**《计算机视觉》实验报告**

**姓名：刘远航 学号：22121883**

实验9

1. **任务1**

1、采用基于背景建模的方法实现运动目标检测，建模方法自选

算法步骤：

（1）读取视频（逐帧读取）

（2）累加权重重构背景（中值/均值建模），可选其他建模方法

（3）计算象素差，即每一帧图像与背景模板的差值

（4）图像去噪：中值滤波、均值滤波、形态学变换（腐蚀、膨胀）

（5）画出候选框，可用cv2.findContours()函数

（6）通过非极大值抑制筛选去除多余候选框

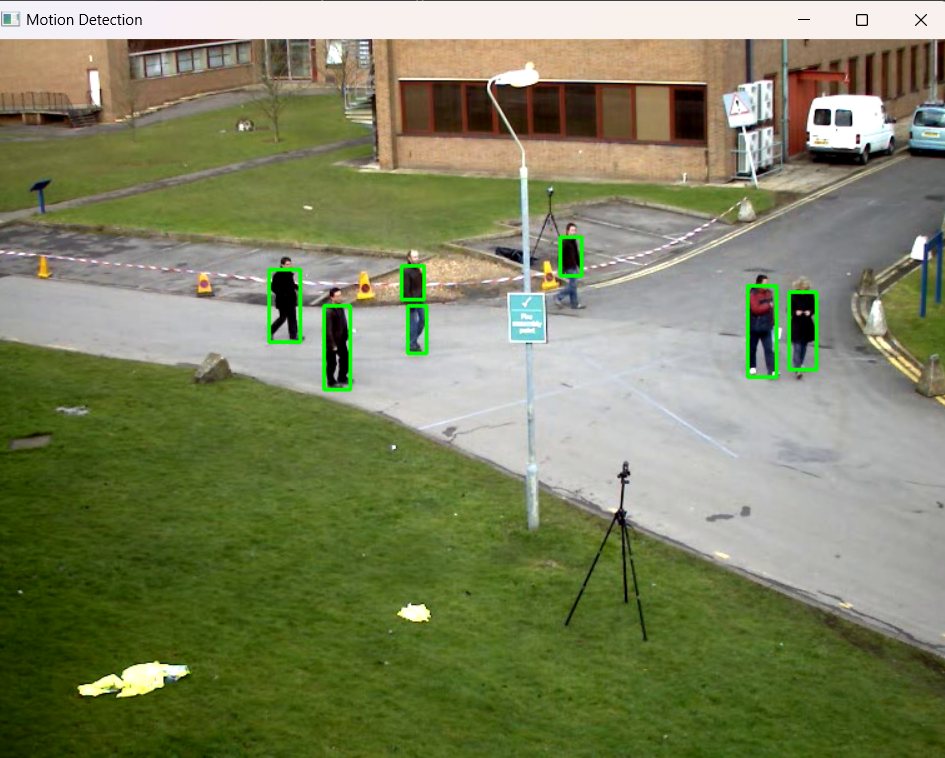
（7）输出检测结果

* 1. **核心代码：**

def non\_max\_suppression(boxes, overlap\_threshold=0.3):  
 if len(boxes) == 0:  
 return []  
 pick = []  
 x1 = boxes[:, 0]  
 y1 = boxes[:, 1]  
 x2 = boxes[:, 2]  
 y2 = boxes[:, 3]  
 # 计算边界框的面积  
 area = (x2 - x1 + 1) \* (y2 - y1 + 1)  
 # 按照边界框的底部坐标排序  
 idxs = np.argsort(y2)  
 # 迭代处理边界框，去除重叠部分  
 while len(idxs) > 0:  
 last = len(idxs) - 1  
 i = idxs[last]  
 pick.append(i)  
 xx1 = np.maximum(x1[i], x1[idxs[:last]])  
 yy1 = np.maximum(y1[i], y1[idxs[:last]])  
 xx2 = np.minimum(x2[i], x2[idxs[:last]])  
 yy2 = np.minimum(y2[i], y2[idxs[:last]])  
 w = np.maximum(0, xx2 - xx1 + 1)  
 h = np.maximum(0, yy2 - yy1 + 1)  
 # 计算重叠的面积  
 overlap = (w \* h) / area[idxs[:last]]  
 # 删除重叠度高于阈值的边界框  
 idxs = np.delete(idxs, np.concatenate(([last], np.where(overlap > overlap\_threshold)[0])))  
 return boxes[pick]

while True:  
 # 读取视频帧  
 ret, frame = cap.read()  
 if not ret:  
 break  
 # 更新背景模型  
 alpha = 1.0 / num\_frames  
 background\_model = (1 - alpha) \* background\_model + alpha \* frame.astype(np.float32)  
 # 计算当前帧与背景模型的差异  
 frame\_diff = cv2.absdiff(frame, background\_model.astype(np.uint8))  
 # 将差异图像转换为灰度图  
 frame\_diff\_gray = cv2.cvtColor(frame\_diff, cv2.COLOR\_BGR2GRAY)  
 # 对灰度图进行阈值处理，得到前景物体  
 \_, thresholded = cv2.threshold(frame\_diff\_gray, 0, 255, cv2.THRESH\_BINARY + cv2.THRESH\_OTSU)  
 # 对二值图像进行中值滤波去噪声  
 denoised = cv2.medianBlur(thresholded, 5)  
 # 对去噪声后的图像进行形态学开运算，进一步去除噪声  
 kernel = np.ones((5, 5), np.uint8)  
 morphed = cv2.morphologyEx(denoised, cv2.MORPH\_OPEN, kernel)  
 # 查找图像中的轮廓  
 contours, \_ = cv2.findContours(morphed, cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE)  
 # 提取轮廓的边界框  
 boxes = []  
 for contour in contours:  
 x, y, w, h = cv2.boundingRect(contour)  
 area = cv2.contourArea(contour)  
 if area > 100: # 过滤掉过小的轮廓  
 boxes.append([x, y, x + w, y + h])  
 # 转换为numpy数组  
 boxes = np.array(boxes)  
 # 使用非最大值抑制去除重叠的边界框  
 selected\_boxes = non\_max\_suppression(boxes)  
 # 在原始帧上绘制边界框  
 for box in selected\_boxes:  
 x1, y1, x2, y2 = box  
 cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 255, 0), 2)  
  
 # 将帧写入输出视频文件并显示  
 out.write(frame)  
 cv2.imshow('Motion Detection', frame)

* 1. **实验结果截图**

****

* 1. **实验小结**

这次实验首先逐帧读取视频，然后经过背景减去，图像去噪，非极大值抑制，绘制边框等过程成功实现了运动物体检测，了解运动检测的基本原理。学习使用 OpenCV 库进行视频处理和运动检测，熟悉非最大值抑制（Non-maximum Suppression）算法。