11 语义分割

像素预测、分辨率恢复与逐类评价

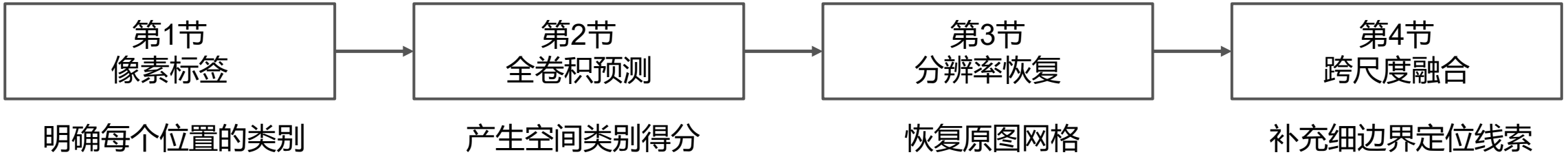
学习目标

- 解释标签掩码、类别通道与逐像素预测的对应关系。
- 跟踪FCN的空间尺寸、类别映射和分辨率恢复。
- 用完整示范说明转置卷积的重叠相加及非逆关系。
- 比较FCN与U-Net的跨层融合，核对拼接条件。
- 解释像素交叉熵，并从给定预测计算IoU与mIoU。

语义分割的整体逻辑

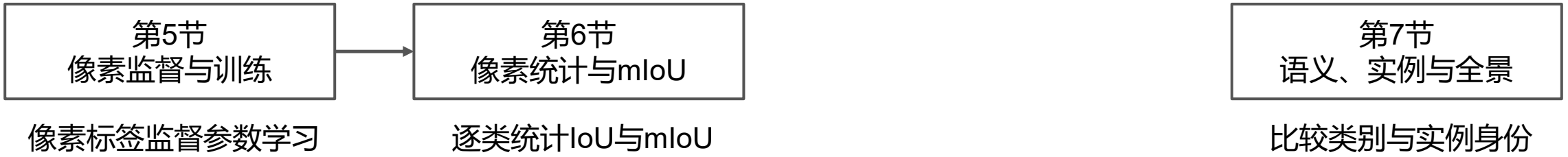
本讲问题地图：恢复网格后仍可能缺少细边界，因此继续引入跨尺度融合。

预测什么、怎样预测



结构建立后：怎样学习、怎样评价

任务扩展



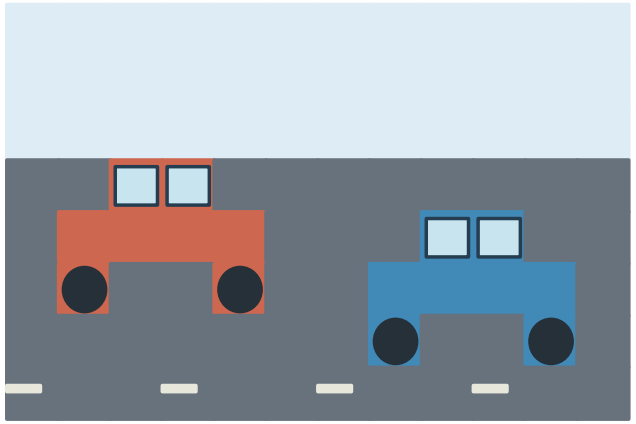
1 像素标签与分割任务

从整图判断转向每个位置的类别

每个像素属于哪一类

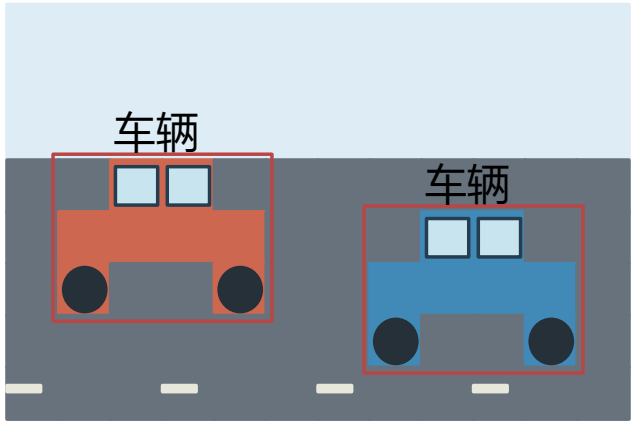
- **语义分割**为每个有效像素指定语义类别；检测框不能给出框内每个像素的归属。
- 两辆车同属“车辆”，语义标签不区分实例；类别集合与未标注区域由数据集规定。

输入场景



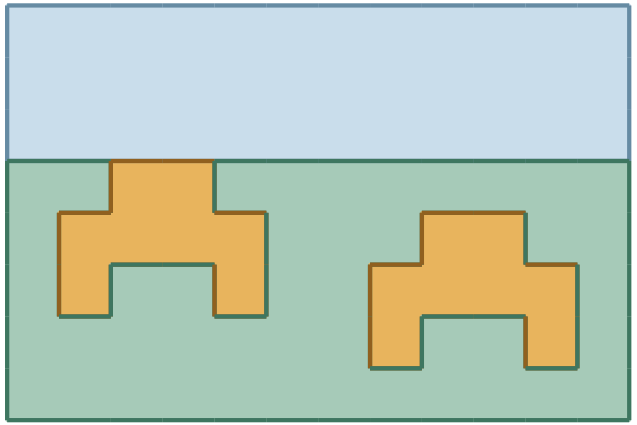
整图类别不提供每个位置的归属

目标检测



框内仍包含非车辆像素

语义分割



按像素区域给出类别

0 背景0（含天空）

1 道路1

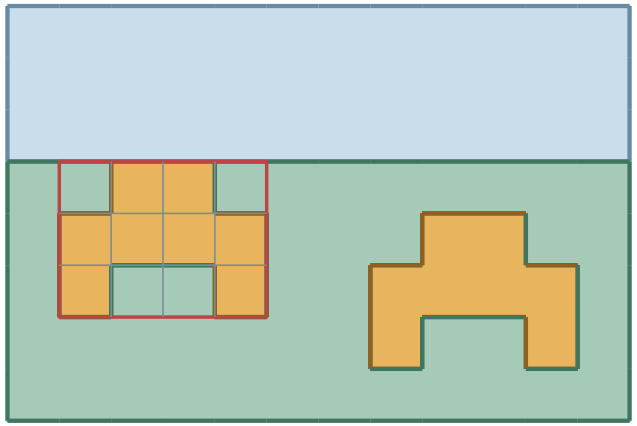
2 车辆2

原生简化道路场景；非真实照片或训练结果。两辆车同类，不编码实例身份。

标签掩码与显示颜色

- 输入 $\boldsymbol{X} \in \mathbb{R}^{3\times H\times W}$；**标签掩码**在同一网格保存类别ID：$Y_{u,v} \in \{0,\dots,C-1\}$。
- u, v 是行、列，C 是类别数；显示颜色通过既定映射得到，RGB标签须先转换为ID。

彩色标签（共同场景）



红框局部：类别ID

1	2	2	1
2	2	2	2
2	1	1	2

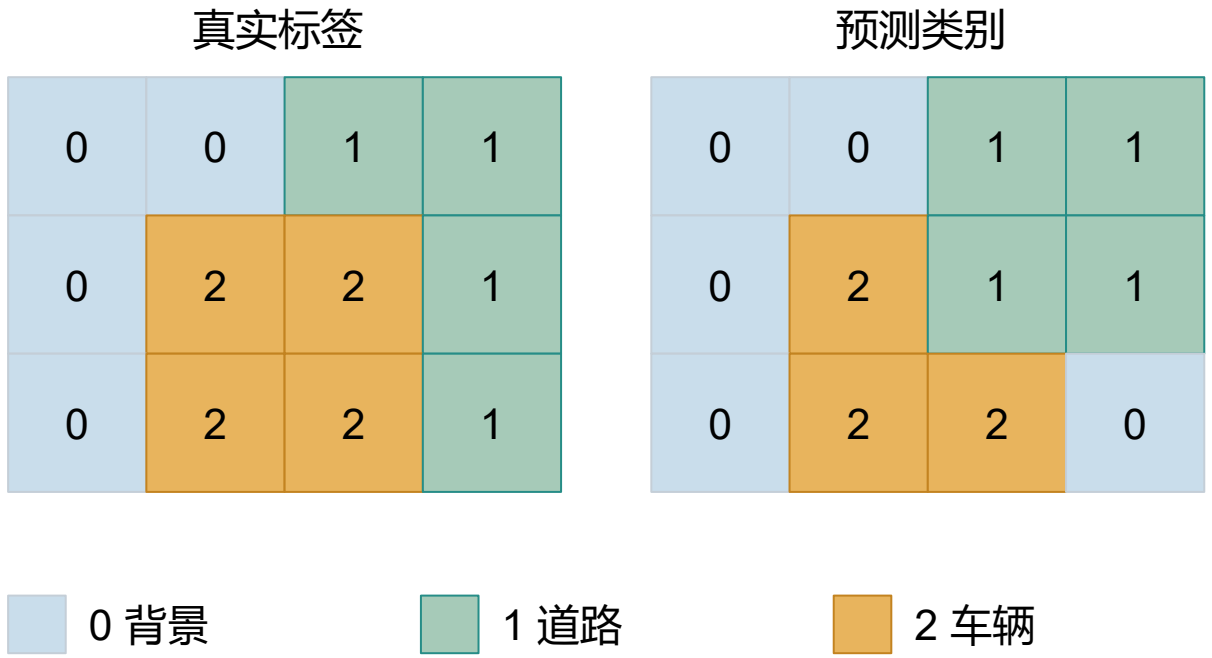
固定显示映射

- 0 → 背景色
- 1 → 道路色
- 2 → 车辆色

同一位置先有类别ID，再映射为显示颜色；颜色值不参与类别编号运算。

红框为场景网格第4–6行、第2–5列；与后面的12像素计算例是不同构造。

一组12像素的标签与预测



- 三类：背景0、道路1、车辆2。
- 数字是类别ID，颜色只帮助识别类别；所有12像素均有效。
- 预测有两处错误：第2行第3列、第3行第4列。
- 这里的预测是教学构造，预留作后续12像素主例；用于混淆矩阵与mIoU计算。

图像与标签必须同步变换

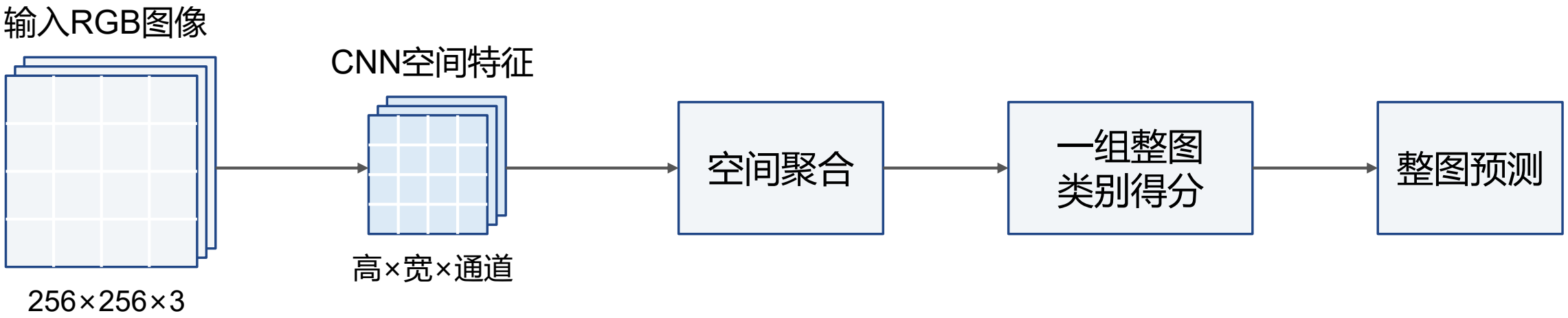
- 每个标签必须仍对应输入中的同一位置。
- 裁剪、翻转、旋转等空间变换须同时作用于图像与掩码，并使用同一几何参数。
- 图像缩放可以使用双线性插值；离散类别掩码应使用最近邻等保留类别的方法。
- 仅改变亮度、对比度等外观时，不改变标签类别。
- 数据划分按原始样本或场景完成，避免相关裁剪跨入训练与验证集。
- 目标是与图像对齐的类别网格，网络也须产生逐位置得分。

2 全卷积的空间预测

目标是像素标签图；现在构造保留空间对应的类别得分。

整图分类为什么不能直接完成语义分割

- CNN仍能提取空间特征；整图分类头将空间信息汇总为一组整图类别得分。
- 一个整图决策不能给出各像素归属；语义分割需要保留位置对应，产生空间类别得分图。



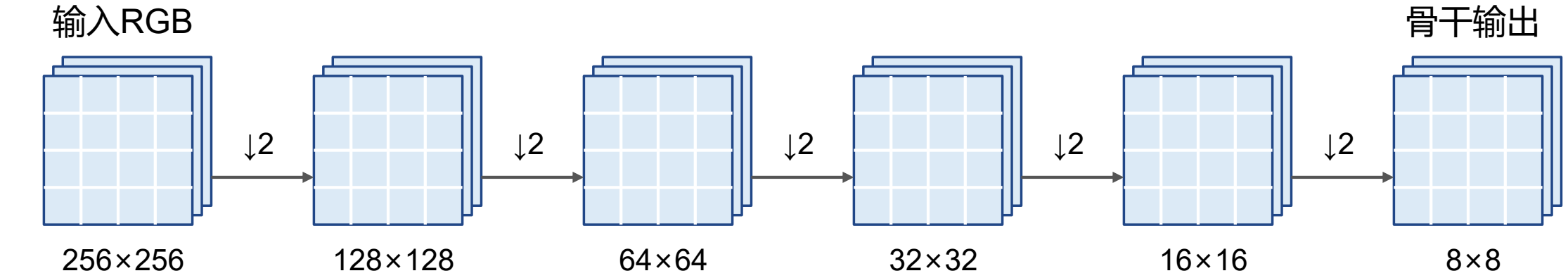
整图分类：每图一次类别决策

语义分割：H×W个位置各有类别

怎样利用CNN空间特征，为每个位置产生类别得分？

CNN骨干为什么产生低分辨率特征?

- 五次空间减半形成步幅32；步幅卷积或池化都可实现。
- 语义特征来自多层学习，下采样本身不保证语义理解。



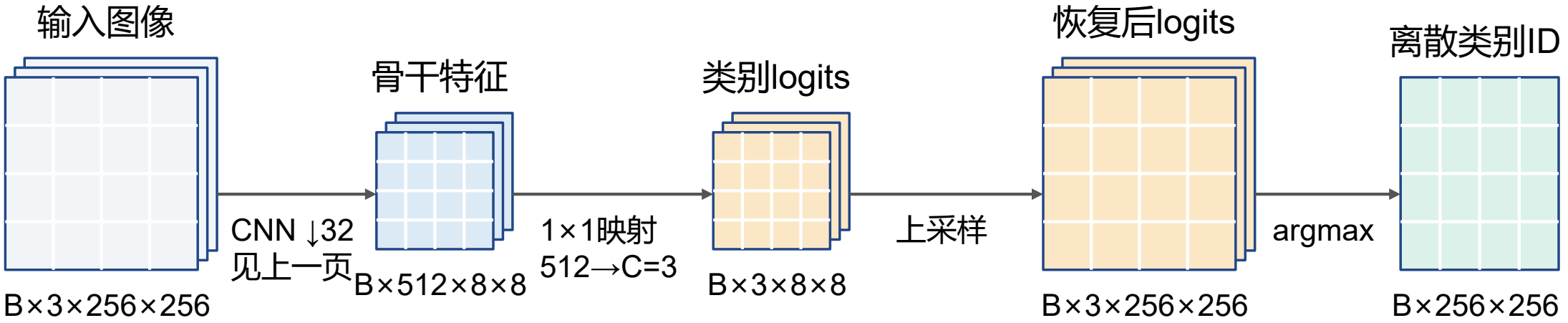
五次空间减半：2×2×2×2×2=32，输出步幅32。

更广上下文	较低后续空间计算	不同尺度表示

通道扩展用于表达多种特征；512由设计选择，不由空间减半自动推出。
代价：定位网格变粗；仅示尺度演化，非具体骨干逐层复现。

FCN如何产生逐像素预测

- FCN保留空间网格，以共享卷积生成类别得分，再恢复到目标分辨率。
- 上采样后仍是连续logits；沿类别轴argmax后才得到离散类别ID。



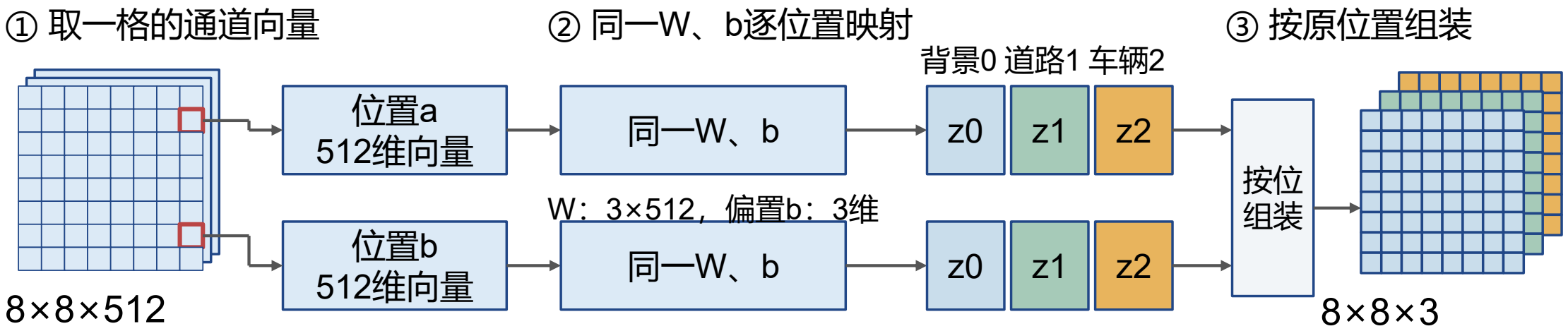
1×1卷积怎样把每个位置的512维特征变成3个类别得分？

教学构造；叠层只表示通道维，不按通道数或空间尺寸比例绘制。

1×1卷积：从512维特征到3个类别得分

放大上一页FCN的类别映射：取一个样本，图内形状按高×宽×通道表示。

D=512为输入特征通道；C=3：背景0、道路1、车辆2；$h = w = 8$，共64处。



全部64处使用同一组W、b；按位置组装，不作空间汇总；叠层不按通道数比例绘制。

计算关系：$\boldsymbol{z}_{u,v} = \boldsymbol{W}\boldsymbol{f}_{u,v} + \boldsymbol{b}$；$\boldsymbol{f}_{u,v}$含512个输入特征值。

逐类为$z_{c,u,v} = \boldsymbol{w}_c^{\mathsf{T}}\boldsymbol{f}_{u,v} + b_c$；$\boldsymbol{w}_c, b_c$是第$c$类权重与偏置。

类别含义由监督标签建立，logits不是概率。1×1指空间核大小，8×8网格不变；再核对全链。

沿FCN核对空间尺寸与通道变化

阶段	输出形状：批量×通道×高×宽	变化原因
输入图像	$B\times3\times256\times256$	RGB输入
CNN骨干	$B\times512\times8\times8$	提取特征，空间下采样32倍
1×1类别映射	$B\times3\times8\times8$	通道512→3，空间不变
上采样恢复	$B\times3\times256\times256$	恢复网格，保留3个得分通道
类别轴argmax	$B\times256\times256$	选类别ID，去掉类别通道维

B为批量大小。先核对哪个维度改变及原因；下一页区分logits、概率与ID，不在此重复概率计算。

类别得分、概率与预测掩码

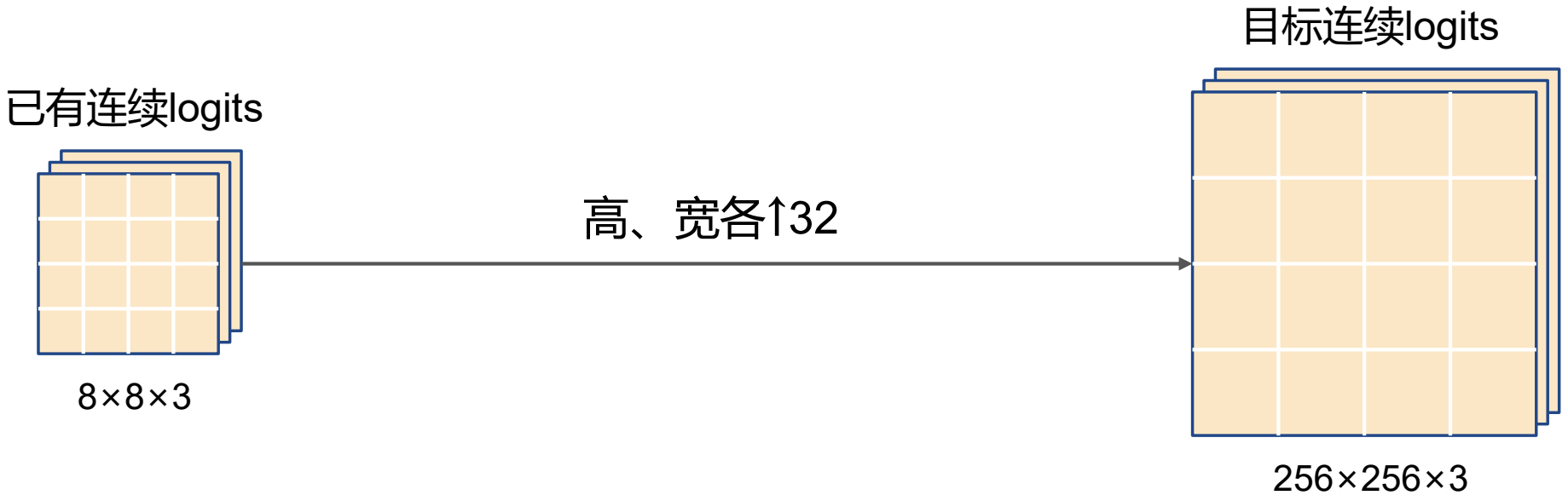
- 每个像素有 C 个实数得分，称为logits；得分可以为负，不要求和为1。
- 对同一像素的类别轴作softmax，才得到各类概率。
- 例：一个像素的概率为 $(0.1, 0.7, 0.2)$，预测类别为1。
- **argmax**返回最大得分对应的类别ID，同一位置作softmax不改变最大者。
- 先将8×8得分图恢复至256×256再argmax；提前转成ID会丢失各类得分差异。

3 分辨率恢复与转置卷积

已有8×8粗得分图；现在展开它怎样恢复到256×256标签网格。

先明确空间网格的恢复任务

- 8×8的连续类别logits要恢复到256×256标签网格，三个类别通道不变。
- 本节不恢复RGB图像，也不对离散ID作双线性插值。

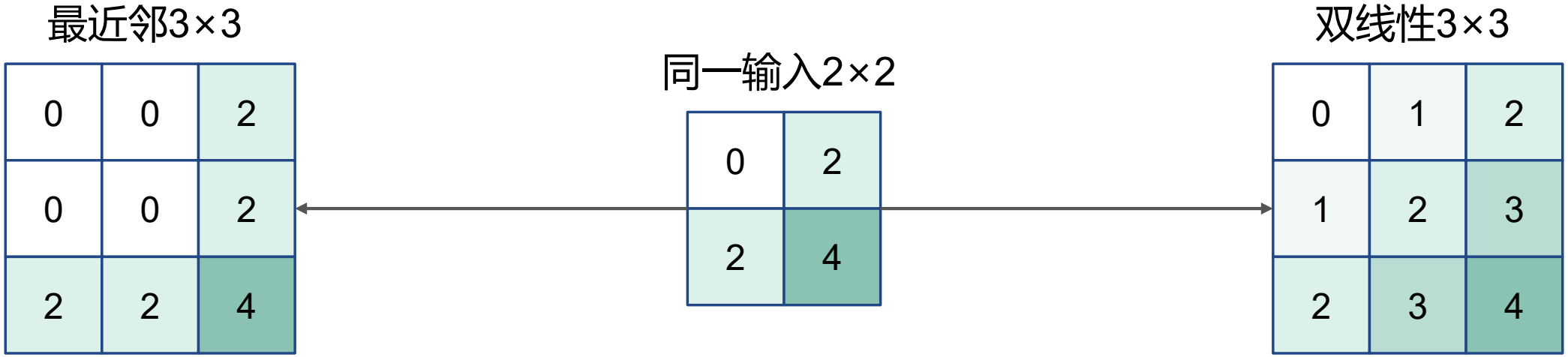


C=3保持；先恢复连续得分，argmax在后。省略批量，形状按高×宽×通道。

怎样为新增位置赋值？先看固定插值规则。

最近邻与双线性插值

- 图中0—4为连续值示例，非类别ID；色深表示数值大小。
- 最近邻取距离最近的采样点；双线性按二维距离对相邻4点加权，适用于连续值。
- 类别ID只是编号；1和3的平均2不代表中间语义类别，掩码缩放须保留离散类别。



输入坐标：0、1；输出坐标：0、0.5、1（两个方向相同）。最近邻等距时取较小坐标。

中心值=(0+2+2+4)/4=2；此教学取样约定不代表所有框架的边界对齐。

固定规则给出位置与权重；能否通过训练学习恢复权重？

固定插值与可学习恢复有什么区别?

- 固定插值依据采样规则赋值；转置卷积使用可训练的核权重。

固定插值	可学习恢复
最近邻 / 双线性	转置卷积
采样规则确定位置与权重	输入值×核权重，分配后相加
无待训练的插值核	核权重随损失反向传播更新

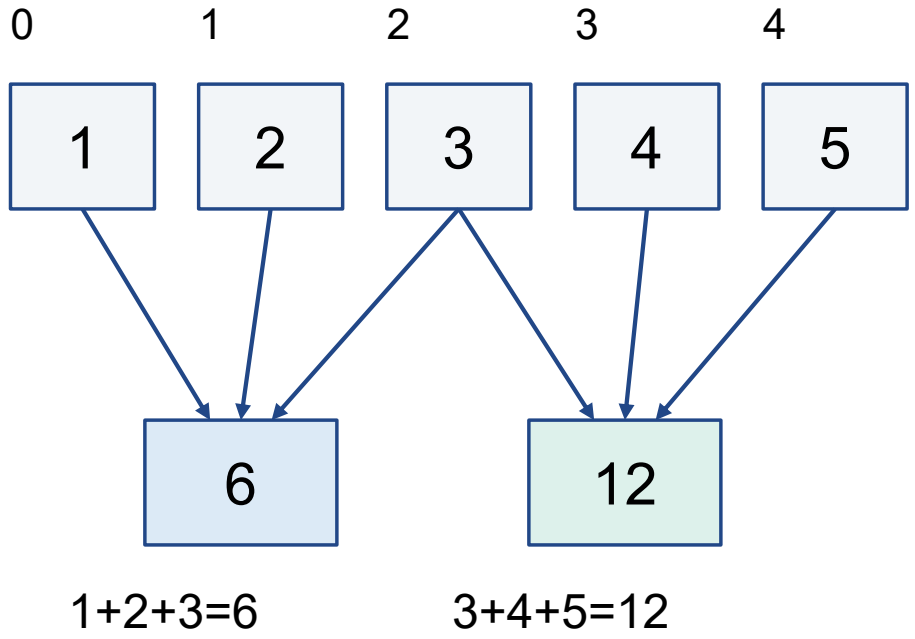
共同要求：目标尺寸与坐标对齐；裁剪 / 填充及训练、推理设置一致。

可学习不保证更好，要在相同数据与评价条件下比较。下一页看核怎样分配贡献。

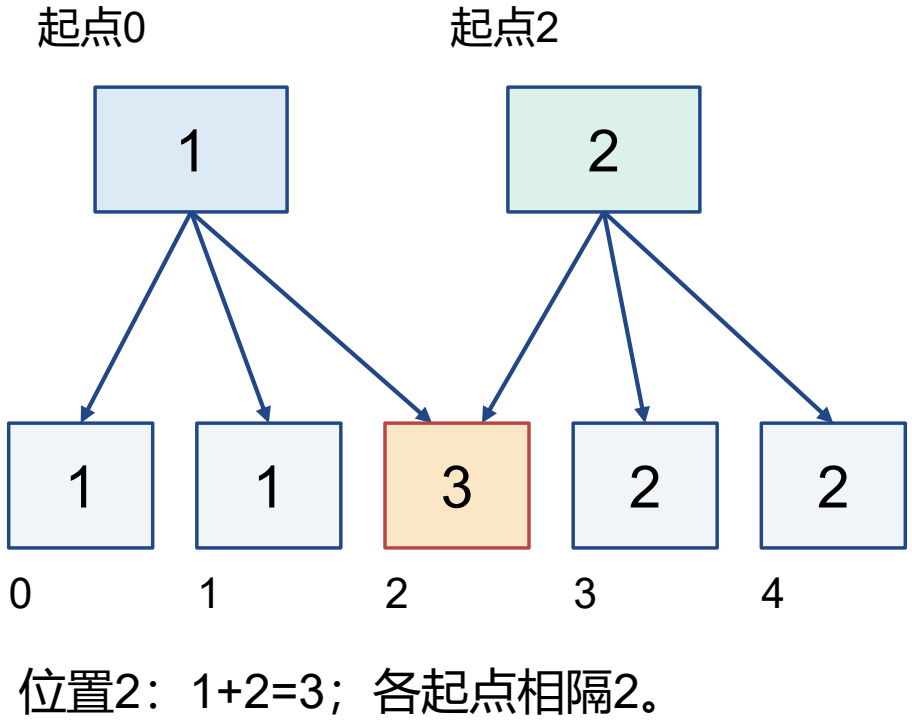
普通卷积与转置卷积：聚合和分配

- 共用核[1,1,1]、步幅2，无填充、无偏置；普通卷积汇聚，转置卷积分配。
- 左侧5个输入得到2个输出；右侧2个输入得到5个输出。改变映射方向不等于求逆。

普通卷积：多个位置汇聚



转置卷积：单个输入分配

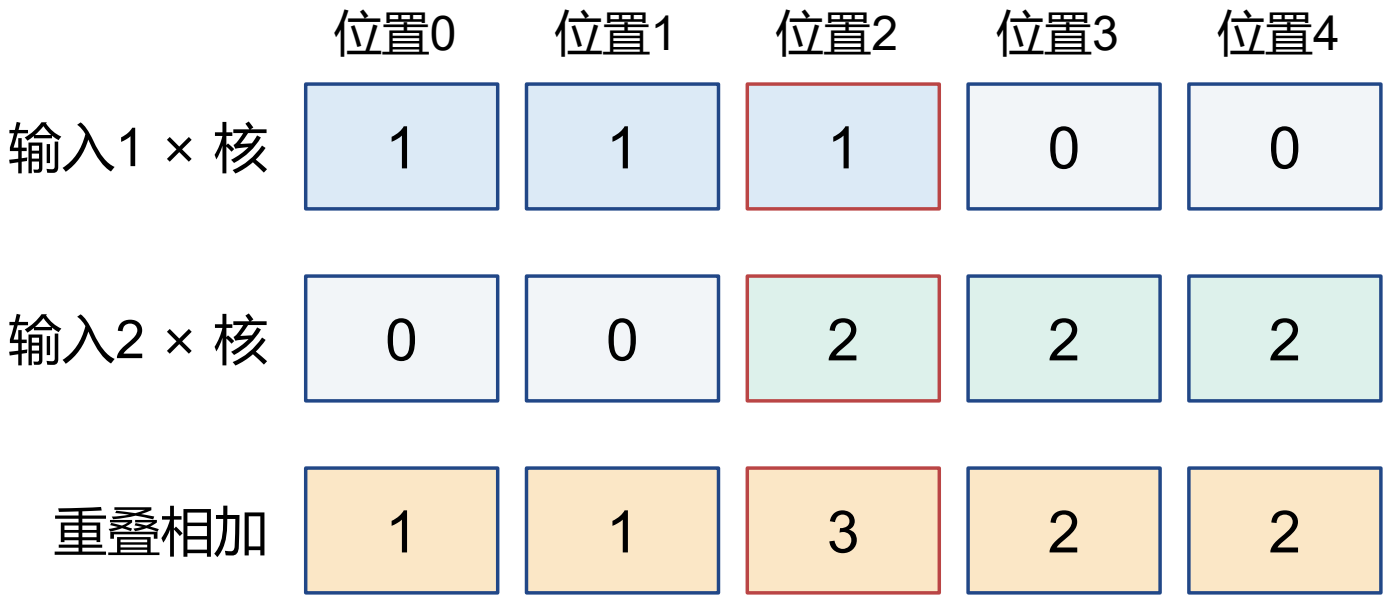


位置2：1+2=3；各起点相隔2。

共用核[1,1,1]、步幅2，无填充和偏置；左右输入不同，只比较覆盖动作，不是互逆。

一维示范：散射贡献并重叠相加

- 输入 $[1,2]$，核 $[1,1,1]$，步幅2、无填充、无偏置。
- 沿上一页的贡献带逐列相加；位置2收到两次贡献：$1+2=3$。



覆盖起点：0、2
步幅决定起点间距

位置2：1+2=3

本例核取[1,1,1]便于算数；实际核可随像素损失反向传播更新。

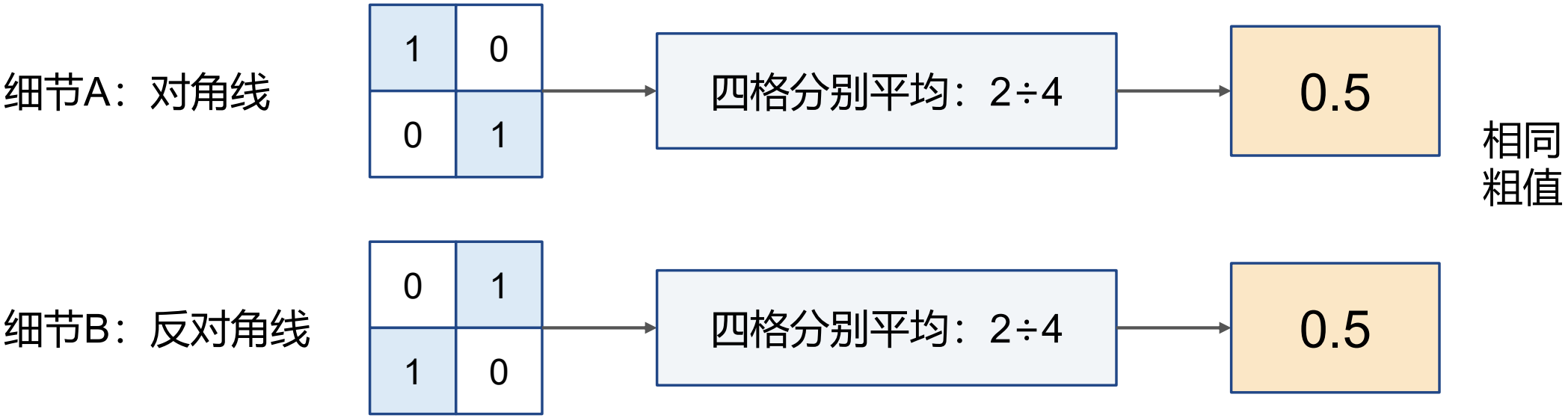
名称来自矩阵转置，非逆运算（讲末选讲）；主讲下一页核对尺寸。

输出尺寸由哪些参数决定

- FCN尺寸核对：将8×8得分图恢复为256×256。
- 对高或宽的任一方向，输出长度为：
  - $L_{\text{out}} = (L_{\text{in}} - 1)s - 2p + d(k-1) + o + 1$
- s 为步幅，p 为padding参数，d 为核内采样间隔，k 为核大小。
- o 为output_padding，用来消除输出长度的歧义；它不表示在输出后补零。
- 例：输入长8，取 $s = 32, k = 64, p = 16, d = 1, o = 0$。
  - 输出长为 $7\times32 - 32 + 63 + 1 = 256$。
- 高宽相同只表示位置数量一致；空间对应还取决于采样坐标、裁剪和填充。

恢复尺寸后，细边界一定回来吗？

- 两个不同的2×2二值细节块，经同一平均操作都得到0.5。
- 一个粗值对应多种细节，无法据此唯一还原原块；扩大网格不能消除该歧义。



只知道0.5，无法判定原来是A还是B。

块平均仅为信息丢失反例；不代表CNN必然这样下采样，也不是对类别ID插值。

能否利用编码过程中保留的较高分辨率特征，补充定位信息？

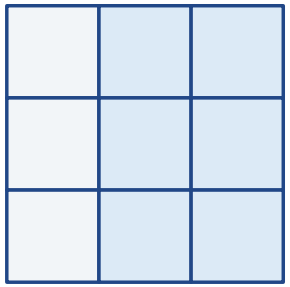
4 跨尺度融合与U-Net

粗得分上采样难恢复细边界；恢复分辨率时引入浅层信息。

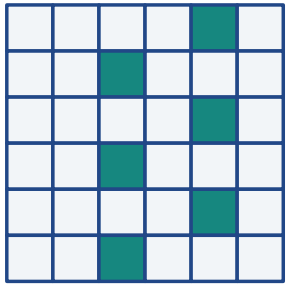
为什么需要跨尺度融合?

- 深层上下文与浅层定位互补；先找同一骨干的三个尺度。
- 各尺度先经1×1映射到C类得分，再比较融合路径。

为何补充：同场景的互补线索

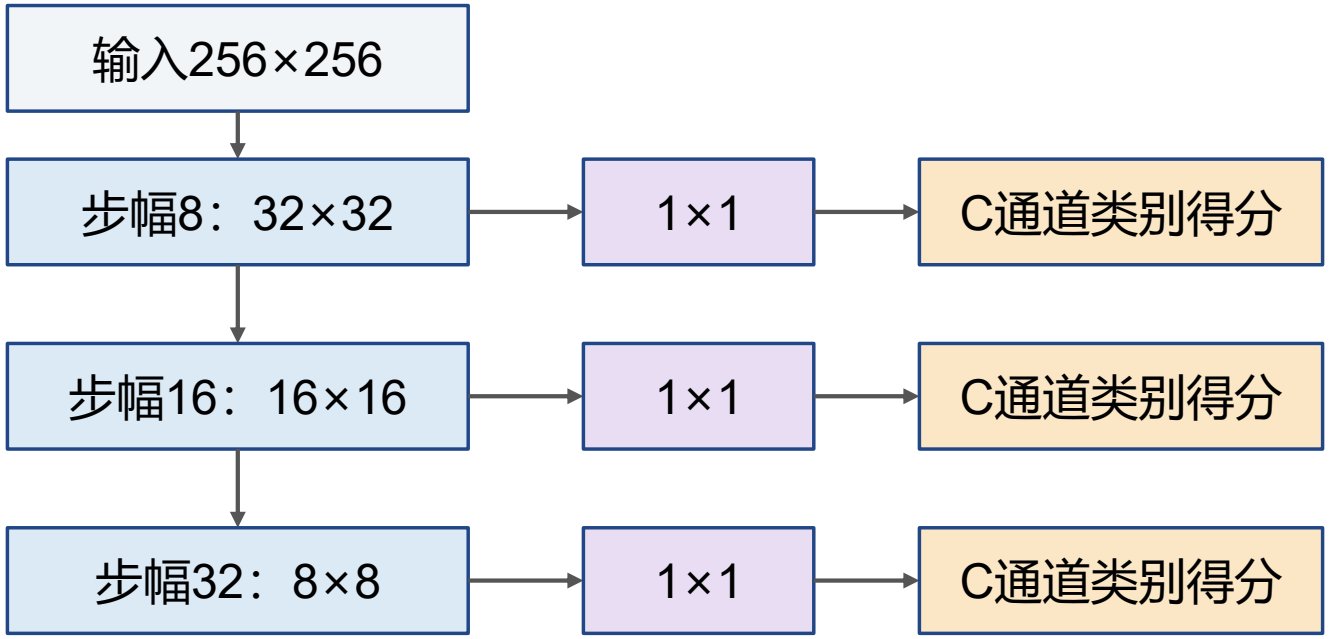


深层上下文
定位网格较粗



浅层位置
单独不保证语义

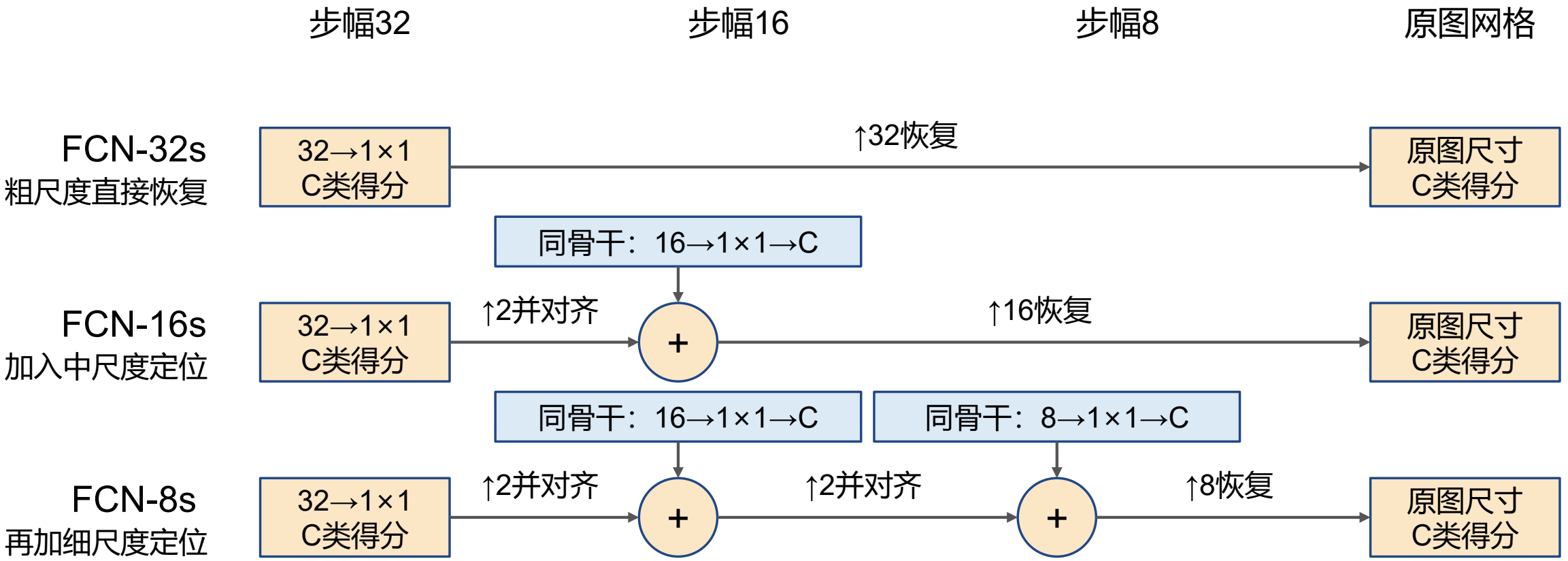
信息来自哪里：同一CNN骨干



左侧为线索示意；右侧三分支来自同一骨干。下一页比较融合哪些尺度的得分。

FCN如何逐步融合更细尺度的预测?

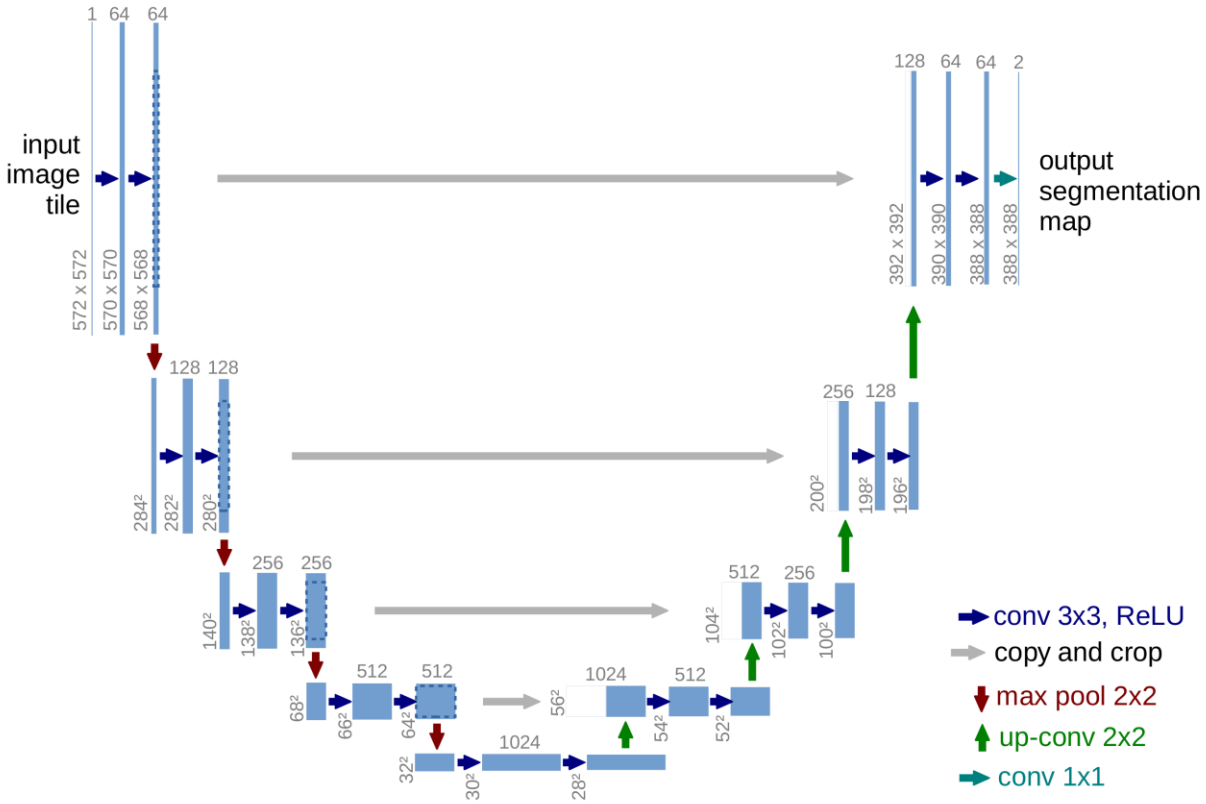
- 三种方案共用上一页的CNN骨干：32、16、8分别表示相对输入的采样步幅。
- ①仅用32步幅得分；②加入16步幅得分；③再加入8步幅得分。新增的是信息来源。



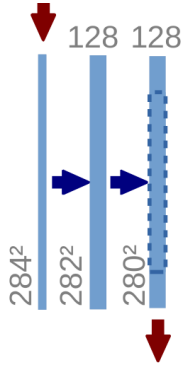
相加前空间坐标与类别通道对齐；不同融合尺度不保证性能递增。Long等，2015（教学简化）

经典U-Net：编码器—解码器与跳跃连接

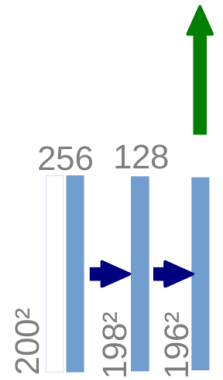
原图572×572×1→388×388×2；valid卷积缩小尺寸，skip复制裁剪后拼接。



① 编码器
卷积 + 池化



② 解码器
上采样 + 卷积



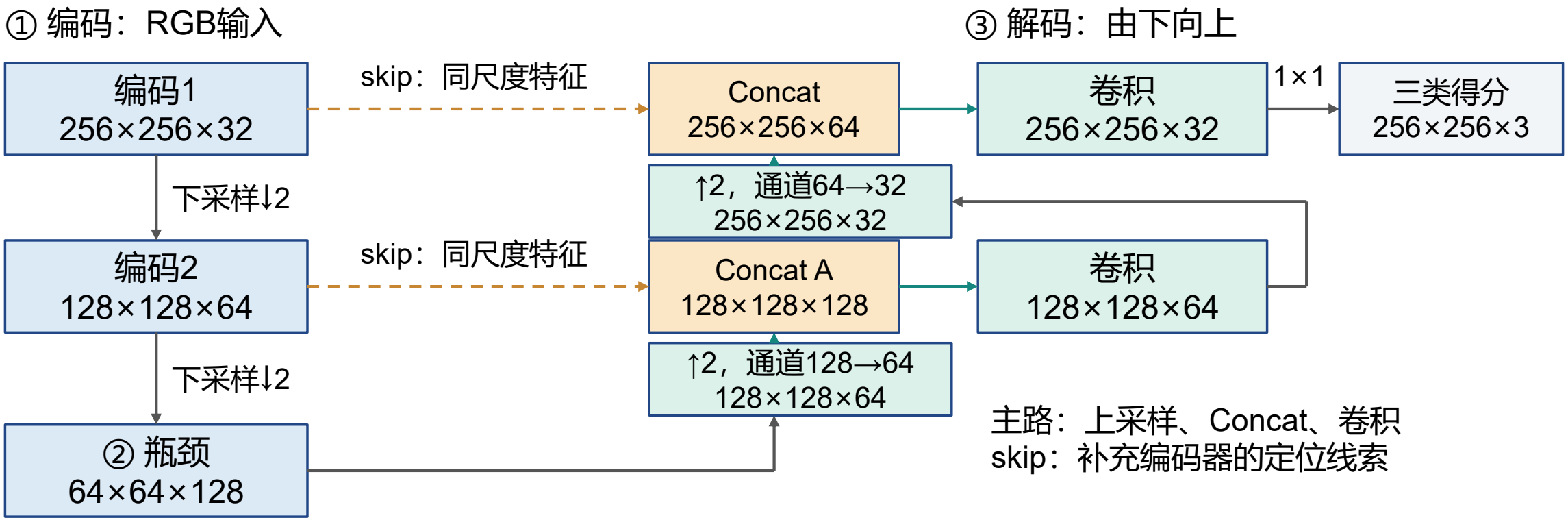
③ 横向skip：复制裁剪后拼接



Ronneberger, Fischer, Brox | MICCAI 2015 | Figure 1 | arXiv:1505.04597v1

U-Net如何逐级解码并融合特征?

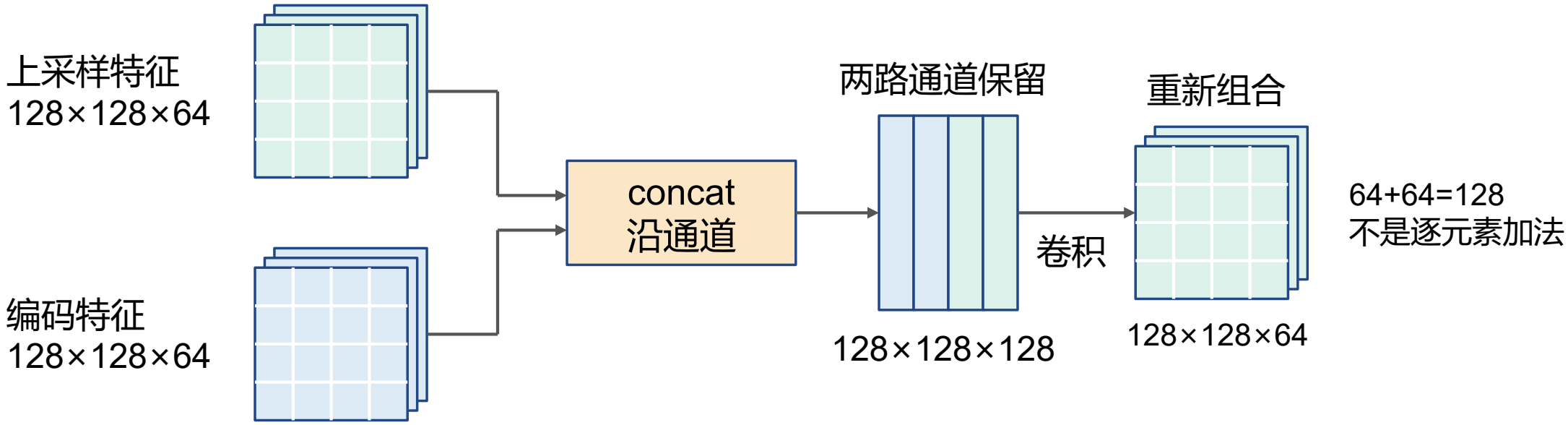
- 教学构造：256×256 RGB、三类、三尺度、same-padding。
- 先按①②③走主路，再看横向skip；每一级解码依次上采样、Concat、卷积。



A对应下页128×128放大；same-padding教学构造，非原图尺寸｜Ronneberger等，2015

Concat：沿通道拼接两组特征

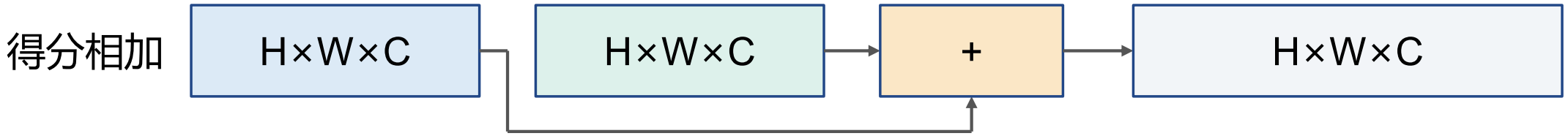
- 放大上一页A：128×128尺度的concat；两路空间尺寸与坐标先对齐。
- 拼接保留两组通道，后续卷积才学习组合；不是逐元素加法。
- 第二级同理：256×256×32 与同尺度32通道拼接为64通道，再卷积为32通道；最后1×1映射为3类。



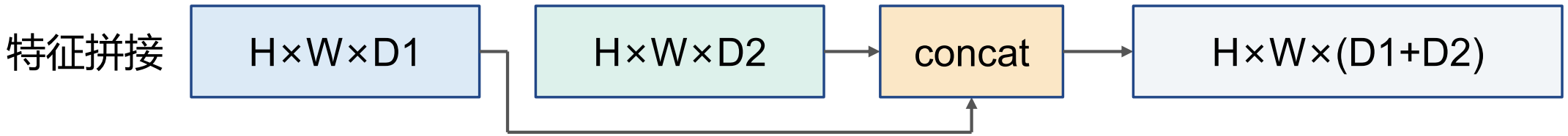
对应上一页128×128节点A；H、W不变。叠层只示意分组，不按通道数比例绘制。

跨层融合：类别得分相加与特征拼接

- 回看FCN与U-Net图：前者相加同位置、同类别得分，后者拼接特征，二者不是同一运算。
- 残差加法也要求形状匹配，可先投影。“跳跃连接”描述跨层路径，不能据此确定融合算子。



相加后通道数不变；空间位置、类别身份及形状须匹配。



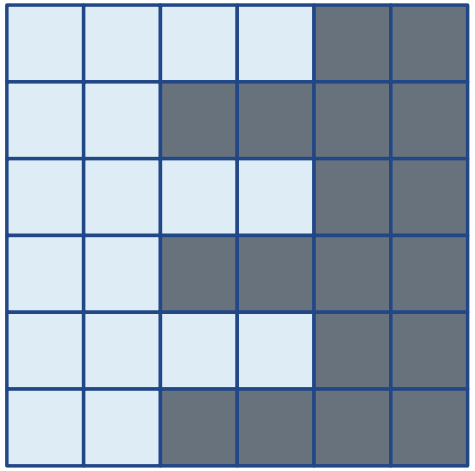
拼接要求高宽及坐标对齐，通道数可以不同；输出保留两组通道。

这里比较经典FCN得分融合与U-Net特征融合；U-Net本身也属于全卷积网络。

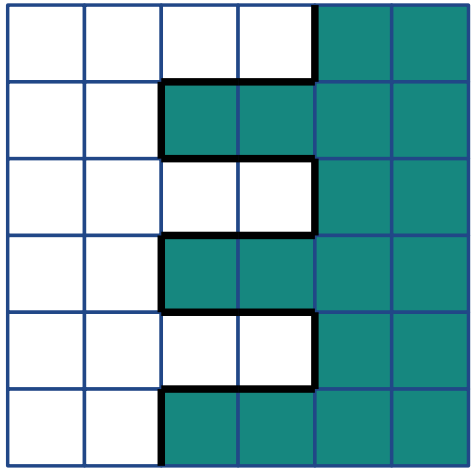
跨尺度融合怎样改善边界定位?

- 同一细网格上，粗预测的大区域大致正确，但细边界错位；融合示意更接近真值。
- 黑色轮廓始终表示真实边界，红框标出错分格；融合仍留一格错误。

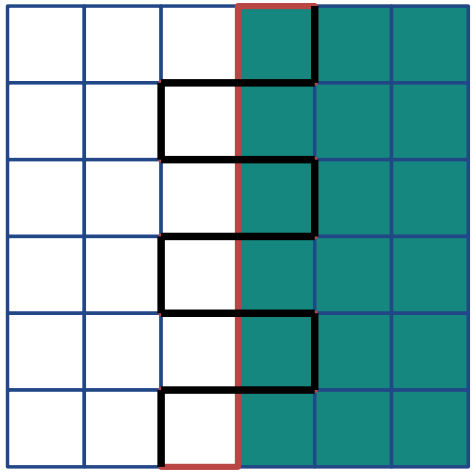
同一输入外观



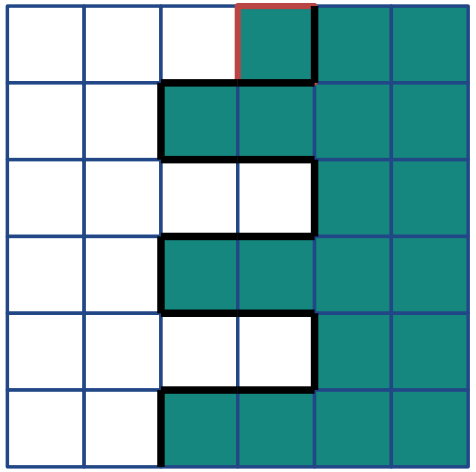
真实边界



仅凭粗特征恢复



融合浅层线索后



黑线：同一真实边界。红框：错分位置；融合示意仍留第1行第4列错误。

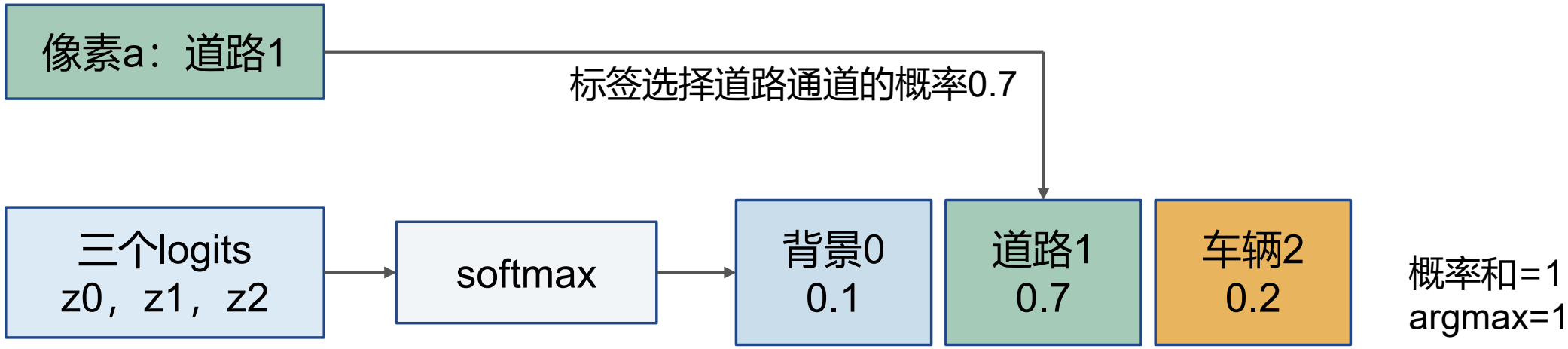
机制示意，非模型实验；额外定位线索可能减少歧义，实际改善仍依赖训练。

5 像素监督与训练

预测结构已建立；现在用像素标签学习其中的参数。

如何判断一个像素的预测是否正确?

- 沿用道路像素a：真实类别为道路1；C=3，对应背景0、道路1、车辆2。
- logits是未归一化得分；固定像素 (u, v)，沿类别轴计算：$p_{c,u,v} = \frac{\exp(z_{c,u,v})}{\sum_{j=0}^{C-1} \exp(z_{j,u,v})}$

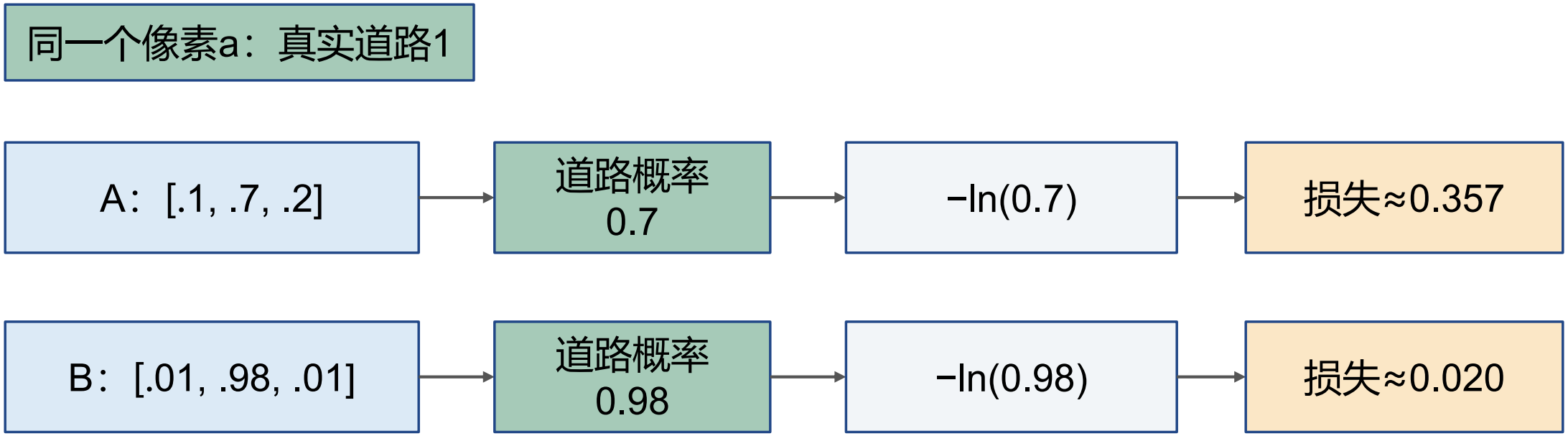


argmax给出类别ID；训练还要利用分配给真实类别的概率。下一页仍看像素a。

概率是给定示意值；未给定具体logits，不将它们当成待验softmax数值。

为什么使用交叉熵?

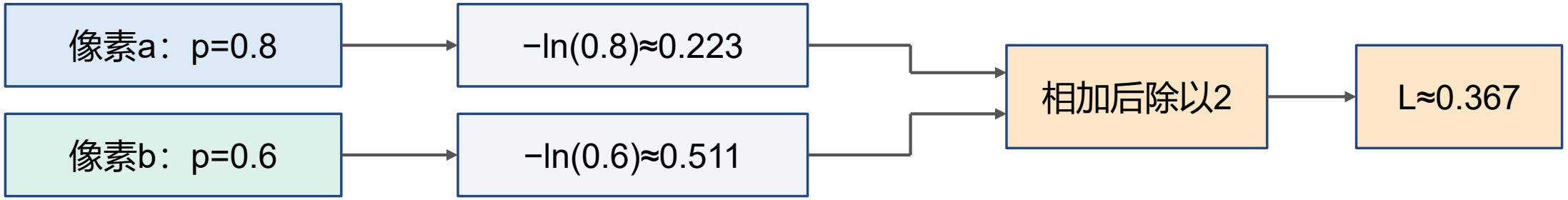
- 同一个道路像素a，真实类别仍为1；两组预测的argmax都为1。
- 用真实类别概率定义损失：$L = -\ln(p_y)$，这里y=1。



两者argmax都为1；交叉熵仍区分给定真实标签的概率分配，不代表总体可靠性。

整张图像的交叉熵怎样计算?

- 从道路像素a扩展到两个有效像素：本例a、b的正确类概率分别设为0.8、0.6。



独立两像素构造：两格均有效，各自进入损失总和与平均分母。

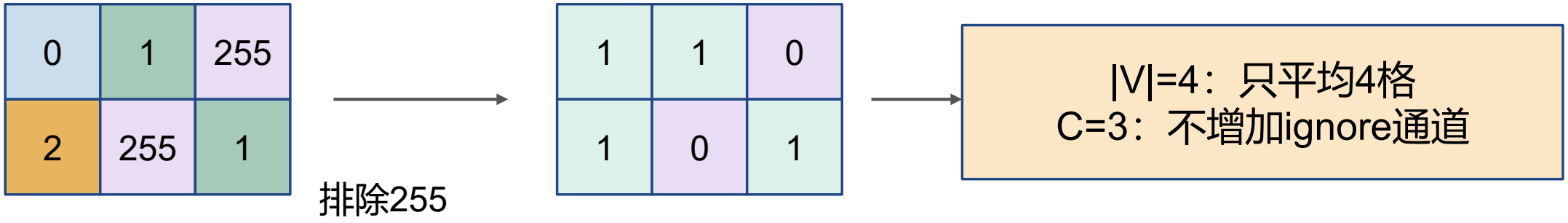
一般写为：

- 有效集合 V 排除忽略像素，$|V| > 0$；Y为真实类别，p为该类概率。
- $\ell_{u,v} = -\ln p_{Y_{u,v},u,v}$；未加权平均 $L = \frac{1}{|V|} \sum_{(u,v)\in V} \ell_{u,v}$。
- $L = \frac{-\ln(0.8)-\ln(0.6)}{2} \approx 0.367$；实现时直接将logits传给交叉熵函数。

背景类别与忽略像素

- 本例背景0、道路1、车辆2均为有效类别；ignore=255不提供本次监督。
- 模型仍只有C=3个得分通道；损失平均的分母只包含有效像素。

标签ID：0 / 1 / 2 / 255　　有效掩码：1参与，0忽略

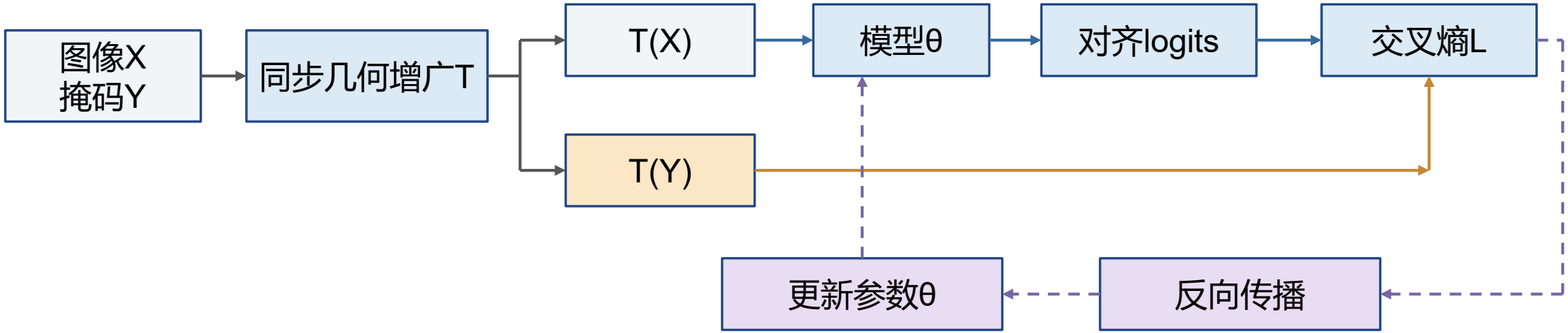


背景0：有效　道路1：有效　车辆2：有效　255：ignore

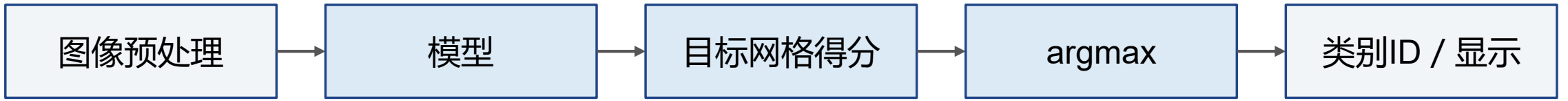
255排除损失分子与分母；评价遵循数据集协议，是否评估背景须另行声明。

语义分割网络怎样训练与推理?

- 图像与标签同步几何增广；**只有图像进入模型前向**，标签只进损失。
- 真实像素标签通过交叉熵提供监督，反向传播更新参数；预测质量还需用指标评价。



蓝色主路：模型前向。琥珀：标签只进损失。紫色虚线：梯度与参数更新。



推理不读取标签、不更新参数；交叉熵接收logits，训练时不预先softmax。

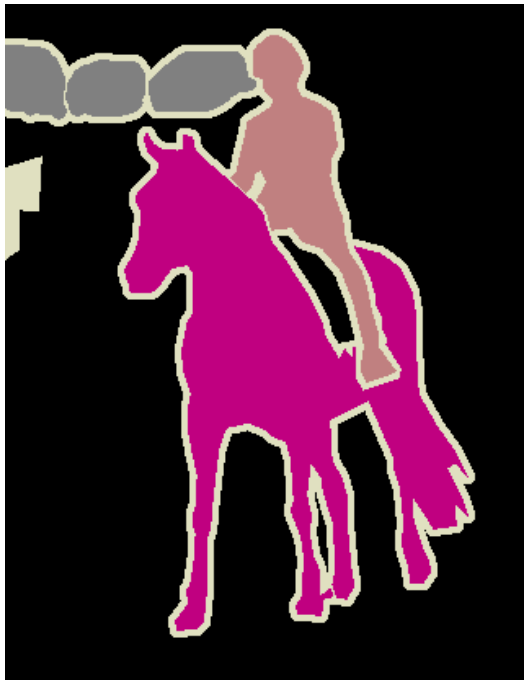
真实图像中的分割误差

- 对照同一图像的GT与FCN-8s预测，先看腿部细结构，再看区域和边界。
- 单个例图只能显示局部错误；接下来按类别统计误报、漏报，并计算IoU与mIoU。

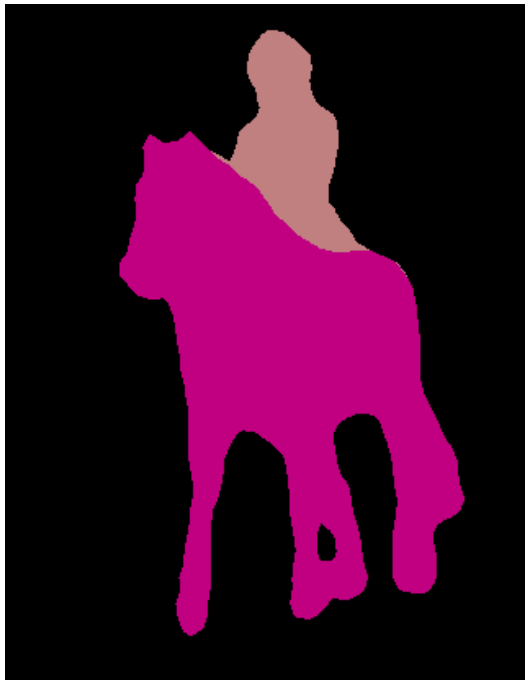
真实图像



真实标签 GT



FCN-8s预测



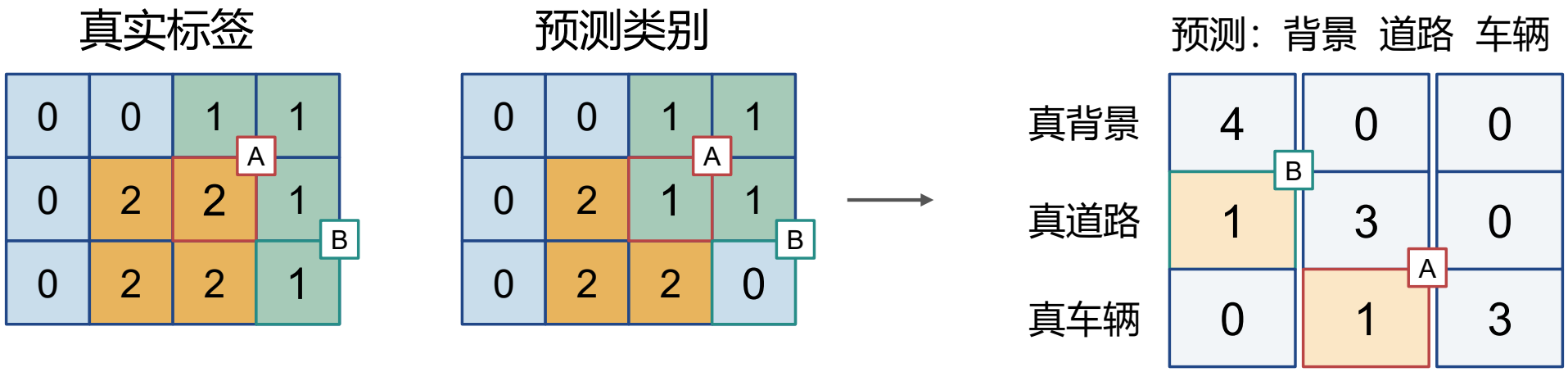
Long等，FCN，图6第一行｜原论文色板；非本课程训练，不与三类机制色混用。

6 像素统计与mIoU

真实例图显示误差；回到12像素主例，演示指标计算。

从像素对应到混淆矩阵

- 延续前面的12像素主例（教学构造）：行是真实类别，列是预测类别，逐位置累加。
- 两处错误增加“车辆→道路”和“道路→背景”；对角线合计10，列和为5、4、3。



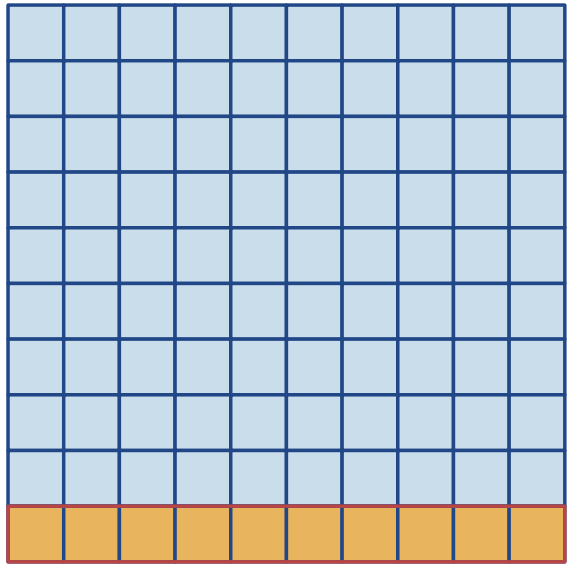
A：真车辆→预测道路；B：真道路→预测背景。对应各增加一次非对角计数。

格内数字分别是类别ID与计数；独立角标A/B只标识两处错误。

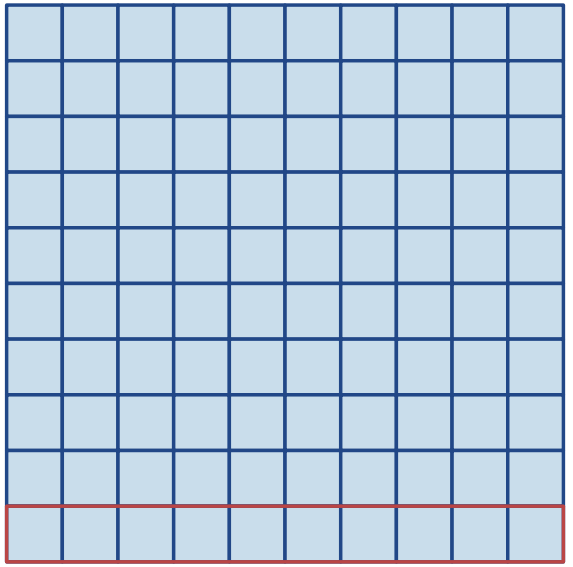
像素准确率可能掩盖少数类错误

- **像素准确率**是有效像素中预测正确的比例；12像素主例为 $10/12 \approx 83.3\%$。
- 另看100像素反例：90背景、10车辆；全预测背景，准确率90%，车辆全部漏检。

真实：90背景 + 10车辆



预测：全部背景



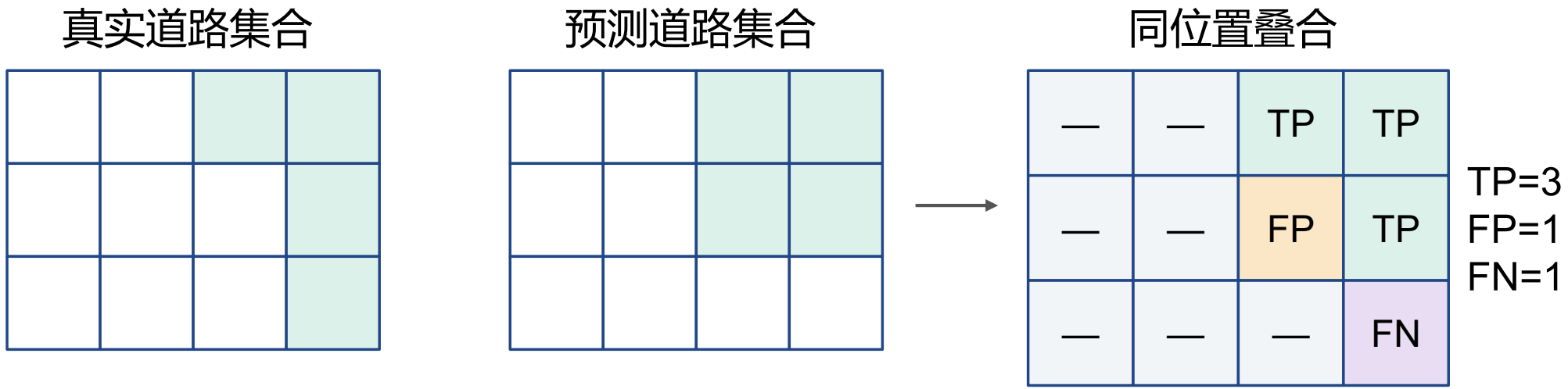
准确率：90/100=90%
车辆：TP=0，FN=10
少数类完全漏检



10×10独立教学构造；红框标出车辆所在行。应继续检查逐类误报与漏报。

逐类IoU同时计入误报与漏报

- 类别 c 的 $I_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c+\mathrm{FP}_c+\mathrm{FN}_c}$：交集占并集的比例。
- 返回12像素主例：TP正确、FP误报、FN漏报；比较同类像素集合，不匹配检测框。



交集：3个TP像素；并集：3 TP + 1 FP + 1 FN = 5个像素

道路 IoU=3/5=0.60　背景 IoU=4/5=0.80　车辆 IoU=3/4=0.75

“—”既不在真实道路也不在预测道路，不进入道路IoU的分母。

mIoU是评估类别的平均

- **mIoU**对声明的评估类别逐类取IoU，再作算术平均。
- 以 C_{eval} 表示评估类别数：$\text{mIoU} = \frac{1}{C_{\text{eval}}} \sum_c I_c$。
- 12像素主例：背景、道路、车辆三类。
  - $\text{mIoU} = (0.80 + 0.60 + 0.75)/3 \approx 71.7\%$。
- 如果只评估道路与车辆，结果是 $(0.60 + 0.75)/2 = 67.5\%$，平均范围已经改变。
- 像素准确率83.3%与mIoU 71.7%回答不同问题，不能互相替代。

100像素反例：车辆IoU为0，背景IoU为0.90，两类mIoU仅45%，准确率却为90%。

报告分割结果前明确评价约定

- 先在整个评价集累计混淆矩阵，再由累计TP、FP、FN计算各类IoU。
- 不把每幅图的mIoU直接等权平均来替代上述数据集统计。
- 声明评估类别、背景是否包含、忽略区域与预测尺寸还原方式。
- 某类真实和预测像素都为零时，IoU分母为零，须按评价协议处理，不能自动记为1。
- 跨论文比较先核对评估类别集合与协议，再看逐类结果和边界错误；一个平均数不能说明全部失误。

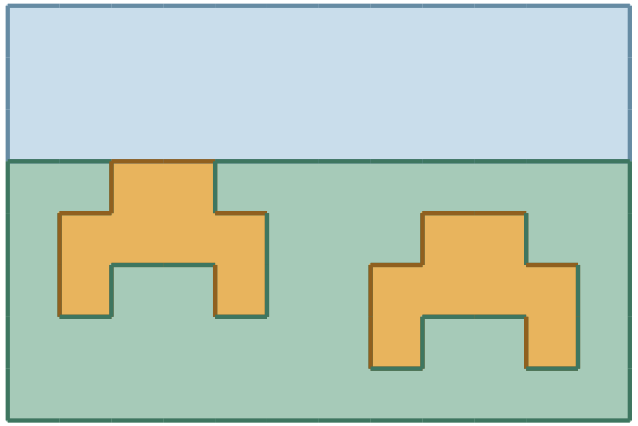
7 语义、实例与全景分割

语义分割的预测、学习与评价已闭合；进一步比较类别与实例身份。

同类区域是否需要区分实例

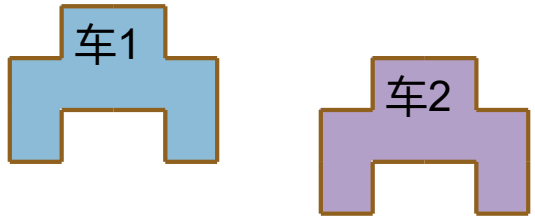
- 前面只要求逐像素类别；若还要区分同类目标，就需引入实例身份。
- 图中车辆颜色在语义图表示类别，在实例 / 全景图表示实例身份；实例评价不能只用语义mIoU。

语义分割



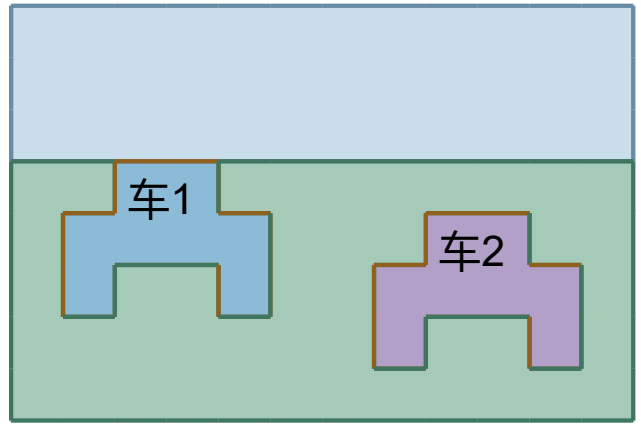
两车同色：同一类别

实例分割



不同车辆实例

全景分割



实例 + 完整分区

实例图白色区域不提供stuff分割；全景图为道路、背景及每辆车组织互不重叠的区域。

此教学三类：背景0含天空、道路1、车辆2；实例色≠类别色。一般stuff分类见下页。

全景分割同时组织类别与实例

- 可数目标（things）如人、车，需要区分各个实例。
- 不区分个体的区域类（stuff）如道路、天空，按语义类别描述；它们不是一个统一的“背景”类别。
- 全景分割将两者组织为统一场景划分，每个被评估像素属于一个区域。
- 它需要解决类别与实例结果的一致组织，不能只把两套有重叠的掩码直接叠放。
- 本讲建立任务区别；具体模型与全景评价指标属于扩展内容。

本讲总结

- 语义分割以像素类别掩码提供空间监督，显示颜色只是类别编码。
- FCN保留空间特征，以共享类别映射产生得分图并恢复输出尺寸。
- 插值与转置卷积扩大采样网格；转置表示线性映射转置，不保证恢复原信息。
- 跨层融合补充定位信息，FCN得分相加与U-Net特征拼接需要不同条件。
- 像素交叉熵联系得分与标签，空间变换、背景及忽略处理必须一致。
- 逐类IoU和mIoU揭示类别质量，必须声明累计方式与平均范围。

本讲以FCN和U-Net建立密集预测及融合基础；后续模型还研究上下文、尺度和预测表示。

选讲：转置矩阵不是逆矩阵

- 与前例对应，普通卷积可写为 $\boldsymbol{A} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$。
- 对5维输入向量 $\boldsymbol{q}$，卷积输出为 $\boldsymbol{Aq}$，维数变为2。
- 转置卷积计算 $\boldsymbol{A}^{\mathrm{T}}[1,2]^{\mathrm{T}} = [1,1,3,2,2]^{\mathrm{T}}$。
- 再作原卷积得到 $\boldsymbol{A}[1,1,3,2,2]^{\mathrm{T}} = [5,7]^{\mathrm{T}}$，并非 $[1,2]^{\mathrm{T}}$。
- $\boldsymbol{A}^{\mathrm{T}}$ 调换线性映射方向；它不保证恢复原输入，不能称为求逆。

	位置0	位置1	位置2	位置3	位置4
A第1行：覆盖0–2	1	1	1	0	0
A第2行：覆盖2–4	0	0	1	1	1

A的两行转为A^{T}的两列，分别乘输入1、2再相加；位置2同时收到两路贡献。