

Encoder

PIF:

intensities: $18 \times h \times w$, 最后一层是背景

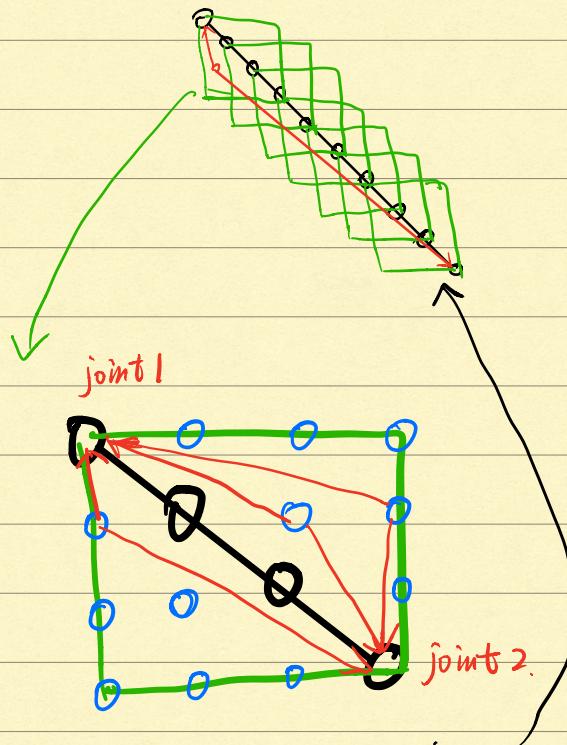
其中以关节点为中心，大小为 side-length 范围内为 1，其余为 0

fields-reg: $17 \times 2 \times h \times w$, 以关节点为中心，大小为 side-length 范围内所有点到中心的二维向量

fields-reg-l: $17 \times h \times w$, 所有二维向量的 L2 范数
用来计算 loss 时使用

fields-scale: $17 \times h \times w$, 用来表示人的大小，计算方式是人的
面积然后开方，对于同一个人不同关节 scale 相同

PAF:



在两个关节点之间
连续取多个点。

一张图象，绿色框范围内的
点才有向量表示。绿色
框外的点不进行考虑。

intensities: $20 \times h \times w$, 在绿色框中的 intensities 为 1,
表示这个点在关节上，其余地方为 0

field-reg1: $19 \times 2 \times h \times w$, 表示这个点到 joint1 的向量

field-reg2: $19 \times 2 \times h \times w$, 表示这个点到 joint2 的向量

注：这两个向量一正一反，都是以当前点为起点

field-reg-l: $19 \times h \times w$, 表示这个点到 joint1 与 joint2 之间
该径距离的长度

field-scale: $19 \times h \times w$, 表示人的大小。计算方式是人的
面积开方。与 Pif 中的计算相同。

Decoder

根据 Pif 与 Paf 连出图像来

(Pif)

$17 \times 4 \times h \times w$,

4 表示 confidence, x, y, scale

其中每个点处的 x, y 表示离这个点最近的关节点的坐标

(Paf)

$19 \times 2 \times 4 \times h \times w$,

2 表示 2 个向量, 4 表示 confidence, x, y, scale.

confidence 为这个点在关节连线上的置信度,

x, y 表示这个点连接的第 1/2 个关节点的坐标

① 连接前首先要得到 high resolution confidence map, 做法是将

Pif 放大为原来的 stride 倍，其中填充 confidence 时使用

Gauss 核，累加后除以 16，填充 scale 时使用了常数核，求了个均值的

③ 得到 high resolution confidence map 之后要产生 seed，因为接下来的连接
操作便是以 seed 为起点。

todo 这时会遇到 seed 连接到一定程度后碰到另一个 seed 的情况。

(seed 可能一口气连下来一个别的 seed 都碰不到也可能碰到别的 seed)

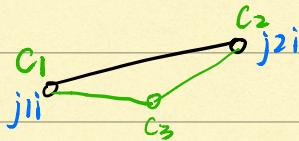
seed 的选择首先要 $> seed_threshold (0.2)$ ，然后按照 confidence 值

从大到小排序。每逢一个 seed 就把它周围 $\frac{w}{2}$ 方形区域内的点设置为不可选，目标是为了防止一个人的一个关节点选多次。

seed 的格式为 (confidence, 关节点序号, x, y)
 hr 非 hr 原因上

② 对得到的 $19 \times 2 \times 4 \times h \times w$ 的 PAF 进行打分

首先过滤出 $confidence > 0.1$ 的连接中间点



$$scores_b = C_3 * (pf + (1-pf) * C_1)$$

其中 pf 为一个常数，为 0.01 或 0.9

用 $scores_b > 0.1$ 来过滤连接，将满足条件
 这个条件与第 1 个关节点的置信度有关

满足 条件 的点的连接按照 $\left\{ \begin{array}{l} scores_b [mask_b] \\ foursd[1, 1:4] [:, mask_b] \\ foursd[0, 1:4] [:, mask_b] \end{array} \right. \quad (3, num2)$
 得分，对应的第 2 个、第 1 个关节点

整体记为 $score_backward$

同理， $scores_f = C_3 * (pf + (1-pf) * C_2)$

$$score_forward = \left\{ \begin{array}{l} scores_f [mask_f] \\ foursd[0, 1:4] [:, mask_f] \\ foursd[1, 1:4] [:, mask_f] \end{array} \right. \quad (3, num3)$$

满足 $scores_f > 0.1$ 的点的得分，对应的第 1 个、第 2 个关节点。

最终过滤出来了 2 个矩阵，分别满足不同的打分函数

$score_forward$ 维数为 $(19, 7, num3)$

7 表示「这个点在这个关节上的得分，joint1 的 $x, y, log_b, joint2 x, y, log_b$ 」

④ 下面就开始连接的过程了。循环遍历所有 seed，从 seed 开始对每一个
 人进行 grow。对每一个人（因为 data 只能存 17 个关节的数据）循环

$grow \leq 19$ 次。grow 的过程包括前向探索与反向验证 两阶段。

前向探索时找出所有目前应该连接但还未连接的线。

这些线可能是正向的 (seed 作为起点), 也可能是反向的 (seed 作为终点), 每次选一条 confidence 最高 的线来考虑。

注意: seed 是由 Pif 得出来的, 接下来得到的连接线由 Paf 产生
因为 Paf 中存有始末关节点的坐标, 这个坐标从道理上讲应与 Pif 中
的接近, 这是用来判断是否是同一个人的主要依据

第一次得到的 x, y, v 由 Pif 得到, ↓ 这里的 confidence 是人体 17 个
关节中所有的 confidence, 每次根据这个找到应连的线。

即从 paf 中找从关节 A \rightarrow B 的所有连接线, 这会找到
非常多条, 因为有很多人, 而且每个人不止一条。这里同时选取了
从关节 B \rightarrow A 的连接, 这是为了后面作反向验证。

接下来根据当前关节 (seed) 的位置信息对上面得到 Paf 连接进行
过滤, 找出那些可能的连接。过滤方法就是以 seed 的 x, y
为中心, $2 * score$ 的范围内。

过滤完后连接可能还剩下多条, 我们首先为这些 paf 评分。

公式: $s(a, \vec{x}) = a_c \exp\left(-\frac{\|\vec{x} - \vec{a}\|}{b_1}\right) f_2(a_{x_2}, a_{y_2})$

得分 \downarrow paf \downarrow seed 点 \downarrow confidence \downarrow 又与 paf 中 x_2, y_2 的模

这个是关于 x_2, y_2 的函数, 两种策略中有所不 同

a 是 paf, 有 7 维, 分别是 (confidence, $x_1, y_1, b_1, x_2, y_2, b_2$)

这里有两种策略, 一种是直接选择最大的 score, 第二种是利用
循环迭代法来找这些点的中位中心 (到所有点距离和最近的点)
其中直接选择最大的 score 比较迅速

(大体测试了几张图片发现 max 比 median 要准)

⑤ 反向验证，倒过来，之前是看 joint1 如何预测出 joint2，
现在变成给了 joint2 看能否反向推出 joint1。