

Histograms of Oriented Gradient Note

FanmingL

2018 年 12 月 1 日

1 梯度方向直方图

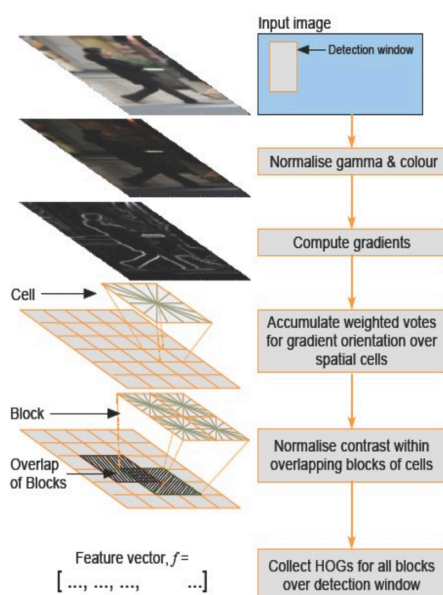


图 1: HOG 特征提取流程

HOG 算法的整体流程如图1所示，下面分别介绍提取 HOG 特征的五个步骤。

1.1 伽马归一化

首先将图像中检测窗口的部分 I 提取出来，并做归一化即 $I^{new} = I/255$ ，然后对图像的每个通道的每个元素取 gamma 次幂，常用的取 $\frac{1}{2}$ 次幂，即完成了伽马归一化。实验证明，这一步对结果的影响不大，因为最后的时候也有一个归一化的步骤。

1.2 计算梯度

由于我们都知道，梯度对噪声敏感，但是这里我们不对图像进行平滑，因为实验表明，平滑之后，往往会使结果更差。

然后我们用 $[-1, 0, 1]$ 算子和 $[-1, 0, 1]^T$ 算子对图像进行卷积计算，得到两幅新的图像，分别表示图像水平方向的梯度 \mathbf{G}_x 和垂直方向的梯度 \mathbf{G}_y 。

对于图像为了方便之后的计算，令梯度的模长为 $\sqrt{\mathbf{G}_x^2 + \mathbf{G}_y^2}$ ，相角为 $\text{atan}(\frac{\mathbf{G}_y}{\mathbf{G}_x})$ ，并将相角映射至 $[0, 180^\circ]$ 的范围内。

对于多通道的图像，我们只保留梯度模值最大的分量以及其对应的相角。

1.3 将图像分割为小的 Cell 单元格

Cell 单元格是 HOG 特征最小的结构单位，每个 Cell 单元格由 $8*8$ 个像素构成，Block 块、检测窗口的滑动步长就是一个 Cell 单元格的长度。

1.4 构建梯度方向直方图

对于每个 Cell，我们用有 9 个 bin 的直方图来统计这 8×8 个像素的梯度信息，即把 $[0, 180^\circ]$ 平均分为了 9 份。

对于每个像素，我们都根据其梯度模值、梯度相角、在检测窗口中的坐标 x 、 y 对其周围的四个 Cell，每个 Cell 中的两个 bin 进行投票，即一共要给 8 个 bin 投票。首先对于每个 Cell，设其中心的在检测窗口中的坐标为 (x_c, y_c) ，Cell 的大小为 (b_x, b_y) ，像素的梯度模值为 p ，相角为 θ 。那么 p 会按照一定比例投给这 8 个 bin，首先我们假设，分给第 i 个 Cell 的权重为 p_i ，那么 p_i 会按照线性插值的方式，投票给离 θ 最近的两个 bin。

即，假设 $[0, 180^\circ]$ 平均分成了九份，那么第一份就是 $[0, 20]$ ，第二份为 $[20, 40]$ ，以此类推，那么第一份的中心为 10° ，第二份的中心为 30° ，为假设 $\theta = 15^\circ$ ，那么投给 $[0, 20]$ 的权重为

$$p_{i1} = \frac{15 - 10}{20} p_i = 0.25 p_i \quad (1)$$

投给 $[20, 40]$ 的权重为

$$p_{i2} = \frac{30 - 15}{20} p_i = 0.75 p_i \quad (2)$$

分配过程如图1.4， p_i 也是由 p 以位置为依据分配的，即

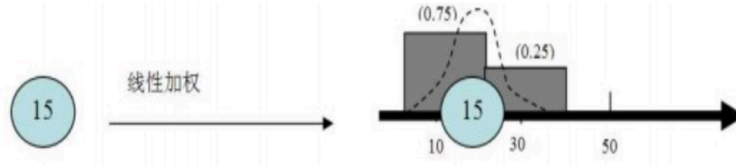


图 2: 相角之间的线性加权

$$p_i = (1 - \frac{|x - x_{ci}|}{b_x})(1 - \frac{|y - y_{ci}|}{b_y})p \quad (3)$$

(x_{ci}, y_{ci}) 是像素周围第 i 个 Cell 的中心的坐标，Cell 之间的分配过程如图1.4。整个分配的

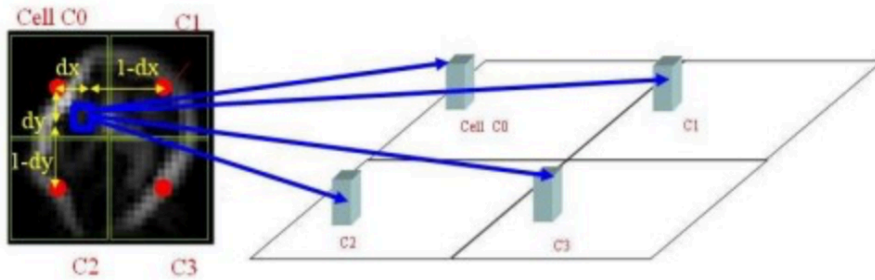


图 3: Cell 之间的线性加权

过程，我们称之为三线插值，如图1.4，左边蓝色的圈圈指的是待投票的像素，用三线插值，将其权重分配给右图蓝色的 8 个 bin 中。于是基于三线插值，对于每个 Cell，我们得到了一个梯度方向直方图。

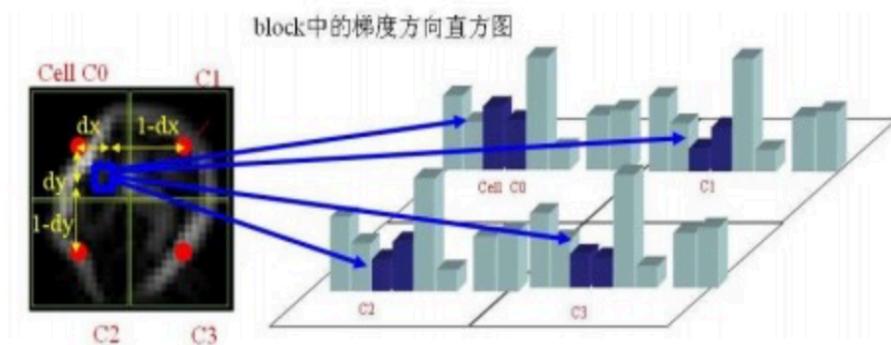


图 4: 三线性插值

1.5 block 块内归一化直方图

其实在1中可以看出，一个 block 块由若干个 Cell 构成，这里是四个，我们将块内的所有直方图拼成一个向量，并对其归一化，归一化方法有四种，令 e 为一很小的常数，以避免除数为零。

L2-norm,

$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \quad (4)$$

L1-norm,

$$f = \frac{v}{\|v\|_1 + e} \quad (5)$$

L1-sqrt,

$$f = \sqrt{\frac{v}{\|v\|_1 + e}} \quad (6)$$

L2-Hys: 先对向量进行 L2-norm，然后大于 0.2 分量的元素就直接等于 0.2，再重新对向量进行 L2-norm。

其中除了 L1-norm 表现较差，另外三种归一化方法效果都差不多，但是相比于没有归一化的数据而言，这里的数据要好很多。

对于 block 块也是有两种几何形状包括矩形 block 和环形 block，二者效果相同。

归一化之后，当前 block 便可以得到一个向量，作为 HOG 的描述向量。然后将 block 移动 Cell 的宽度个像素的距离，计算新的 HOG 描述符。不管是从图1还是从这里的描述，我们都很容易想到，相邻的两个 block 会有一半重复的像素，这样的交叠会模糊块与块之间的边界，是有利的。

1.6 得到检测窗口的 HOG 描述符

将 block 以 Cell 宽的滑动距离在检测窗口上滑动，最后将所有 block 的 HOG 描述向量拼起来形成一个更大的向量，这就是这个检测窗口的 HOG 描述符。

举个例子，若我们的检测窗口大小为 128×64 像素，block 块大小为 16×16 像素，Cell 大小为 8×8 像素，每个 Cell 的梯度直方图有 9 个 bin，那么一个检测窗口的 block 的数量为 $((128 - 16)/8 + 1) * ((64 - 16)/8 + 1) = 105$ ，而一个 block 有 4 个 Cell，一个 Cell 的

描述向量的长度为 9，所以一个 block 的描述向量长度为 $4 * 9 = 36$ ，所以整个检测窗口的 HOG 描述符的向量长度为 $105 * 36 = 3780$ 维。

1.7 总结

HOG 的用梯度方向直方图作为一个 Cell 的描述向量，这与 SIFT 算子是类似的，但是 HOG 有个很重要的操作——块内对直方图归一化，梯度方向直方图能够很好的描述局部的特征，而归一化的操作增强了其鲁棒性，使其被光照背景的影响更为微弱。

论文中用 HOG 算子和线性 SVM 算法实现了一个准确度很高的行人检测的模型，由于 SVM 的特性，让它很适合处理高维的数据，所以二者的结合是非常完美的，在 2005 年能够做到这样一个准确率，非常强。