

# 基于目标检测及跟踪的实时人流量统计说明文档

## 1 环境配置

### 1.1 操作系统

- ubuntu18.04
- windows10

推荐在 ubuntu18.04 上运行。经测试，在 ubuntu 上运行帧数比在 win10 上高约 10 帧。

### 1.2 需要使用的库

- scipy
- numpy
- opencv-python
- matplotlib
- pycocotools
- pytorch>=1.0.1
- filterpy==1.4.1
- numba==0.38.1
- scikit-learn==0.19.1

## 2 代码运行方法

在代码所在文件夹的路径，执行：

**python sort\_yolov3.py**

以下参数必须输入：

**--v (str)**所要计数视频的路径

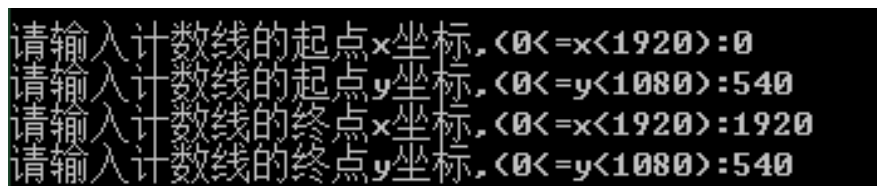
以下参数可选：

**--d (bool)**是否显示计数视频

**--w (bool)**是否保存计数视频

**--o (str)**保存视频路径

输入以上命令后会出现如图 2.1 所示的输入计数线端点坐标的提示。根据提示，输入计数线两个端点的坐标，计数线会设定为两端点连线及其延长线在视频范围内的部分，如图 2.2 所示。输入完成后即可开始计数。



```
请输入计数线的起点x坐标,<0<=x<1920>:0
请输入计数线的起点y坐标,<0<=y<1080>:540
请输入计数线的终点x坐标,<0<=x<1920>:1920
请输入计数线的终点y坐标,<0<=y<1080>:540
```

图 2.1 输入计数线端点坐标的提示

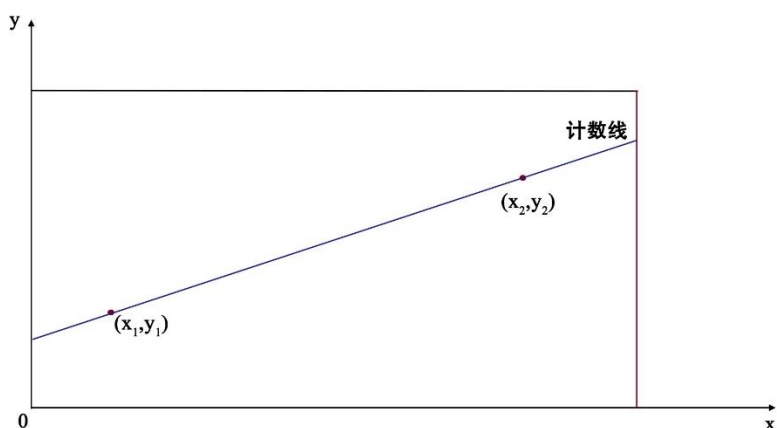


图 2.2 由端点坐标得到计数线的示意图

计数完成后，可以得到如图 2.2 所示的结果，分别为计数总用时以及双向的人数。

```
total time: 14.992015361785889s
正向人数: 0
反向人数: 10
```

图 2.3 代码执行结果

### 3 软件结构及原理详解

#### 3.1 功能及结构

本代码基于目标检测及跟踪，用于实时统计处于头顶点位置的摄像机所摄画面中的人数。代码中实现主要功能的模块包括目标检测模型 YOLOv3，多目标跟踪模型 SORT 以及人头计数策略。代码的逻辑结构如图 3.1 所示。

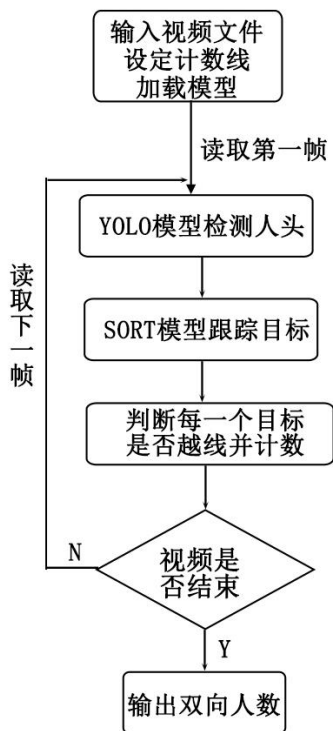


图 3.1 代码结构框图

## 3.2 输入输出项

### 3.2.1 输入项

命令行输入参数：

**--v (str)**所要计数视频的路径

**--d (bool)**是否显示计数视频

**--w (bool)**是否保存计数视频

**--o (str)**保存视频路径

执行后输入计数线起点及重点坐标。

具体说明详见第 2 节。

### 3.2.2 输出项

- 正向人数：det.sort.forward\_cnt
- 反向人数：det.sort.backward\_cnt

## 4 性能测试

### 4.1 测试序列信息

测试视频序列：test.mp4

测试视频时长：6'38"(398s)

人工统计正向人数（下至上）：27

人工统计反向人数（上至下）：50

### 4.2 测试环境

- ubuntu18.04
- CUDA10.0
- CUDnn7.5
- python3.7

### 4.3 测试结果

计数用时：335s

正向人数（下至上）：27

反向人数（上至下）：51