai et al.)
RetinaNet (Lin et al.)
Mask R-CNN (He et al.)
RefineDet (Zhang et al.)
Cascade RCNN (Cai et al.)
CornerNet (Law and Deng)
FSAF (Zhu et al.)
ExtremeNet (Zhou et al.)
NAS-FPN (Ghiasi et al.)
FCOS (Tian et al.)
CenterNet (Duan et al.)
Detnas (Chen et al.)
2012
2013
2014
2015
2016
2017
2018
2019
AlexNet (Krizhevsky et al.)
VGGNet (Simonyan and Zisserman)
GoogleNet (Szegedy et al.)
ResNet (He et al.)
Hourglass (Newell et al.)
ResNet v2 (He et al.)
ResNeXt (Lin et al.)
DenseNet (Huang et al.)
DPN (Chen et al.)
MobileNet (Howard et al.)
SENet (Hu et al.)
NASNet (Zoph et al.)
EfficientNet (Tan and Le)
backbone网络

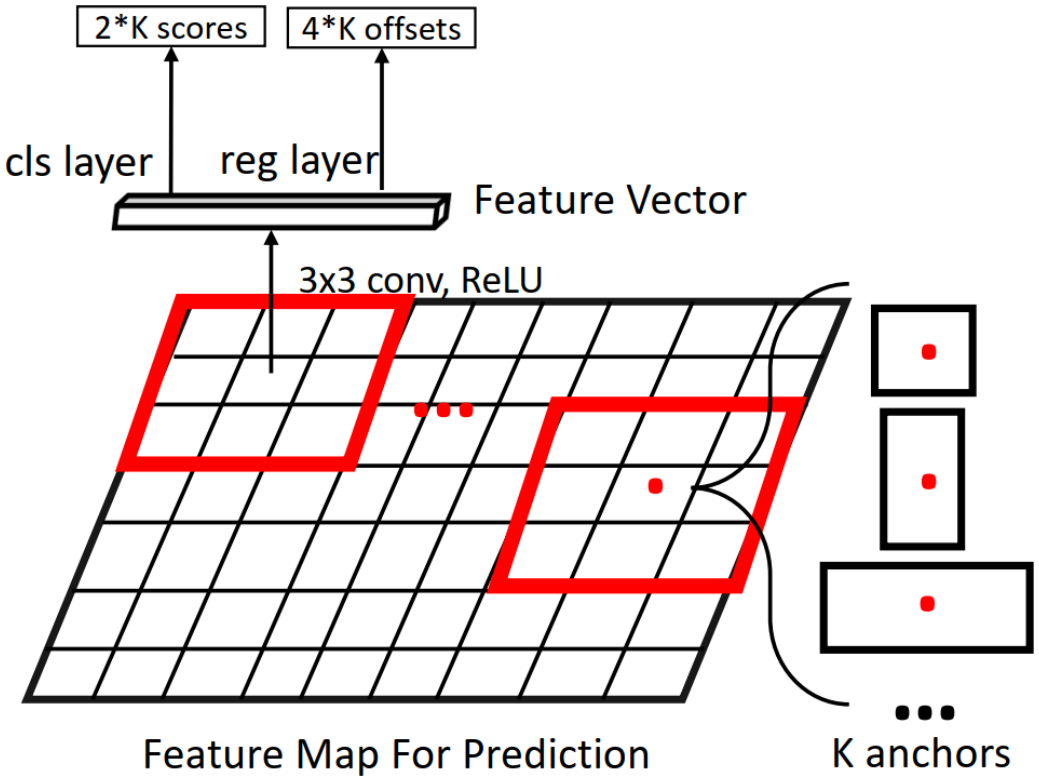
Faster R-CNN（2-stage，anchor-based）



S. Ren, K. He, R. Girshick, J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, in: *NeurIPS*, 2015

Faster R-CNN（2-stage，anchor-based）

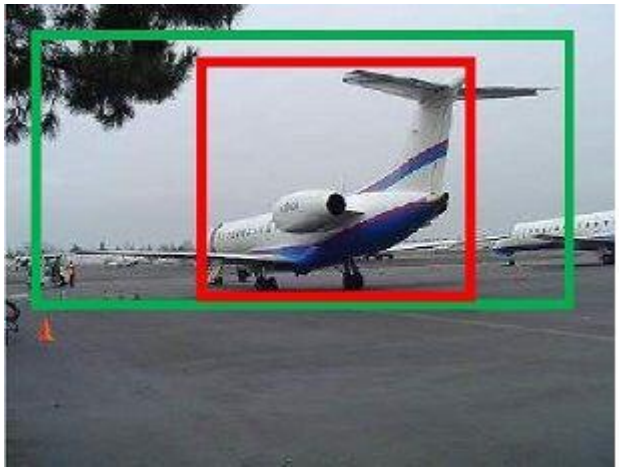
- RPN: Region Proposal Network

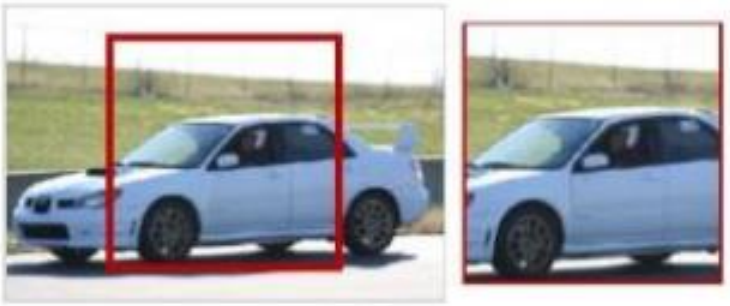


- anchor起参照物的作用
- ground-truth box和predicted box在这个参照系下算loss
- anchor也可看做是predicted box回归的初值

Faster R-CNN（2-stage，anchor-based）

- RPN and ROI Align（ROI Pooling）





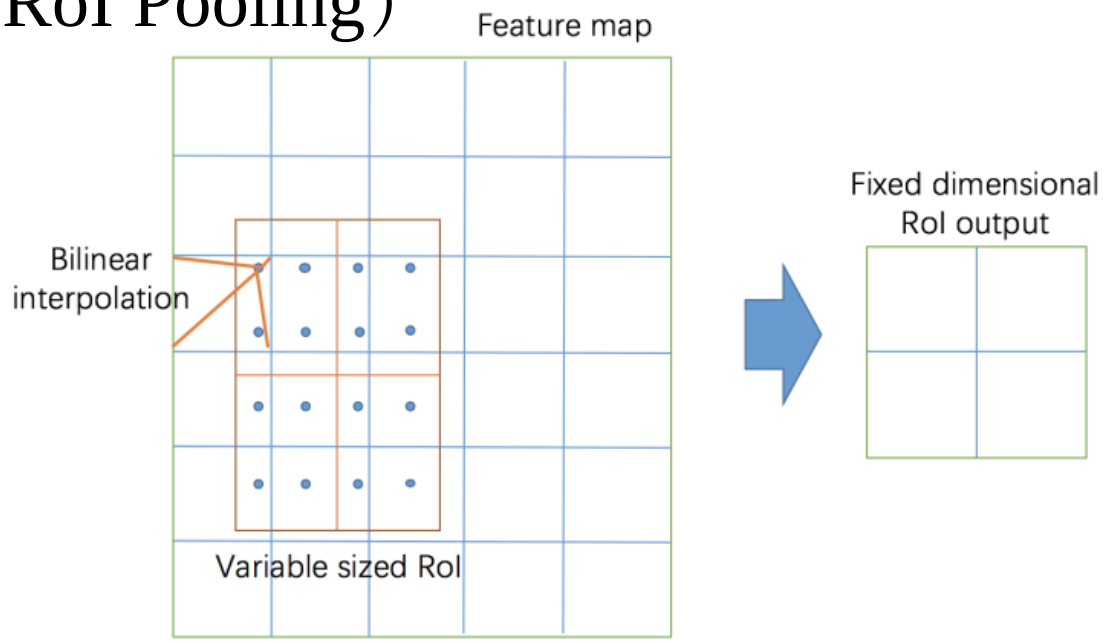
crop

warp

Faster R-CNN（2-stage，anchor-based）

- RoIAlign（RoI Pooling）

图示中
n=2, m=4



- 提取任意大小RoI内的特征，将其变换成固定大小的输出，需使用RoIAlign操作。
- 将任意大小的RoI划分为n*n网格，每个网格内有m个采样点，每个采样点上的特征值由双线性插值得到，取每个网格内所有采样点中最大或平均作为该网格的输出

Faster R-CNN（2-stage，anchor-based）

- Loss $L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i, t_i^*)$

p^*: 类别ground-truth　　t^*: 位置ground-truth

$$t_x = (x - x_a)/w_a, \quad t_y = (y - y_a)/h_a, \quad t_w = \log(w/w_a), \quad t_h = \log(h/h_a),$$
$$t_x^* = (x^* - x_a)/w_a, \quad t_y^* = (y^* - y_a)/h_a, \quad t_w^* = \log(w^*/w_a), \quad t_h^* = \log(h^*/h_a)$$

下标a代表anchor，t和t^*都是相对anchor的offset

定位使用smoothL1损失

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \qquad L_{reg}(t, t^*) = \sum_{i\in\{x,y,w,h\}} \text{smooth}_{L_1}(t_i - t_i^*)$$

分类使用交叉熵损失

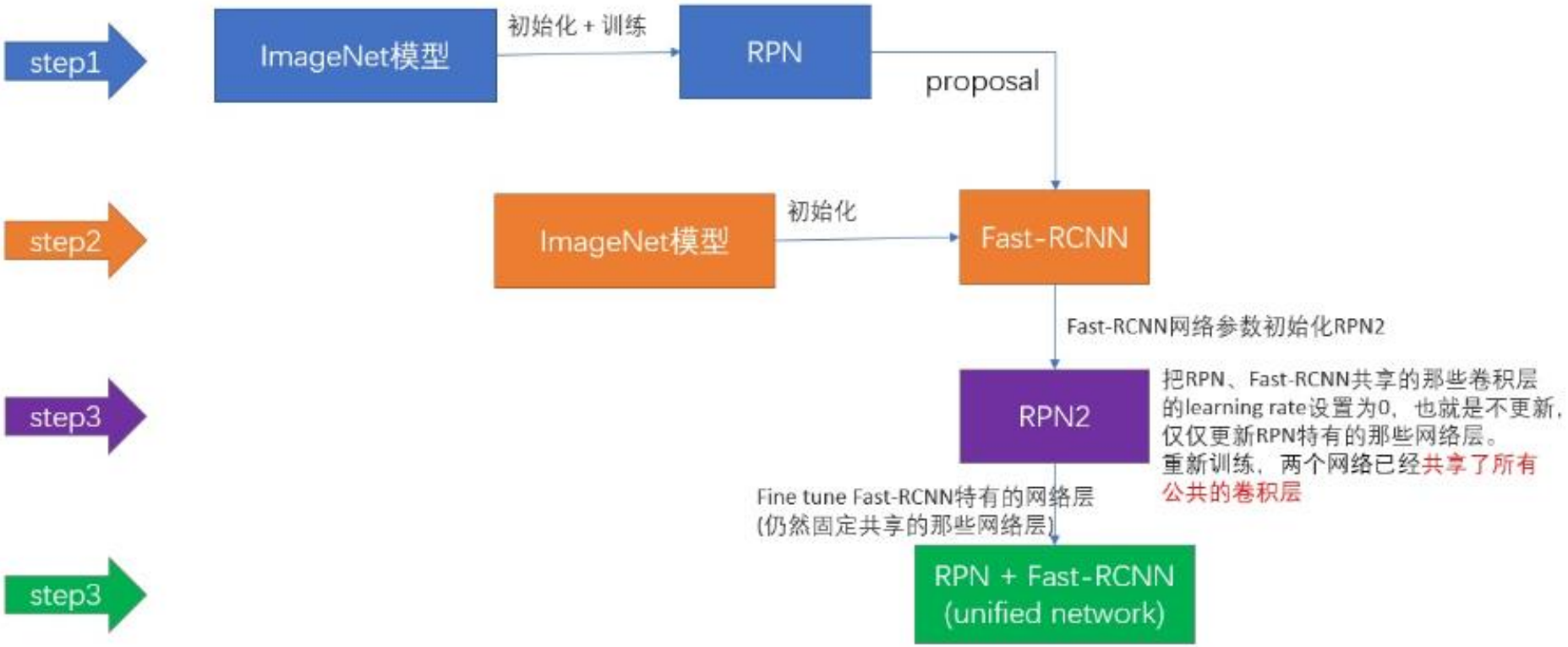
$\text{CE}(p, p^*) = -p^*\log(p) - (1 - p^*)\log(1 - p)$　　sigmoid激活

或　$\text{CE}(p, p^*) = -\log(p_j)$　　softmax激活

j为正确类别的下标

Faster R-CNN（2-stage，anchor-based）

- Training Process

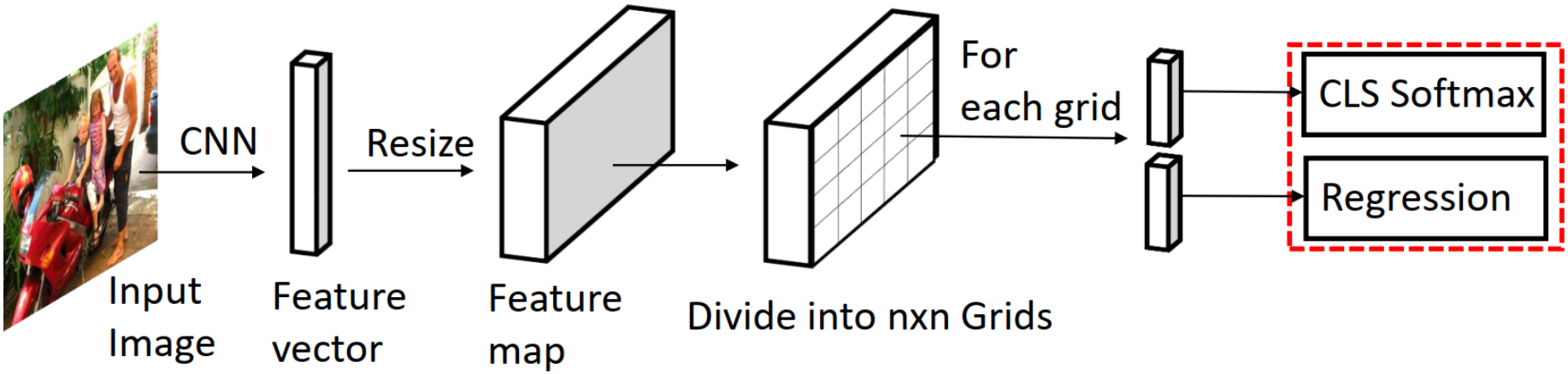


Faster R-CNN（2-stage，anchor-based）

- Faster R-CNN的缺陷及改进
  - 需要在特征图上取完ROI后，对每个Region卷积，进行分类和回归，耗时较长；
  - 共享ROI后的分类和回归计算，只对不同位置输出进行投票（R-FCN，NIPS 2016）
  - 正负样本的分类仅仅由IOU的阈值进行划分，风险较大（低阈值误检，高阈值漏检）
  - 采用级联式结构，增加高质量（高IOU）Proposal的数量，提高IOU阈值（Cascade R-CNN，CVPR 2018）

YOLO（1-stage，anchor-free）

- YOLO： You Only Look Once



在backbone网络里卷积层后接了全连接，变成了特征向量

J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: *CVPR*, 2016.

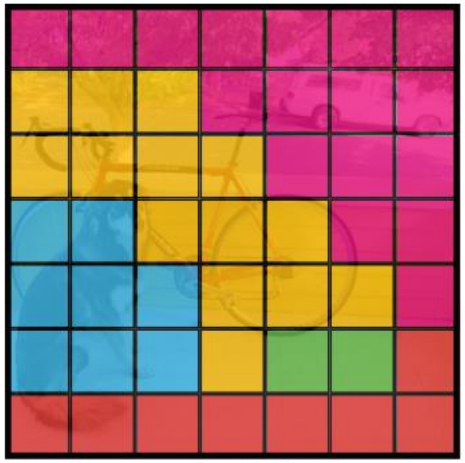
YOLO（1-stage，anchor-free）

每个网格负责预测B个box

S × S grid on input

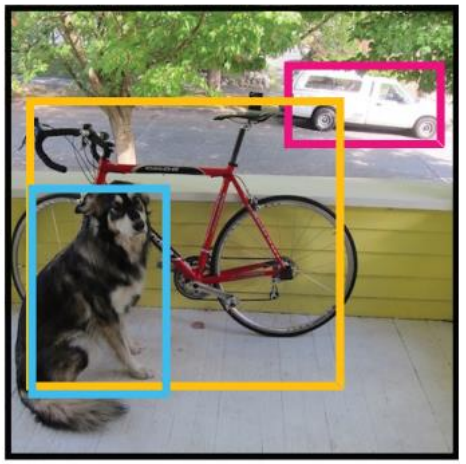
若物体中心落在网格内，则该网格负责预测该物体

Bounding boxes + confidence



Class probability map

Confidence表示框内包含物体的置信度定义为 $P(Object) \times IoU_{truth\_pred}$



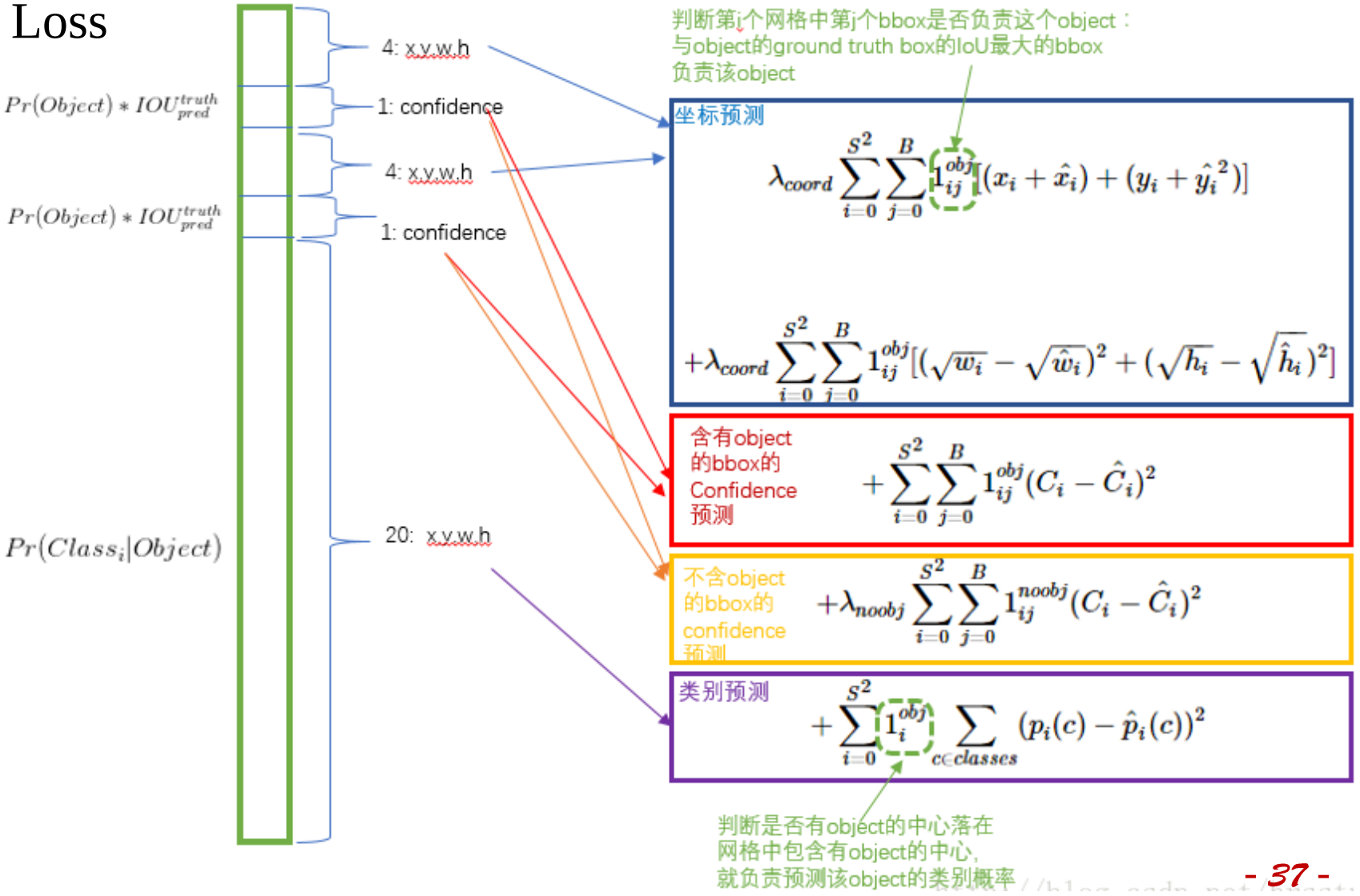
Final detections

预测时将每个网格的条件类概率和单个box的置信度相乘，得到每个box的类别置信度

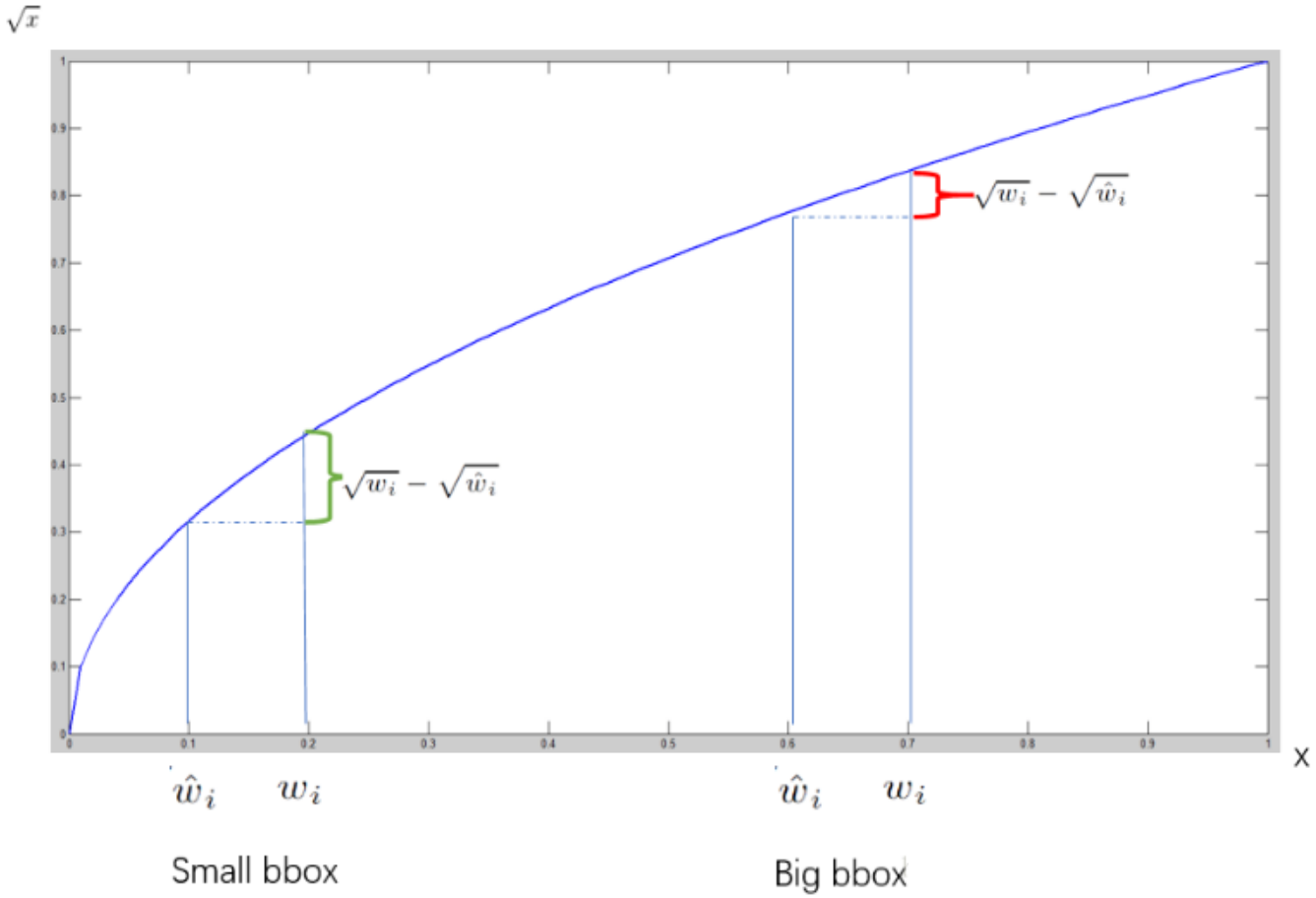
无论B为多少，每个网格只预测一组类别概率（C类）

YOLO（1-stage，anchor-free）

Loss

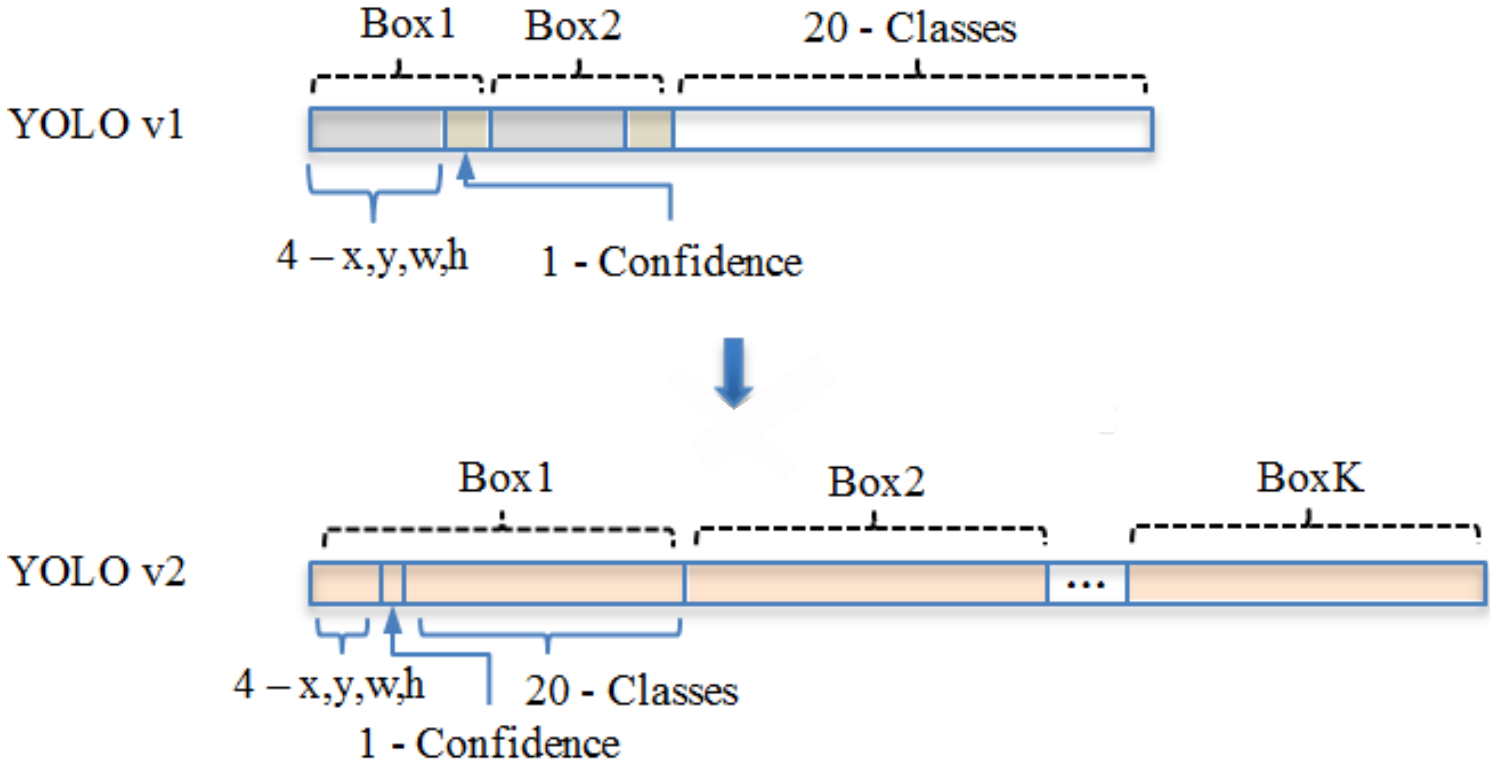


YOLO（1-stage，anchor-free）



相比于大bbox，相同偏差下小bbox的IoU变化更大，而相同偏差下的sum-square error是一样的，因此将bbox的宽和高取平方根

YOLO（1-stage，anchor-based）

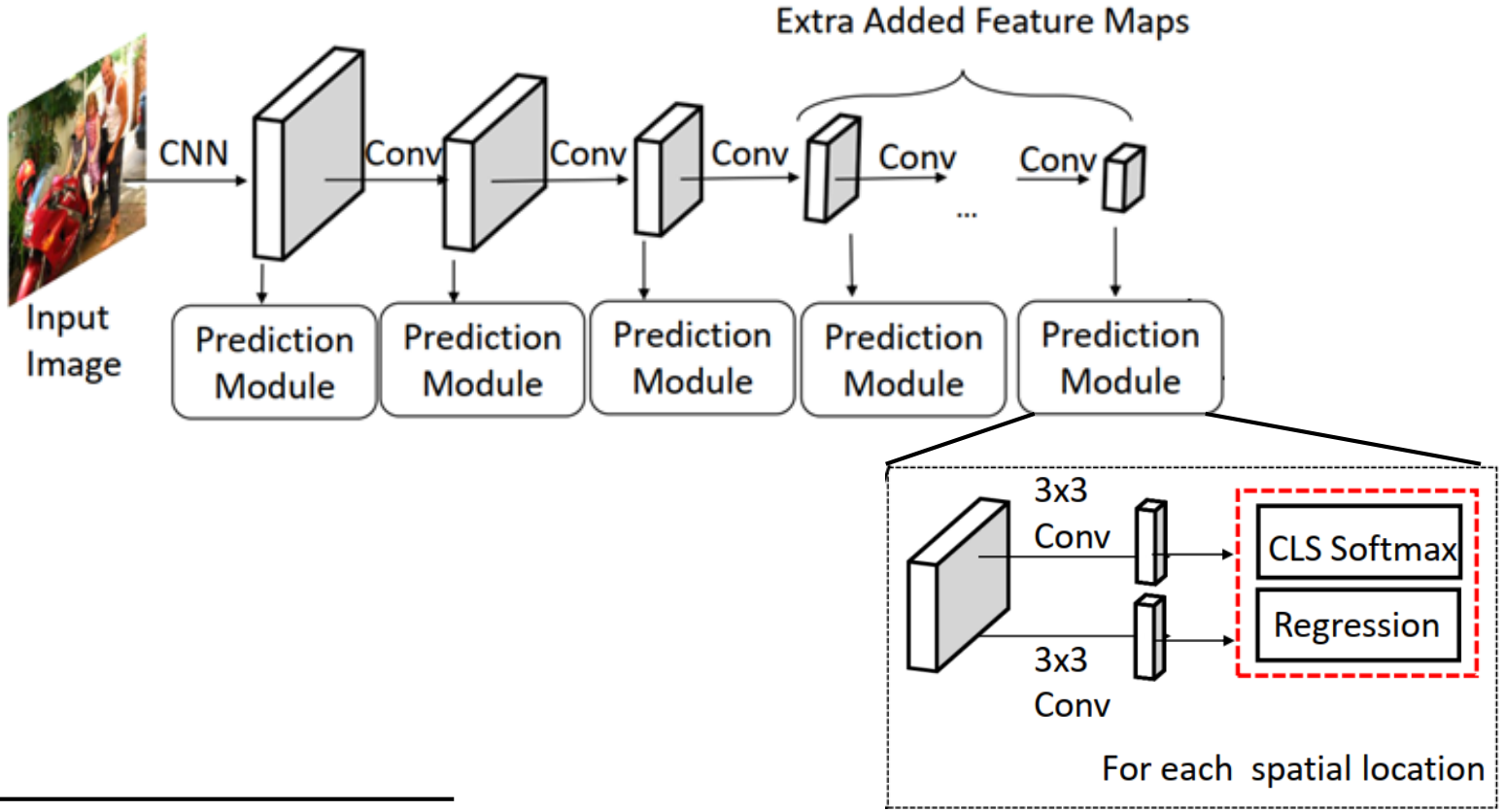


- YOLOv1中每个网格预测B个box，这B个box共用一个分类结果（20类），导致一个网格只能预测一类，易漏检
- YOLOv2中每个网格预测K个box（类似Faster R-CNN），每个box都预测一组类别

J. Redmon, A. Farhadi, Yolo9000: better, faster, stronger, in: *CVPR*, 2017.

SSD（1-stage，anchor-based）

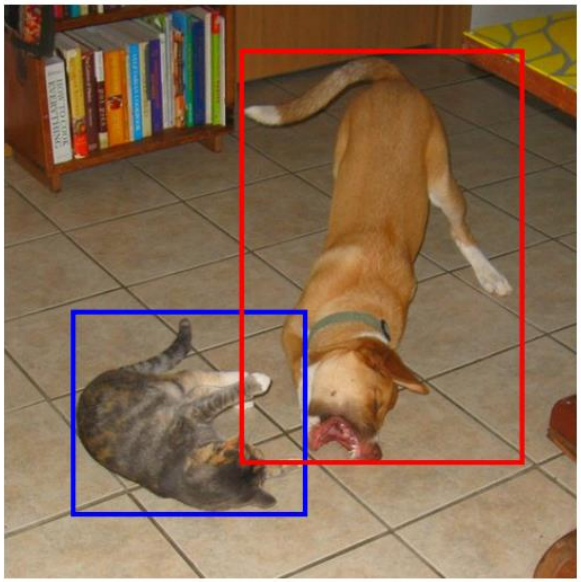
- SSD： Single Shot Multibox Detector



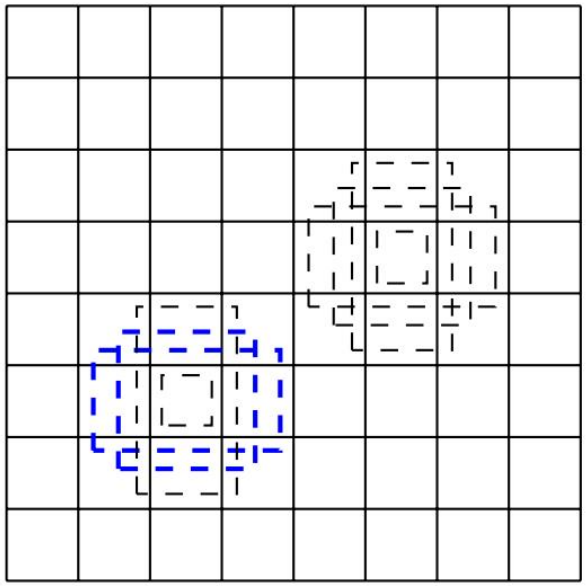
W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, SSD: Single shot multibox detector, in: *ECCV*, 2016.

SSD（1-stage，anchor-based）

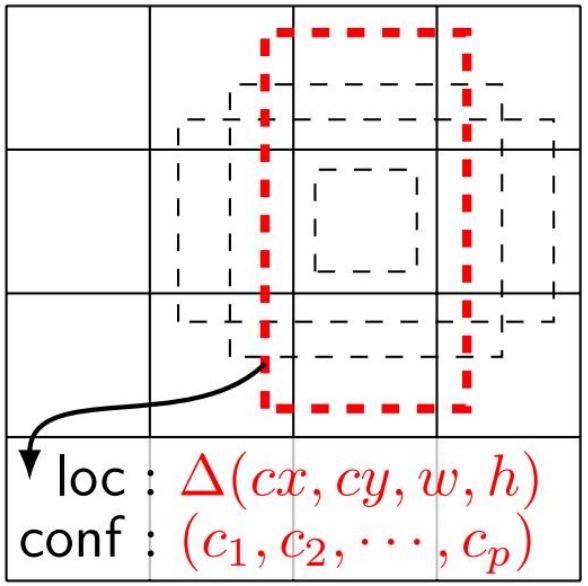
- Anchor



(a) Image with GT boxes



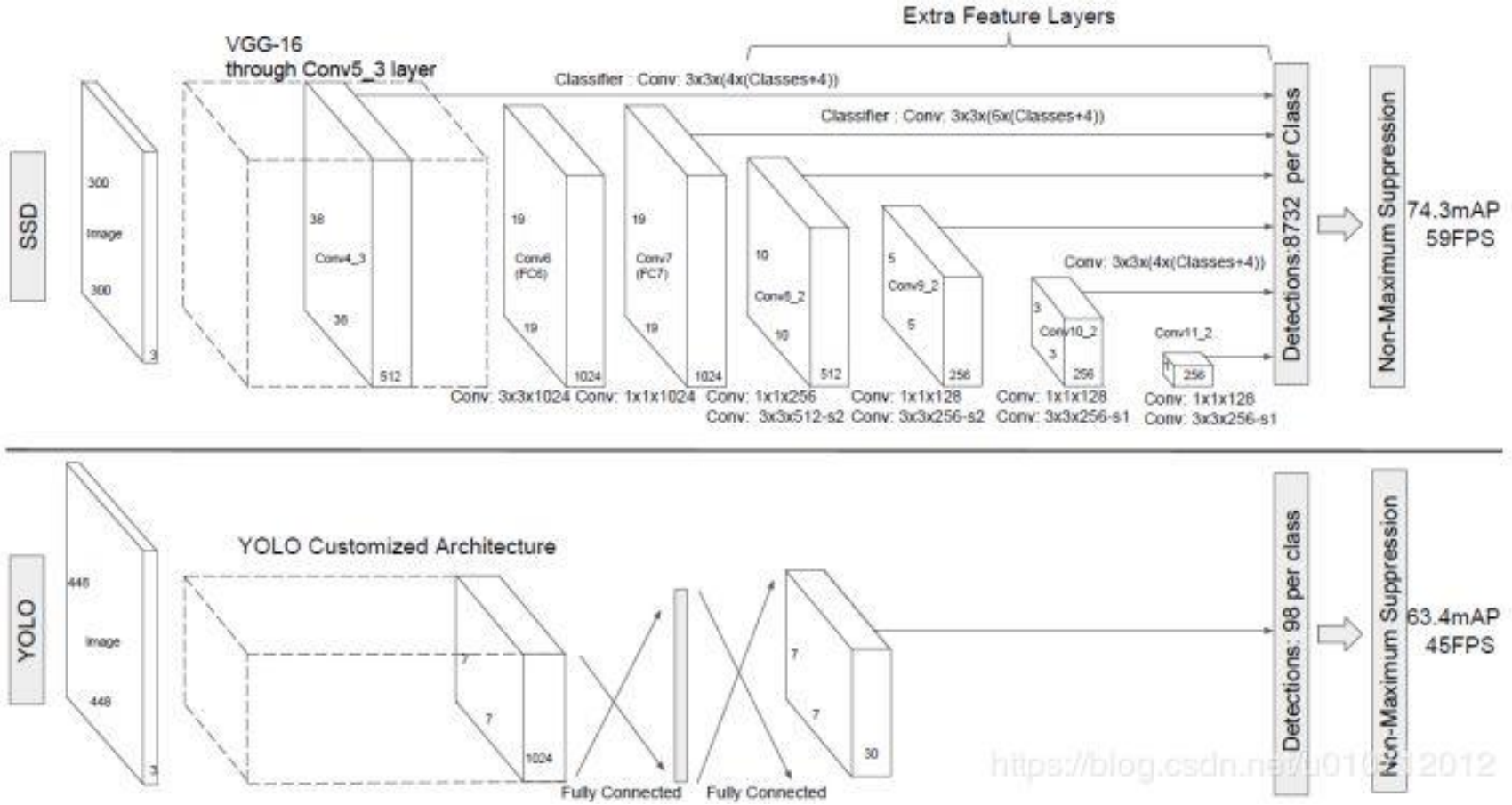
(b) 8×8 feature map



(c) 4×4 feature map

预测的是相对anchor的偏移量，同Faster R-CNN

Comparison with YOLO

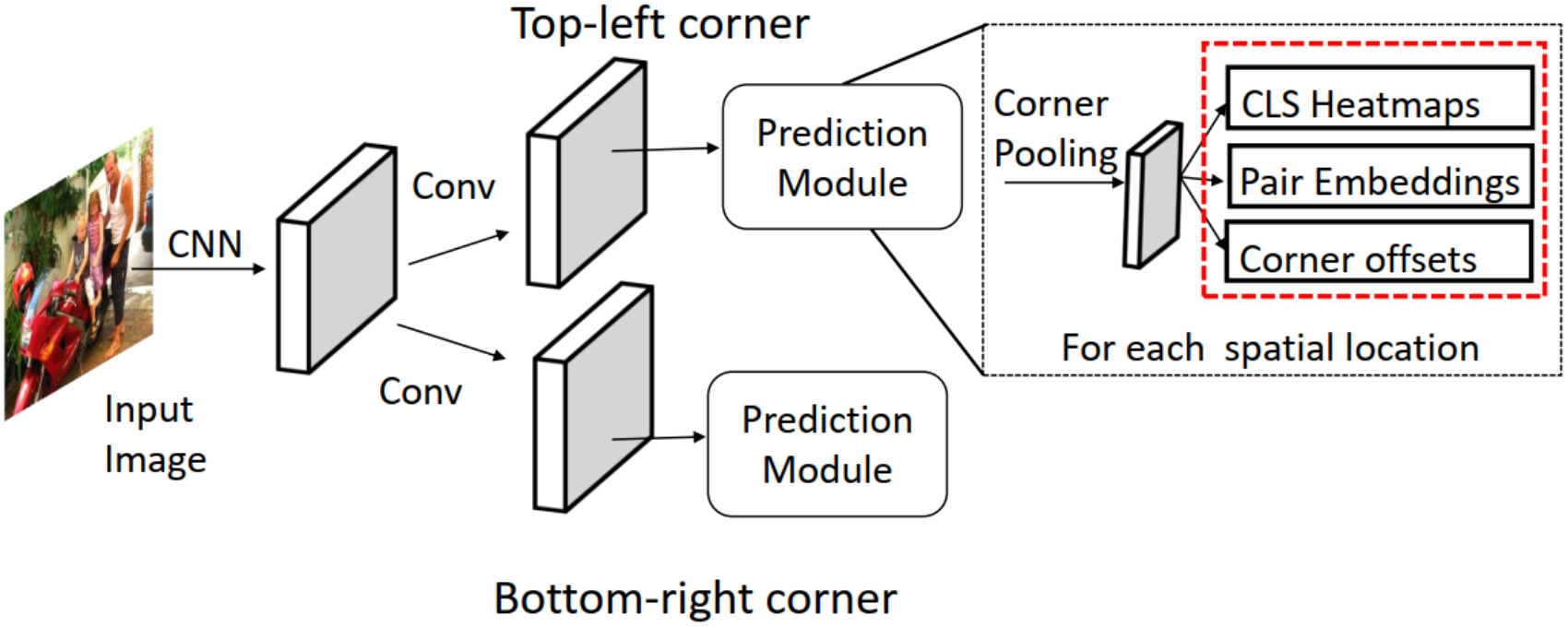


SSD（1-stage，anchor-based）

- SSD的缺陷及改进
  - 一阶段检测器由于没有Region Proposal，通常会存在正负样本不平滑的问题；
  - 改进分类损失，增大难样本的权重，降低简单样本的权重（Focal loss，ICCV 2017）
  - 通过特征图细化Anchors，过滤简单负样本，提高一阶段检测器的准确率（RefineDet，CVPR 2018）

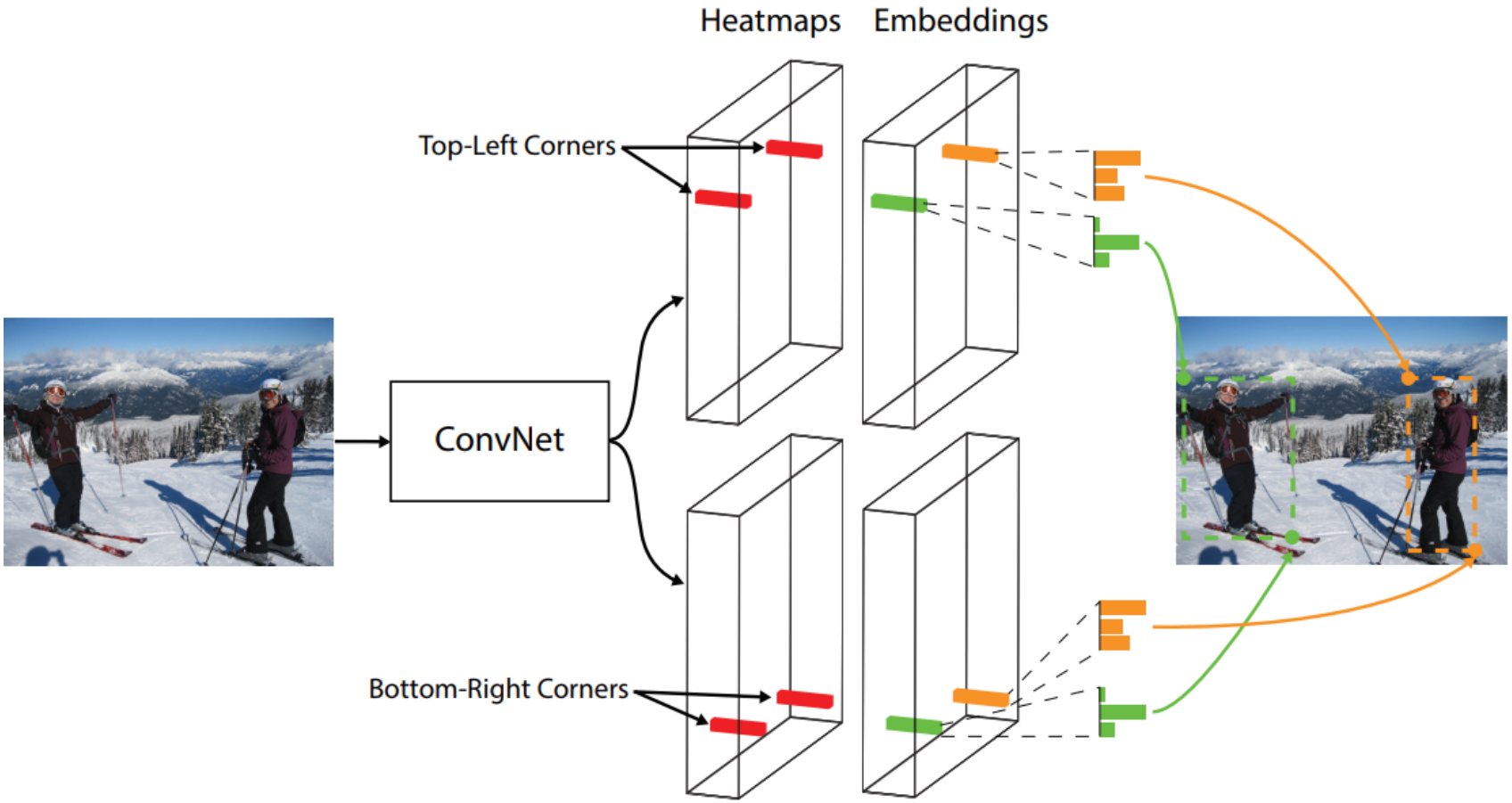
CornerNet（1-stage，anchor-based）

- Anchor-based以anchor为单位进行预测，anchor-free以特征像素点为单位进行预测



H. Law, J. Deng, Cornernet: Detecting objects as paired keypoints, in: *ECCV*, 2018.

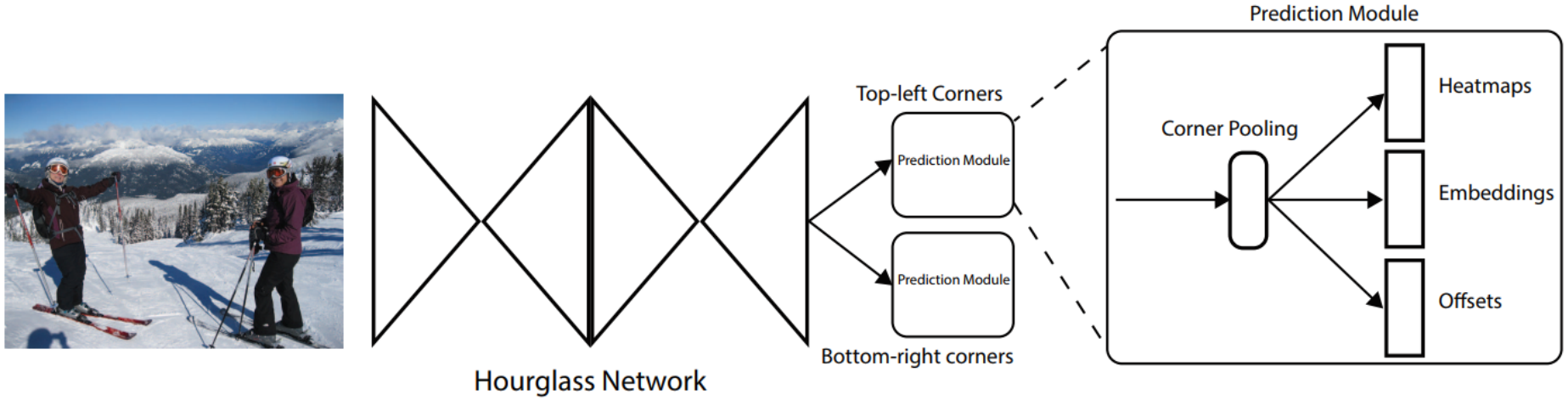
CornerNet（1-stage，anchor-based）



不同类别预测一个Heatmap，表示在该类别下，该点是左上角点/右下角点的概率，常用sigmoid

CornerNet（1-stage，anchor-based）

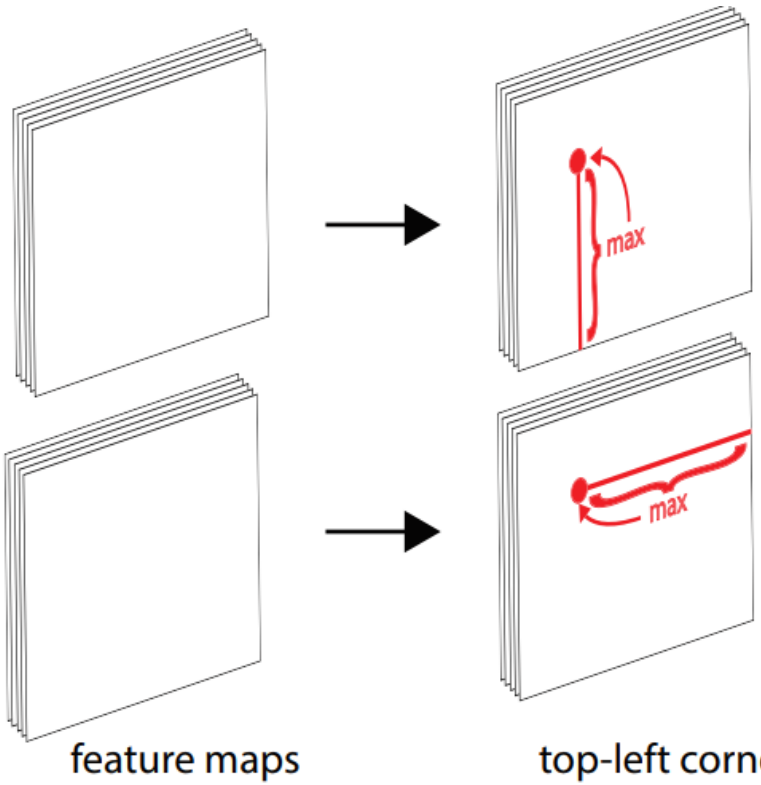
- Overview



该backbone常用于
人体关键点检测

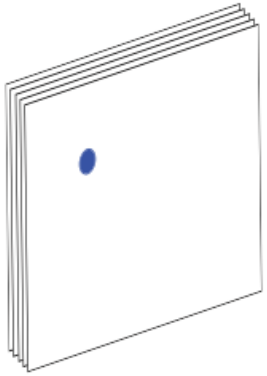
CornerNet（1-stage，anchor-based）

- Corner-pooling：用来处理角点在物体外的情况



垂直向下看，看到图像下边界，将最大值填入当前点，相当于找物体左边界

相加



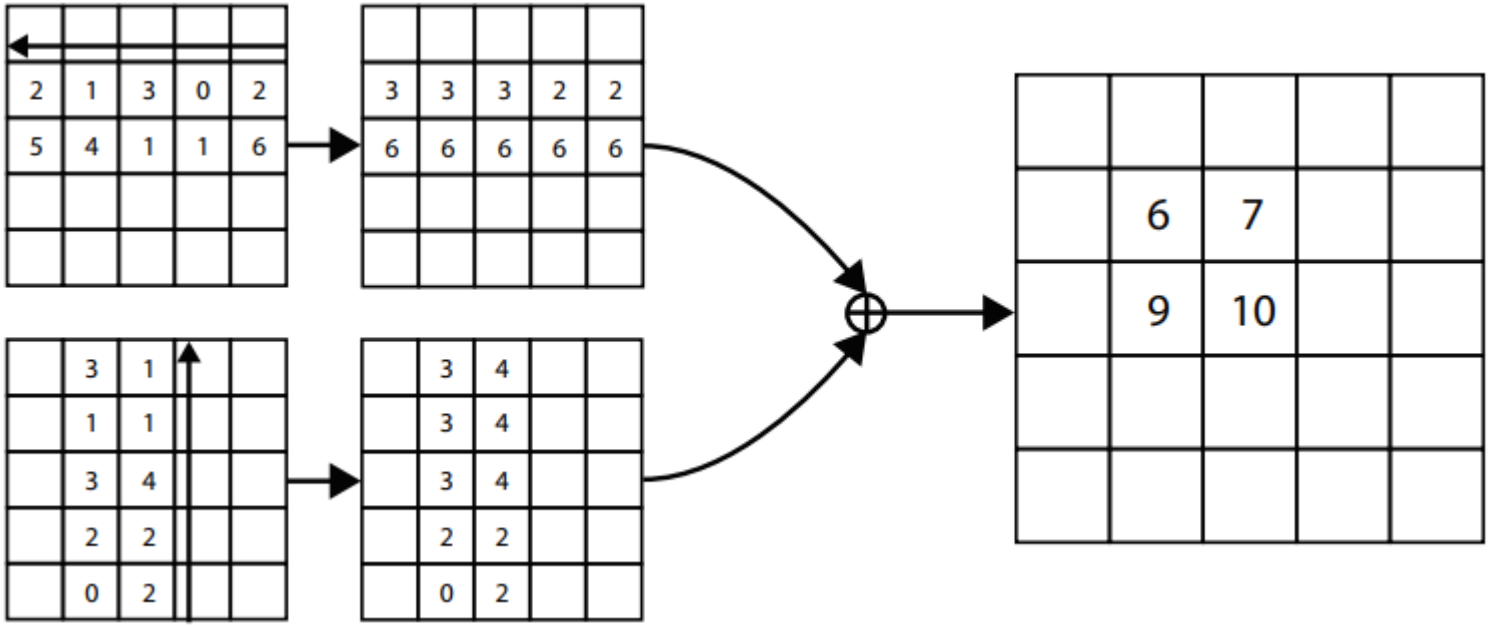
得到左上角点
(左边界与上边界的交点)

水平向右看，看到图像右边界，将最大值填入当前点，相当于找物体上边界

CornerNet（1-stage，anchor-based）

- Corner-pooling：用来处理角点在物体外的情况

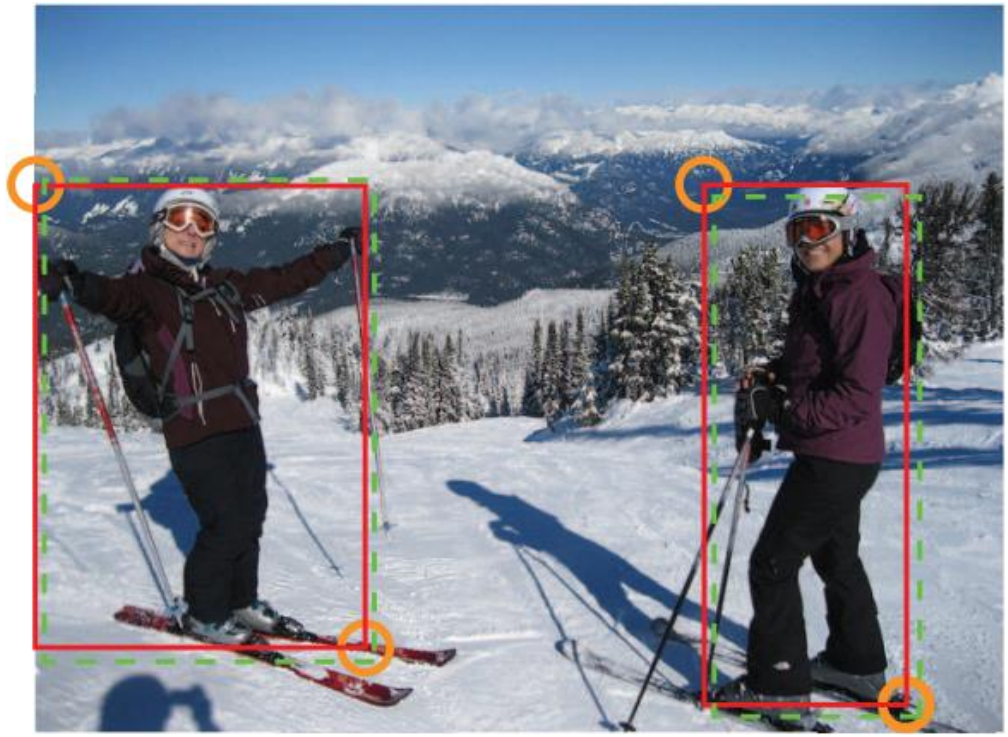
从右向左，每个点填入目前看到的最大值



从下向上，每个点填入目前看到的最大值

CornerNet（1-stage，anchor-based）

- Corner detection loss



由于ground-truth box只有一个（红框），所以ground-truth角点也只有一个，其它点都为负样本；

但在ground-truth box角点一定半径范围内（圆圈内）的点构成的框（绿色）与ground-truth box仍有很大的重叠度（大于0.7）；因此对这些负样本的惩罚应比其它负样本低

根据点到ground-truth角点的距离，被惩罚的权重由二维高斯定义：$e^{-\frac{x^2+y^2}{2\sigma^2}}$

$$L_{det} = \frac{-1}{N} \sum_{c=1}^{C} \sum_{i=1}^{H} \sum_{j=1}^{W} \begin{cases} (1-p_{cij})^{\alpha} \log(p_{cij}) & \text{if } y_{cij} = 1 \\ (1-y_{cij})^{\beta} (p_{cij})^{\alpha} \log(1-p_{cij}) & \text{otherwise} \end{cases}$$

基于focal loss, 1-y_{cij}是惩罚因子

CornerNet（1-stage，anchor-based）

- Embedding loss： 同一box角点的embedding要接近，不同box角点的embedding要远离

$$L_{pull} = \frac{1}{N}\sum_{k=1}^{N}\left[(e_{t_k} - e_k)^2 + (e_{b_k} - e_k)^2\right],$$

$$L_{push} = \frac{1}{N(N-1)}\sum_{k=1}^{N}\sum_{\substack{j=1 \\ j\neq k}}^{N}\max(0, \Delta - |e_k - e_j|)$$

e_{t_k}是第k个物体左上角点的embedding

e_{b_k}是第k个物体右下角点的embedding

e_k是e_{t_k}和e_{b_k}的平均

Embedding的具体值不重要，重要的是相对距离。这里的embedding为一维特征

- Total loss: $L = L_{det} + \alpha L_{pull} + \beta L_{push} + \gamma L_{off}$

heatmaps　　embeddings　　offsets

(smooth L1 loss)

Any Questions?