### 

****

**数字图像处理实验报告**

**实验4 目标检测**



**学 院 智能与计算学部**

**专 业 计算机科学与技术**

**学 号 3022206045**

**姓 名 陆子毅**

# 1. 实验内容

（1）使用HOG目标检测算法对自选图像进行目标检测测试，图像HOG特征可视化

（2）在深度学习目标检测算法中，选择一种单阶段（One-Stage）或两阶段（Two-Stage）的代表性算法，将其与传统目标检测算法进行对比

# 2. 实验原理

**2.1 HOG目标检测算法与图像HOG特征可视化**

HOG（Histogram of Oriented Gradients）特征是一种常用的计算机视觉特征描述符，广泛应用于物体检测任务中，尤其是行人检测。HOG特征的基本思想是通过图像的局部梯度信息来描述物体的形状和边缘结构。以下是HOG算法的基本步骤：

1. 图像预处理：对输入图像进行灰度化处理，简化信息，减小计算复杂度。
2. 梯度计算：使用Sobel算子计算每个像素点的梯度值，得到图像的梯度方向和强度。
3. 局部块划分：将图像分割成8x8的多个小块，并在每个小块内统计梯度方向的分布。
4. 梯度方向直方图：每个块的梯度方向通过构建一个方向直方图来表示，将梯度方向分为9个方向，每个方向的直方图存储该方向的梯度值。
5. 特征标准化：为了提高鲁棒性和减少光照变化的影响，需要对每个块的特征进行归一化处理。
6. 特征融合：将所有局部块的HOG特征融合成一个向量，形成图像的HOG特征描述。
7. 目标检测：利用训练好的HOG特征分类器（如SVM）进行目标检测，逐步滑动窗口检测图像中的物体。

HOG特征的可视化通常是通过反向映射图像块中的梯度方向和大小来展示。可以将每个小块中的梯度方向作为线条进行绘制，线条的长度与梯度大小成正比，从而直观地展示出图像的局部形状和边缘信息。

**2.2 深度学习目标检测算法对比：单阶段与两阶段算法**

目标检测任务可以使用**传统算法**（如HOG+SVM）和**深度学习算法**。深度学习方法可以分为单阶段（One-Stage）和两阶段（Two-Stage）算法。

**单阶段算法**（One-Stage）：例如YOLO（You Only Look Once）和SSD（Single Shot MultiBox Detector）。单阶段算法通过将目标检测问题作为一个回归问题来解决，在一次前向传递中预测出所有的目标框和类别。其优点是速度较快，适合实时检测，但可能会在精度上稍逊色。

* **YOLO**：YOLO算法通过一个卷积神经网络（CNN）直接预测目标框的位置和类别，网络在每个位置生成一个bounding box并给出概率，快速且高效，适合实时检测。
* **SSD**：与YOLO类似，SSD将目标检测任务分解为多个不同尺度的预测问题，使用不同大小的卷积核在不同尺度的特征图上预测目标边界框。

**两阶段算法**（Two-Stage）：例如R-CNN（Region-based Convolutional Neural Network）、Fast R-CNN、Faster R-CNN。两阶段算法首先通过候选区域生成（Region Proposal）提取可能包含目标的区域，然后对这些区域进行更精细的分类和回归调整。其优点是精度高，但计算开销较大，不适合实时检测。

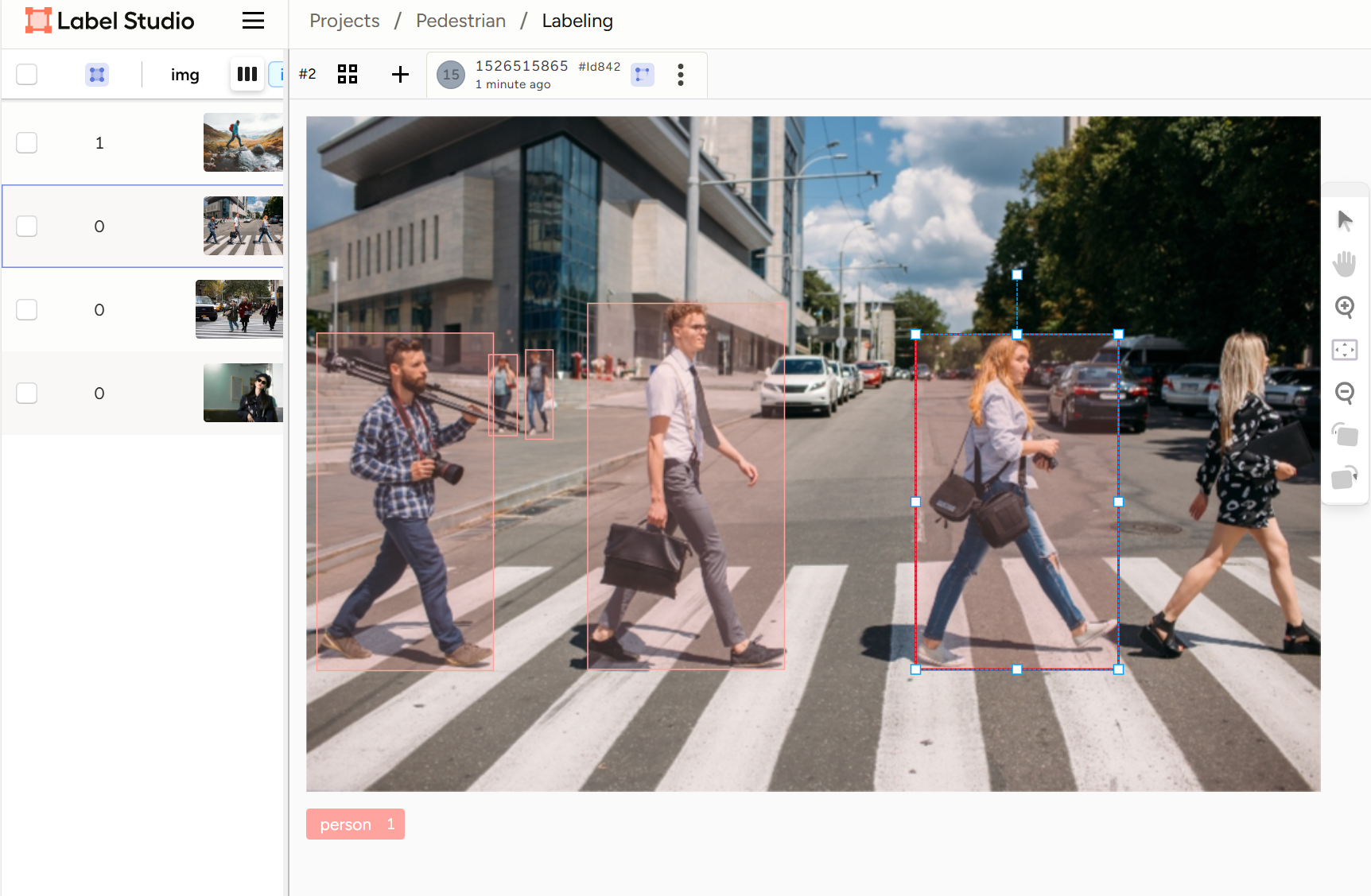
* **Fast R-CNN**：**Fast R-CNN** 引入了**共享卷积特征**和**RoI Pooling**（Region of Interest Pooling）技术来加速目标检测的过程。在Fast R-CNN中，目标检测被分为两个主要部分：**候选区域生成**和**分类与回归**。通过这些技术，Fast R-CNN显著提高了检测的速度和精度。

# 3.实验过程

**1. 数据准备**

**收集数据**：数据集包括正样本和负样本（不含目标物体的图像）。每个图像都会标注出目标的位置。

**图像标注**：为正样本提供目标的位置（如边界框坐标），使用Lable Studio进行图片数据的标注



**2. 特征提取：HOG特征计算**

**图像预处理**：对图像进行灰度化处理，去除颜色信息，减少计算量。然后进行归一化处理，确保图像的光照不对目标检测结果造成太大影响。

**梯度计算**：对于每一幅图像，使用Sobel算子计算每个像素点的梯度（方向和强度）。

**局部块划分**：将图像划分为多个8x8像素块。每个块内计算梯度的方向直方图，归一至9个方向。

**特征标准化**：为了减少光照变化和对比度变化的影响，对每个块的梯度直方图使用L2-Hys进行归一化处理，以提高特征的鲁棒性。

**全图特征融合**：将所有小块的HOG特征向量拼接成一个大的特征向量，作为该图像的HOG特征表示。

**3. 训练支持向量机（SVM）分类器**

**构建训练数据**：利用HOG特征从正样本和负样本中提取特征。每个图像的HOG特征作为输入，标签为1或-1。

**训练SVM模型**：

1. 使用正样本和负样本的HOG特征训练一个支持向量机分类器。SVM是一种二分类算法，它通过构造一个超平面来最大化正负样本之间的间隔。具体来说，SVM的目标是找到一个最优的超平面，将正样本和负样本分隔开来，并且尽量避免误分类。
2. 选择高斯核来帮助处理非线性可分的问题。
3. 通过交叉验证等方式调节SVM的超参数，以避免过拟合。

**SVM训练结果**：训练完成后，SVM模型将会学到一个超平面，该超平面能够将行人和背景图像区分开来。

**4. 目标检测**

**滑动窗口**：训练完成的SVM分类器将应用于图像中的每一个位置。使用滑动窗口方法，在图像中以不同的尺度和位置遍历每个区域。每个窗口内的图像片段将提取HOG特征，并输入SVM分类器进行分类。

**多尺度检测**：为了处理不同尺寸的目标，通常会在不同的尺度上应用滑动窗口。

**候选框生成**：对于每个滑动窗口，SVM会返回一个分类结果。如果窗口中的图像被分类为行人，则该位置和尺寸的边界框被认为是一个候选框。

**5. 非极大值抑制（NMS）**

**去除冗余框**：在目标检测过程中，滑动窗口可能会产生多个重叠的检测框。为了去除冗余的检测结果，通常会应用非极大值抑制算法。NMS通过计算候选框之间的IOU，并保留置信度最高的框，删除与其重叠度较高的框。

**优化结果**：经过NMS处理后，最终输出的框将是目标检测的最终结果，生成json检测结果。

**6. 使用Fast R-CNN进行目标检测，生成json检测结果**

**生成目标检测结果**：使用COCO预训练模型对图片进行目标检测，生成json检测结果方便后续计算mAP。

# 4.实验结果

**4.1 HOG特征图像**

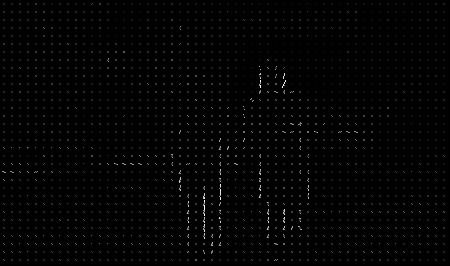
****

图1 HOG特征图像

**4.2 HOG+SVM行人检测结果**

****

图2 HOG+SVM行人检测结果

从实验结果来看，HOG+SVM传统检测方法将路灯和树也识别成了人，可信度很低，速度较慢具有较大误差，在复杂背景环境下，HOG+SVM传统目标检测算法的可靠程度并不高。能够正确识别行人，但是会引入错误的识别目标

**4.3 Fast R-CNN目标检测结果**



图3 Fast R-CNN目标检测结果

对于行人检测，Fast R-CNN准确将行人检测出来，给出的置信度的可靠性也比较高，在具有复杂背景的图片上获得了较好的效果。

**4.4 HOG+SVM和Fast R-CNN的比较**

通过计算目标检测结果的mAP来对检测效果进行比较。直接调用cocoeval

来进行检测

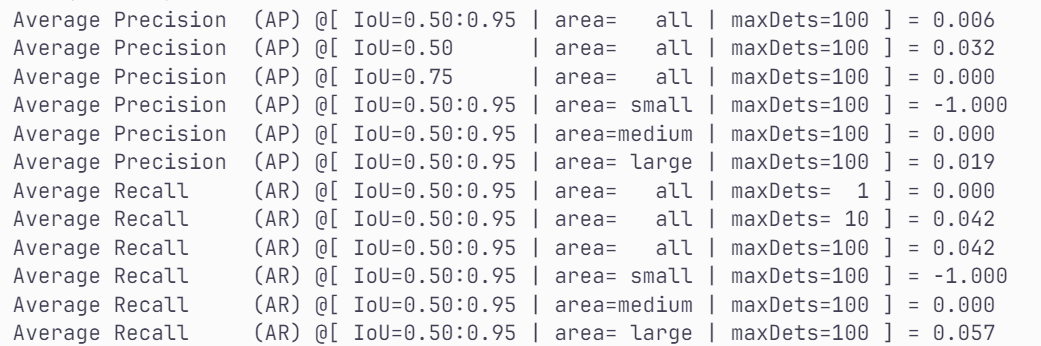


图4 HOG的mAP结果

**平均精度（AP）：**

1. 在所有IoU阈值和所有目标大小下，平均精度非常低，仅为0.006。
2. 在IoU阈值为0.50时，平均精度稍高，但仍然很低，仅为0.032。
3. 在IoU阈值为0.75时，平均精度为0，说明模型在高IoU阈值下几乎没有正确检测。
4. 没有小目标检测
5. 对于中等大小的目标，平均精度为0，说明模型在中等大小目标上没有正确检测。
6. 对于大目标，平均精度稍高，但仍然很低，仅为0.019。

**平均召回率（AR）：**

1. 在所有IoU阈值和所有目标大小下，最大检测数量为1时，平均召回率为0。
2. 最大检测数量为10时，平均召回率为0.042，稍有提升。
3. 最大检测数量为100时，平均召回率仍为0.042。
4. 对于小目标，平均召回率为-1，表示没有小目标的检测结果。
5. 对于中等大小的目标，平均召回率为0，说明模型在中等大小目标上没有正确检测。
6. 对于大目标，平均召回率稍高，但仍然很低，仅为0.057。

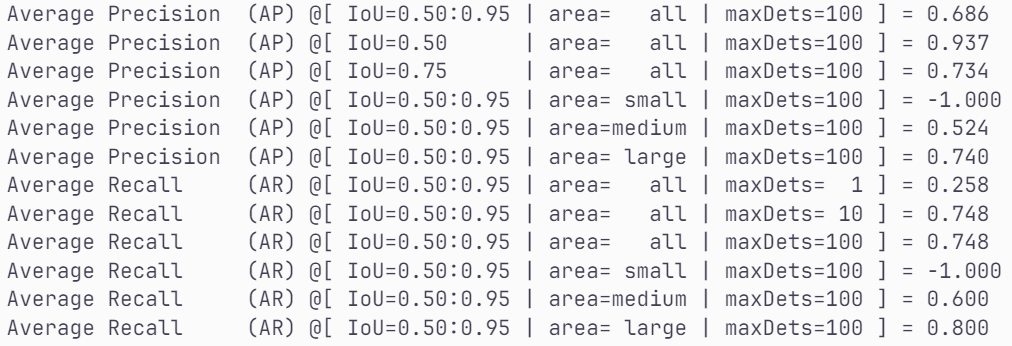


图5 Fast R-CNN的mAP结果

**平均精度（AP）**：

1. 在所有IoU阈值和所有目标大小下，平均精度为0.686，表现良好。
2. 在IoU阈值为0.50时，平均精度为0.937，表现非常好。
3. 在IoU阈值为0.75时，平均精度为0.734，表现良好。
4. 对于小目标，平均精度为-1，表示没有小目标的检测结果。
5. 对于中等大小的目标，平均精度为0.524，表现良好。
6. 对于大目标，平均精度为0.740，表现非常好。

**平均召回率（AR）**：

1. 在所有IoU阈值和所有目标大小下，最大检测数量为1时，平均召回率为0.258，表现一般。
2. 最大检测数量为10时，平均召回率为0.748，表现良好。
3. 最大检测数量为100时，平均召回率为0.748，表现良好。
4. 对于小目标，平均召回率为-1，表示没有小目标的检测结果。
5. 对于中等大小的目标，平均召回率为0.600，表现良好。
6. 对于大目标，平均召回率为0.800，表现非常好。

结果说明HOG+SVM的传统检测方式的泛用性较低，准确性受图像质量影响较大，对于复杂背景的图像，传统目标检测识别正确率较低。

相较于传统目标检测方式，Fast R-CNN的速度更快准确率更高。

Fast R-CNN对整个输入图像进行一次卷积操作，生成全图的特征图。所有候选区域（RoI）都共享同一个特征图，而不需要为每个候选区域重新计算卷积特征。通过这种方式，大幅度减少了重复计算的开销，从而提高了速度。

Fast R-CNN引入了**RoI Pooling**（Region of Interest Pooling）层，它能够将不同大小的RoI映射到统一大小的特征图上。RoI Pooling通过对RoI区域进行最大池化，将每个区域统一为固定大小的输出，避免了对每个区域进行重新缩放的复杂过程，同时提高了处理效率。

Fast R-CNN通过端到端训练机制，所有网络层（包括卷积层、RoI Pooling层、全连接层等）都可以通过反向传播共同训练。这样，Fast R-CNN能够同时优化特征提取、RoI Pooling和分类/回归任务，避免了多个独立训练过程，提高了训练效率，也让模型可以更加高效地学习到图像的高层次特征。

虽然Fast R-CNN仍然依赖外部的区域提议方法（如选择性搜索）生成候选区域，但通过共享卷积特征和RoI Pooling，减少了对候选区域特征的重复计算，并且通过优化分类和回归损失，能够提高检测精度，减少冗余框。

作为一种基于深度学习的目标检测方法，Fast R-CNN能够自动从数据中学习特征，并且通过卷积神经网络提取丰富的高层次特征。这些特征比传统的手工特征（如HOG）更具表达能力，能够更好地处理复杂背景、光照变化、姿态变化等问题，从而提高了检测精度。

附（所有实验图片）：

