

1. 实验目的

利用机器学习方法（分类）实现静态场景下的测试车辆检测。

2. 实验步骤^[1]

（1）构建词典

用 Forstner 算子从训练集的图片中提取特征点，以特征点为中心取 13×13 个像素作为一个 patch。用 k-means 聚类算法自底而上地将相似的 patch 聚成一类，每一个类看成是词典中的“单词”，本次实验中 K 设置为 500。

（2）图像特征表示

从图片中提取的 patch 集合为 P ，比较 P_i 和每一个“单词”的相似度，相似度度量为 P_i 与“单词”中 patch 的相似度最大前 λ 个的平均值。如果 P_i 和某个单词的相似度大于阈值，则表示该“单词”在图像中出现，特征值记为 1 否则为 0。

此外，我们还需要用特征刻画 patch 之间的空间关系。将图片中 patch 间的距离分成 5 个 bin，角度为 4 个 bin。这样就有 20 种组合关系。如组合关系在 patch 间出现，则特征值记为 1 否则为 0。

组合这两类特征，我们就可以得到该图片在词典映射下的特征向量。

（3）分类器学习

将训练集的特征和标签放入 SNoW 模型中训练，该模型适用于稀疏特征表达的样本。

（4）找到最佳的物体检测位置

因为训练样本的尺度与测试集中车辆的尺度相同（均为 34×94 ），因此设滑动窗口大小为 34×94 ，在测试集图片上移动，每个窗口提取特征交给分类器判断。

对于包含车辆的图片，在识别目标附近可能存在多个位置被判断为 positive，因此还需要筛选出最佳检测位置。这里采用近邻抑制法（Neighborhood suppression）。

我们将分类器判断为 positive 并且置信度大于阈值的区域称为激活区域，初始状态都是非抑制，每步算法都找出置信度最高的非抑制区域，如果该区域附近有一些激活区域，则其他区域的状态变为抑制，该区域为最佳检测位置。在本次实验中，我们设定 71×81 像素为邻域。

3. 实验结果分析

标注后的测试集见文件夹 **labeledTest**。

（注：代码直接调用了 MATLAB 中自带的 `bagOfFeature()` 函数，该函数是基于参考文献[2]实现的，与参考文献[1]相比在特征点提取（SURF 算法），图像的特征表达（在词典上出现的频率）上有所不同，分类器使用了 SVM 模型（因为特征是连续的），但大体思路都是类似的，都是基于 BoW 模型。）

（1）性能评价

Activation Threshold	TP	FP	Recall TP/200	Precision TP/(TP+FP)	F1 $2 \cdot R \cdot P / (P + R)$
0.6	177	83	88.50%	68.07%	76.05%

（2）计算速度

测试集共有 170 张图片，算法执行时间为 331.432895 秒，平均 1.95 秒/张。

（3）误差分析

误差可能来自于以下几个方面：

1. 分类器的误差。从结果中可以看出，分类器也将很多根本不包含车辆的窗口判断为 positive，给后面 suppression 部分带来干扰。
2. 对于测试集的每张图片，滑动窗口都以步长 5 移动。然而测试集的图片大小并不唯一，可能会错过某些最佳识别位置，而分类器对像素的变化（尤其是涉及边缘部分）又十分敏感，导致出现误判或者 score 不高的情况。
3. 为了加速计算，程序中规定只有分类器判断 `label=2` 并且 `abs(loss(2)-loss(1))>threshold` 的 bounding box 才能进入后面的 suppression 部分，threshold 对于所有测试数据是一个常数，为了保证在所有测试数据上都成立，因此也不能设的过高。因此会导致 FN 值比较大。

4. 参考文献

- [1] Shivani Agarwal, Aatif Awan, and Dan Roth, "Learning to Detect Objects in Images via a Sparse, Part-Based Representation", PAMI, 2004
- [2] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, Cedric Bray, "Visual Categorization with Bag of Keypoints", ECCV, 2004.