

我上一周了解了几种常用的行人检测的算法并选择 HOG 来实现行人检测功能，因此我这周继续搜索 HOG 的资料并了解 HOG 算法实现细节。经过查看网上的资料以及一些例子代码，我弄清了 HOG 所用到的 windows、block、blockstride、cell 之间的关系。

HOG 算法的主要过程是在计算图像的梯度向量（Gradient Vector），而梯度向量主要用于边缘检测和特征抽取，并且经过规格化的梯度向量还有光照不变性的性质。

此外，我还使用 opencv 自带的 HOGDescriptor 对几种不同大小的图片进行行人检测并计算平均处理时间，最后得出的结果如下表。从结果可以得出，HOG 的处理速度较慢，因此不适用于实时检测的应用，即使做离线检测，测试数据的分辨率也不能太高，要不然需要的处理时间太长。

size	Average time(ms)
1920*1080	5991.572
1200*800	2740.144
800*600	1349.194
600*400	641.019
320*240	177.939

（所用电脑配置：CPU: i3@3.40GHz 3.40GHz Memory: 8GB）

(PS: 因为有其他事情要忙，所以我这周花在学习上的时间比较少。)