



(12) 发明专利

(10) 授权公告号 CN 114926984 B

(45) 授权公告日 2024.06.25

(21) 申请号 202210536933.7

(22) 申请日 2022.05.17

(65) 同一申请的已公布的文献号

申请公布号 CN 114926984 A

(43) 申请公布日 2022.08.19

(73) 专利权人 华南理工大学

地址 510640 广东省广州市天河区五山路
381号

(72) 发明人 漆巍巍 朱玉寒 李腾 张哲睿

邹振宇 侯颖琦

(74) 专利代理机构 哈尔滨市松花江联合专利商

标代理有限公司 23213

专利代理师 岳昕

(51) Int. Cl.

G08G 1/01 (2006.01)

(56) 对比文件

CN 107248290 A, 2017.10.13

CN 111462488 A, 2020.07.28

CN 113919731 A, 2022.01.11

审查员 何敏

权利要求书3页 说明书8页 附图4页

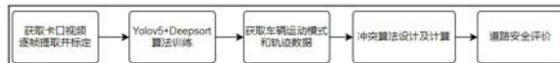
(54) 发明名称

一种实时交通冲突收集与道路安全评价方

法

(57) 摘要

本发明公开了一种实时交通冲突收集与道路安全评价方法,包括步骤:利用无人机或道路卡口监控对路况进行拍摄,获取视频数据,对收集到的视频图像进行特征标定,使用目标检测和目标追踪算法获得车辆轨迹数据,根据所得数据分析车辆轨迹特征及运动模式,计算视频中的交通冲突,获得交通冲突评价指标TTC,立数据集,分配权重指标,建立实时的道路安全评价机制本发明可以有效解决实时道路交通冲突信息收集和道路安全评价问题。



1. 一种实时交通冲突收集与道路安全评价方法,其特征在于,包括以下步骤:

1) 获取道路卡口视频数据,视频中包括道路基本情况、车辆行驶情况;

2) 基于所获得的视频数据,对数据集进行目标检测和目标跟踪;

3) 车辆轨迹特征及运动模式分析:提取出轨迹点中的车辆运动状态数据,对大量的交通微观数据进行筛选,获取道路车辆的距离碰撞时间,判定交通冲突类型,分析车辆的时空分布规律和车辆的运动行为;

4) 进行道路安全的实时评价:获得道路上车辆的冲突指标TTC后,针对不同的冲突类型设置不同的权重,获取道路安全指数,并对道路安全状况进行实时评价;

在步骤1)中,所述视频数据的获取需要通过道路卡口定点拍摄,所述道路基本情况包括道路宽度、道路车道数信息,所述车辆行驶情况包括车辆行驶方向、车辆类型、车辆数量信息;

步骤2)中,使用labelimg进行图像特征的标定,基于YOLOv5算法进行目标检测,基于DeepSort进行目标跟踪;

在步骤2)中,基于所获得的视频数据,对数据集进行目标检测和目标跟踪,具体如下:

a、视频特征标定:对视频中每一帧图像中的车辆进行标定以获取原始训练数据集,标定的数据包括车辆的类型、车辆大小信息;

b、目标检测:利用YOLOv5算法对经过车辆标定后的数据集进行目标检测算法训练,在卷积网络中对车辆进行特征提取、锚框回归,车辆特征包括车辆类型、大小、颜色信息;

c、目标追踪:利用DeepSort算法对YOLOv5算法的训练结果进行分析,实现对视频中每一帧车辆的位置与类型的识别,提取车辆的运动轨迹,并借助坐标系展现车辆的位置变化和运动模式,实现基于DeepSort的车辆轨迹追踪;

YOLOv5算法在卷积网络中对车辆进行特征提取、锚框回归,车辆特征包括车辆类型、大小、颜色信息,YOLOv5算法经过训练,实现对车辆类型、位置的自动识别;

DeepSort算法首先利用卡尔曼滤波预测轨迹,随后使用匈牙利算法将预测后的轨迹和当前帧中的检测目标进行匹配,最后对卡尔曼滤波进行更新;

在步骤3)中,包括以下步骤:

31) 提取轨迹数据:通过卷积神经网络提取出轨迹点中的车辆运动状态数据,对交通数据依据车辆之间的距离、方向和速度差参数,判断是否产生交通冲突,筛选出存在交通冲突的车辆,根据两车之间的距离、速度与加速度获取道路车辆的距离碰撞时间;

32) 建立数据集:将步骤2)检测出的车辆运动状态数据输出成文本格式,车辆运动状态数据包括车辆在摄像框的中心坐标 x 和 y 、车辆识别框的长宽、车辆类型及车辆标记ID;对于文本格式的数据,将车辆类型替换成数值型数据,再将所有车辆信息重新提取并加入对应的时间帧,形成数组形式;

33) 坐标透视变换:使用透视变换对数组中的车辆中心坐标即 x 和 y 进行坐标变换以获得车辆间的真实距离;

34) 对输出的车辆数据进行危险研判:提取出采样视频起始帧的车辆数据,放入起始列表,对起始列表中的车辆依次遍历,根据两车相对位置初步判断两车之间是否具有产生追尾冲突和侧击冲突的可能,将具有潜在冲突风险的车辆放入到处理列表中;

追尾冲突判别:

$$x_2 - x_1 \leq 1 \quad (2)$$

$$|y_1 - y_2| \leq d \quad (3) \quad |v_2 - v_1| > 0 \quad (4)$$

式中, x_1 和 y_1 为从起始帧中提取的前车中心坐标, x_2 和 y_2 为处于判别范围内的后车中心坐标, 1 为前后车中心点沿车道线方向相距离, d 为前后车中心点沿垂直车道线方向相距离, v_1 和 v_2 分别是前后车速度;

侧击冲突判别:

$$|x_1 - c_i| \leq b, i = 0, 1, 2 \dots \quad (5)$$

$$x_2 - x_1 \leq 1 \quad (6)$$

$$y_1 - y_2 \leq d \quad (7)$$

式中, x_1 和 y_1 为从起始帧中提取的前车中心坐标, c_i 为各车道线的横坐标, b 为判断前车处于变道行为的阈值, x_2 和 y_2 为处于判别范围内的后车中心坐标, 1 为前后车中心点沿车道线方向相距离, d 为前后车中心点沿垂直车道线方向相距离;

35) 计算冲突指标TTC:对处理列表中存放的可能产生交通冲突的前后车进行冲突指标TTC的计算;

追尾冲突:以处理列表中存放的可能产生交通冲突的时刻帧数为起始帧,获得该帧之后的前后车数据信息,前后车数据信息包括前后车中心坐标,计算速度和加速度随时间的变化,以后车加速度最大对应的时刻作为冲突点,计算该时刻前后车以各自对应速度行驶直到追尾所需时间作为冲突量化指标TTC:

$$v_1 = \frac{\Delta x_1}{t} \quad (8)$$

$$v_2 = \frac{\Delta x_2}{t} \quad (9)$$

$$a_2 = \frac{\Delta v_2}{t} \quad (9)$$

$$TTC = \frac{x_{2m} - x_{1m}}{v_{2m} - v_{1m}} \quad (10)$$

式中, Δx_1 和 Δx_2 分别为提取的相邻两个前后车中心点坐标沿行车方向的差, t 为提取的相邻两个车辆坐标之间的帧数差, v_1 和 v_2 为根据前后车中心点坐标计算得出的前后车各帧速度, a_2 为后车各帧计算所得加速度, $x_{2m} - x_{1m}$ 为后车加速度最大时后车和前车的相对距离, v_{1m} 为后车加速度最大时前车的速度, v_{2m} 为后车加速度最大时后车的速度;

侧击冲突:由于发生侧击冲突的两车行驶方向不同,其冲突轨迹交叉只有一个点,错过交叉点就不发生碰撞;

根据冲突风险车辆数据,利用相邻帧之间的坐标差求取行驶方向上前后两车的速度,制作车辆的速度列表;当速度列表中存在后车速度大于前车速度,则两车存在追尾冲突;利用车辆的速度列表中相邻速度求取后车加速度列表,取后车加速度最大时的帧数,并利用该帧内前后两车的运动数据计算出冲突指标TTC;

得到起始帧的追尾冲突和侧击冲突的冲突指标TTC后,删除起始帧,将下一帧作为起始帧再次判断并计算TTC,同时限制两车之间只产生一次交通冲突,对每一帧的冲突类型和

TTC进行计数统计,直到视频所有帧处理结束,获得整个视频的全部交通冲突信息;
坐标变换如下公式:

$$[x', y', z'] = [x, y, z] \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

式中, x, y 为原始图像坐标, x', y' 为经过透视变换的图像坐标, z 和 z' 都取1,变换矩阵为 3×3 形式, a_{ij} 均为变换矩阵参数,在原始图像和变换后的图像中分别取对应的四个边角点计算出变换矩阵, $i, j \in \{1, 2, 3\}$;

车辆间的侧击冲突采用如下方法计算TTC:

$$TTC_A = S_A / V_A \quad (12)$$

$$TTC_B = S_B / V_B \quad (13)$$

$$TTC = \begin{cases} TTC_A & \begin{cases} TTC_A \geq TTC_B \\ TTC_A < TTC_B + L_B / V_B \end{cases} \\ TTC_B & \begin{cases} TTC_A \leq TTC_B \\ TTC_B < TTC_A + L_A / V_A \end{cases} \end{cases} \quad (14)$$

式中, V_A, V_B 分别是两车速度, L_A, L_B 分别是两车车长, S_A, S_B 分别是两车距碰撞点的距离, TTC_A 和 TTC_B 分别是两车行驶至碰撞点的时间;

在步骤4)中根据计算出的交通冲突数据,获取道路安全评价指数,具体如下:

41) 统计道路上车辆交通冲突的类型和数量,建立数据集;

42) 计算道路安全指数RI,根据结果对道路进行实时安全评价,其计算公式如下:

$$RI = (R_1 * a_1 + R_2 * a_2) * c_1 + (S_1 * b_1 + S_2 * b_2) * c_2 \quad (15)$$

式中, R_1 是严重追尾冲突TTC均值, R_2 是一般追尾冲突TTC均值, S_1 是严重侧击冲突TTC均值, S_2 是一般侧击冲突TTC均值,其中 a_1 和 a_2 分别为严重追尾冲突和一般追尾冲突的权重; b_1 和 b_2 分别是严重侧击冲突和一般侧击冲突的权重; c_1 和 c_2 分别是追尾冲突与侧击冲突的权重。

一种实时交通冲突收集与道路安全评价方法

技术领域

[0001] 本发明涉及交通冲突理论、道路安全评价的及时性与适用性的技术领域,尤其是指一种实时交通冲突收集与道路安全评价方法。

背景技术

[0002] 随着城市化的高速发展和机动车数量的快速增长,车流的随机性和道路的时空特性导致车辆运行过程中难以避免交通冲突的产生,不仅可能造成交通拥堵,严重时甚至会导致交通事故,造成人员和财产的损失。传统的对于交通冲突数据的获取方式主要分为基于GPS导航系统和基于视频图像处理两类。

[0003] 对于利用GPS及网联车进行冲突的获取,一方面存在车辆定位技术精度有限、易受环境影响、存在一定延迟等问题,另一方面还会受路段网联车渗透率影响。往往需要耗费一定的时间进行时间误差、距离误差的矫正,其精度非常有限。

[0004] 而对于基于视频图像处理的方法,目前普遍采用的是利用tracker软件对无人机拍摄的视频进行标定追踪,但是该方法由于无人机续航等问题的限制,无法做到长时间的固定拍摄,并且全程需要人为控制与调整,若想实现大量路段交通冲突数据连续的采集,需耗费大量人力物力。虽可以实现很高的精度,但是由于受到系统适配等问题,人工标定的过程繁琐且耗时长。故使用拍摄视频加软件处理的方式获得交通冲突数据虽在精度上得到大幅优化,但是需要较长的处理周期与人工处理,无法做到交通冲突数据的实时获取与道路安全水平的实时评价(机动车交通冲突技术研究综述_朱顺应)。

发明内容

[0005] 发明的目的在于克服现有的交通冲突收集方式的不足,提出一种实时交通冲突收集与道路安全评价方法,利用目标识别跟踪算法的优势,实现对道路上车辆交通冲突的实时、高效收集,并对道路进行实时安全评价,为车辆出行和交通管控提供依据,可以提供一个真实环境下、长时间的车辆连续轨迹追踪的大数据获取手段,弥补GPS在精度上的不足,并达到比tracker更高的效率。

[0006] 本发明至少通过如下技术方案之一实现。

[0007] 一种实时交通冲突收集与道路安全评价方法,包括以下步骤:

[0008] 1) 获取道路卡口视频数据,视频中包括道路基本情况、车辆行驶情况;

[0009] 2) 基于所获得的视频数据,对数据集进行目标检测和目标跟踪;

[0010] 3) 车辆轨迹特征及运动模式分析:提取出轨迹点中的车辆运动状态数据,对大量的交通微观数据进行筛选,获取道路车辆的距离碰撞时间(TTC),判定交通冲突类型,分析车辆的时空分布规律和车辆的运动行为;

[0011] 4) 进行道路安全的实时评价:获得道路上车辆的冲突指标TTC后,针对不同的冲突类型设置不同的权重,获取道路安全指数,并对道路安全状况进行实时评价。

[0012] 进一步地,在步骤1)中,所述视频数据的获取需要通过道路卡口定点拍摄,所述道

路基本情况包括道路宽度、道路车道数信息,所述车辆行驶情况包括车辆行驶方向、车辆类型、车辆数量信息。

[0013] 进一步地,步骤2)中,使用labelimg进行图像特征的标定,基于YOLOv5算法进行目标检测,基于DeepSort进行目标跟踪。

[0014] 进一步地,在步骤2)中,基于所获得的视频数据,对数据集进行目标检测和目标跟踪,具体如下:

[0015] a、视频特征标定:对视频中每一帧图像中的车辆进行标定以获取原始训练数据集,标定的数据包括车辆的类型、车辆大小信息;

[0016] b、目标检测:利用YOLOv5算法对经过车辆标定后的数据集进行目标检测算法训练,在卷积网络中对车辆进行特征提取、锚框回归,车辆特征包括车辆类型、大小、颜色信息;

[0017] c、目标追踪:利用DeepSort算法对YOLOv5算法的训练结果进行分析,实现对视频中每一帧车辆的位置与类型的识别,提取车辆的运动轨迹,并借助坐标系展现车辆的位置变化和运动模式,实现基于DeepSort的车辆轨迹追踪。

[0018] 进一步地,YOLOv5算法在卷积网络中对车辆进行特征提取、锚框回归,车辆特征包括车辆类型、大小、颜色信息,YOLOv5算法经过训练,实现对车辆类型、位置的自动识别。

[0019] 进一步地,DeepSort算法首先利用卡尔曼滤波预测轨迹,随后使用匈牙利算法将预测后的轨迹和当前帧中的检测目标进行匹配,最后对卡尔曼滤波进行更新。

[0020] 进一步地,在步骤3)中,包括以下步骤:

[0021] 31)提取轨迹数据:通过卷积神经网络提取出轨迹点中的车辆运动状态数据,对交通数据依据车辆之间的距离、方向和速度差参数,判断是否产生交通冲突,筛选出存在交通冲突的车辆,根据两车之间的距离、速度与加速度获取道路车辆的距离碰撞时间(TTC);

[0022] 32)建立数据集:将步骤2)检测出的车辆运动状态数据输出成文本格式,车辆运动状态数据包括车辆在摄像框的中心坐标x和y、车辆识别框的长宽、车辆类型及车辆标记ID;对于文本格式的数据,将车辆类型替换成数值型数据,再将所有车辆信息重新提取并加入对应的时间帧,形成数组形式;

[0023] 33)坐标透视变换:使用透视变换对数组中的车辆中心坐标即x和y进行坐标变换以获得车辆间的真实距离;

[0024] 34)对输出的车辆数据进行危险研判:提取出采样视频起始帧的车辆数据,放入起始列表,对起始列表中的车辆依次遍历,根据两车相对位置初步判断两车之间是否具有产生追尾冲突和侧击冲突的可能,将具有潜在冲突风险的车辆放入添加到处理列表中;

[0025] 追尾冲突判别:

$$[0026] \quad x_2 - x_1 \leq 1 \quad (2)$$

$$[0027] \quad |y_1 - y_2| \leq d \quad (3)$$

$$[0028] \quad v_2 - v_1 > 0 \quad (4)$$

[0029] 式中, x_1 和 y_1 为从起始帧中提取的前车中心坐标, x_2 和 y_2 为处于判别范围内的后车中心坐标,1为前后车中心点沿车道线方向相差距离,d为前后车中心点沿垂直车道线方向相差距离, v_1 和 v_2 分别是前后车速度;

[0030] 侧击冲突判别:

$$[0031] \quad |x_1 - c_i| \leq b, i = 0, 1, 2, \dots \quad (5)$$

$$[0032] \quad x_2 - x_1 \leq l \quad (6)$$

$$[0033] \quad |y_1 - y_2| \leq d \quad (7)$$

[0034] 式中, x_1 和 y_1 为从起始帧中提取的前车中心坐标, c_i 为各车道线的横坐标, b 为判断前车处于变道行为的阈值, x_2 和 y_2 为处于判别范围内的后车中心坐标, l 为前后车中心点沿车道线方向相差距离, d 为前后车中心点沿垂直车道线方向相差距离;

[0035] 35) 计算冲突指标TTC:对处理列表中存放的可能产生交通冲突的前后车进行冲突指标TTC的计算;

[0036] 追尾冲突:以 d 中危险研判中的时刻帧数为起始帧,获得该帧之后的前后车数据信息,前后车数据信息包括前后车中心坐标,计算速度和加速度随时间的变化,以后车加速度最大对应的时刻作为冲突点,计算该时刻前后车以各自对应速度行驶直到追及所需时间作为冲突量化指标TTC:

$$[0037] \quad v_1 = \frac{\Delta x_1}{t} \quad (8)$$

$$[0038] \quad v_2 = \frac{\Delta x_2}{t} \quad (9)$$

$$[0039] \quad a_2 = \frac{\Delta v_2}{t} \quad (10)$$

$$[0040] \quad TTC = \frac{x_{2m} - x_{1m}}{v_{2m} - v_{1m}} \quad (11)$$

[0041] 式中, Δx_1 和 Δx_2 分别为提取的相邻两个前后车中心点坐标沿行车方向的差, t 为提取的相邻两个车辆坐标之间的帧数差, v_1 和 v_2 为根据前后车中心点坐标计算得出的前后车各帧速度, a_2 为后车各帧计算所得加速度, $x_{2m} - x_{1m}$ 为后车加速度最大时后车和前车的相对距离, v_{1m} 和 v_{2m} 为前后车加速度最大时后车的速度;

[0042] 侧击冲突:由于发生侧击冲突的两车行驶方向不同,其冲突轨迹交叉只有一个点,错过交叉点就不发生碰撞;

[0043] 根据冲突风险车辆数据,利用相邻帧之间的坐标差求取行驶方向上前后两车的速度,制作车辆的速度列表;当列表中存在后车速度大于前车速度,则两车存在追尾冲突;利用车辆的速度列表中相邻速度求取后车加速度列表,取后车加速度最大时的帧数,并利用该帧内前后两车的运动数据计算处冲突指标TTC;

[0044] 得到起始帧的追尾冲突的侧击冲突的冲突指标TTC后,删除起始帧,将下一帧作为起始帧再次判断并计算TTC,同时限制两车之间只产生一次交通冲突,对每一帧的冲突类型和TTC进行计数统计,直到视频所有帧处理结束,获得整个视频的全部交通冲突信息。

[0045] 进一步地,坐标变换如下公式:

$$[0046] \quad [x', y', z'] = [x, y, z] \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

[0047] 式中, x, y 为原始图像坐标, x', y' 为经过透视变换的图像坐标, z 和 z' 都取1,变换矩阵为 3×3 形式, a_{ij} 均为变换矩阵参数,在原始图像和变换后的图像中分别取对应的四个边角点计算出变换矩阵, $i, j \in \{1, 2, 3\}$ 。

[0048] 进一步地,车辆间的侧击冲突采用如下方法计算TTC:

$$[0049] \quad TTC_A = S_A / V_A \quad (12)$$

$$[0050] \quad TTC_B = S_B / V_B \quad (13)$$

$$[0051] \quad TTC = \begin{cases} TTC_A & \begin{cases} TTC_A \geq TTC_B \\ TTC_A \leq TTC_B + L_B / V_B \end{cases} \\ TTC_B & \begin{cases} TTC_A < TTC_B \\ TTC_B \leq TTC_A + L_A / V_A \end{cases} \end{cases} \quad (14)$$

[0052] 式中, V_A 、 V_B 分别是两车速度, L_A 、 L_B 分别是两车车长, S_A 、 S_B 分别是两车距碰撞点的距离, TTC_A 和 TTC_B 分别是两车行驶至碰撞点的时间。

[0053] 进一步地,在步骤4)中根据计算出的交通冲突数据,获取道路安全评价指数,具体如下:

[0054] 41) 统计道路上车辆交通冲突的类型和数量,建立数据集;

[0055] 42) 计算道路安全指数RI,根据结果对道路进行实时安全评价,其计算公式如下:

$$[0056] \quad RI = (R_1 * a_1 + R_2 * a_2) * c_1 + (S_1 * b_1 + S_2 * b_2) * c_2 \quad (15)$$

[0057] 式中, R_1 是严重追尾冲突, R_2 是一般追尾冲突, S_1 是严重侧击冲突; S_2 是一般侧击冲突,其中 a_1 和 a_2 分别为严重追尾冲突和一般追尾冲突的权重; b_1 和 b_2 分别是严重侧击冲突和一般侧击冲突的权重; c_1 和 c_2 分别是追尾冲突与侧击冲突的权重。以上权重系数可以根据路段的各类型冲突所造成的财产损失的历史统计数据占比所得。

[0058] 本发明所述一种实时交通冲突收集与道路安全评价方法,采用以上技术方案与现有技术相比,具有以下技术效果:

[0059] 本发明基于实时采集的视频图像数据进行分析,有效克服传统方法精度不佳与存在时延的问题,可以解决实时道路交通冲突收集和道路安全评价问题,有效提高道路交通安全评价结果的可信度和可靠性,降低道路交通安全评价结果的偏差,并且对于城市道路交通安全评价有较好的适用性。

附图说明

[0060] 图1为实施例的实时交通冲突收集与道路安全评价方法整体结构示意图;

[0061] 图2为实施例所用追尾冲突原理示意图;

[0062] 图3为实施例追尾冲突算法示意图;

[0063] 图4为实施例所用侧击冲突原理示意图;

[0064] 图5为实施例侧击冲突算法示意图;

[0065] 图6为实施例自适应参数调整原理示意图。

具体实施方式

[0066] 下面结合具体实例对本发明作进一步说明:

[0067] 实施例1

[0068] 调用东莞市天桥路口往张坑村方向的道路监控视频进行实例说明,如图1、图4所示本发明所提供的实时交通冲突收集与道路安全评价方法,包括以下步骤:

[0069] 1) 获取道路卡口视频数据,视频中要包含道路基本情况、车辆行驶情况。所述视频

数据的获取需要通过道路卡口定点拍摄,所述道路基本情况应该包括道路宽度、道路车道数等精确信息,所述车辆行驶情况应该包括车辆行驶方向、车辆类型、车辆数量等信息。

[0070] 视频中道路基本情况为双向四车道,车道宽度为3.5m,行驶车辆包括汽车、公交、货车,车辆数总计102。

[0071] 2) 实现目标检测和目标追踪算法,使用labeling进行图像特征的标定,基于YOLOv5算法进行目标检测,基于DeepSort进行目标跟踪。

[0072] 基于所获得的视频数据,对数据集进行目标检测和目标跟踪,具体如下:

[0073] 21) 视频特征标定:对视频中每一帧图像中的车辆进行标定是获取原始训练数据的重要手段,为了更加精确的识别车辆和其运动轨迹,需要大量的原始数据用作训练,标定的数据中应包含车辆的类型、车辆大小等信息。

[0074] 22) 目标检测:利用YOLOv5对经过车辆标定后的数据集进行目标检测算法训练,在卷积网络中对车辆进行特征提取、锚框回归,车辆特征应包含车辆类型、大小、颜色等信息。经过足够次数的训练,实现对车辆类型、位置的精确自动识别。

[0075] YOLOv5算法:YOLO算法为目前目标检测领域使用较为广泛的程序算法,它在目标检测时无需提取候选区域,可以直接进行在卷积网络中进行特征提取、锚框回归。YOLOv5算法主要由三大主要模块构成,他们是backbone模块,neck模块和head模块,backbone模块主要功能是在不同图像细粒度上聚合并形成图像特征的卷积神经网络,neck模块可以建立一系列混合和组合图像特征的网络层,并将图像特征传递到预测层。Head模块主要功能为对图像特征进行预测,生成边界框和并预测类别。

[0076] 23) 目标追踪:利用DeepSort算法对YOLOv5的训练结果进行分析,实现对视频中每一帧车辆的位置与类型的精准识别,提取车辆的运动轨迹,并借助坐标系展现车辆的位置变化和运动模式,实现基于DeepSort的车辆轨迹追踪。

[0077] DeepSort算法:DeepSort追踪算法是在Sort算法的数据关联基础上加入外观模型优化的来,改进了匈牙利算法的匹配程度,对被遮掩的物体识别度较。其关键步骤包括三个,首先利用卡尔曼滤波预测轨迹,随后使用匈牙利算法将预测后的tracks和当前帧中的detections进行匹配(IOU匹配),最后对卡尔曼滤波进行更新。

[0078] 3) 车辆轨迹特征及运动模式分析:提取出轨迹点中的车辆运动状态数据,对大量的交通微观数据进行筛选,获取道路车辆的距离碰撞时间(TTC),判定交通冲突类型,分析车辆的时空分布规律和车辆的运动行为。

[0079] 利用提取到的车辆轨迹特征,掌握车辆的时空分布规律,进行距离碰撞时间(TTC)的计算,分析车辆的时空分布规律和车辆的运动行为。具体如下:

[0080] 31) 提取轨迹数据:通过卷积神经网络提取出轨迹点中的车辆运动状态数据,对大量的交通微观数据进行筛选,获取道路车辆的距离碰撞时间(TTC)。

[0081] 32) 建立数据集:将上文所述YOLOv5和DeepSort算法检测出的车辆运动状态数据输出成文本格式,数据信息包括车辆在摄像框的中心坐标x和y、车辆识别框的长宽、车辆类型及车辆标记ID。并对上述输出数据进行处理,将车辆类型替换成数值型数据,再将所有车辆信息重新提取并加入对应的时间帧,形成数组形式。

[0082] 数据信息格式如下所示:

[0083] $[[x, y, h, w, type, ID], \dots, [x, y, h, w, type, ID]]$

$$[0084] \quad \begin{bmatrix} [x, y, h, w, type, ID, t] & \cdots & [x, y, h, w, type, ID, t] \\ & \vdots & \\ [x, y, h, w, type, ID, t] & \cdots & [x, y, h, w, type, ID, t] \end{bmatrix}$$

[0085] 式中, $[x, y, h, w, type, ID]$ 为一个六维向量, 以左上角为中心, x 是以右侧为正轴的车辆中心横坐标, y 是以下侧为正轴的车辆中心纵坐标, 检测框长为 h , 宽为 w , 车辆类型为 $type$, 车辆序号标记为 ID , 在六维向量的最后增加一维时间帧 t , 形成 7 维向量, 多个 7 维向量组成同一帧的数组, 多行数组组成视频所有帧的车辆运动状态数据。

[0086] 33) 坐标透视变换: 由于道路卡口视频监控架设在道路上方, 朝向或背对车辆行驶方向斜向下拍摄, 摄像机高度、拍摄角度会对检测所得车辆中心坐标产生影响, 使用透视变换对数组中的车辆中心坐标即 x 和 y 进行坐标变换以获得车辆间的真实距离。

$$[0087] \quad [x', y', z'] = [x, y, z] \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (1)$$

[0088] 式中, x, y 为原始图像坐标, x', y' 为经过透视变换的图像坐标, z 和 z' 都取 1, 变换矩阵为 3×3 形式, a_{ij} 均为变换矩阵参数。可在原始图像和变换后的图像中分别取对应的四个边角点计算出变换矩阵, $i, j \in \{1, 2, 3\}$ 。

[0089] 34) 对输出的车辆数据进行危险研判: 提取出采样视频起始帧的车辆数据, 放入起始列表, 对起始列表中的车辆依次遍历, 根据两车相对位置初步判断两车之间是否具有产生追尾冲突和侧击冲突的可能, 将具有潜在冲突风险的车辆放入添加到处理列表中。

[0090] 追尾冲突判别:

$$[0091] \quad x_2 - x_1 \leq 1 \quad (2)$$

$$[0092] \quad |y_1 - y_2| \leq d \quad (3)$$

$$[0093] \quad v_2 - v_1 > 0 \quad (4)$$

[0094] 式中, x_1 和 y_1 为从起始帧中提取的前车中心坐标, x_2 和 y_2 为处于判别范围内的后车中心坐标, 1 为前后车中心点沿车道线方向相差距离, 取 150 个像素点, 实际距离为 1.5m, d 为前后车中心点沿垂直车道线方向相差距离, 取 2000 个像素点, 实际距离为 1.5m, v_1 和 v_2 分别是前后车速度。

[0095] 侧击冲突判别:

$$[0096] \quad |x_1 - c_i| \leq b, i = 0, 1, 2, \dots \quad (5)$$

$$[0097] \quad x_2 - x_1 \leq 1 \quad (6)$$

$$[0098] \quad |y_1 - y_2| \leq d \quad (7)$$

[0099] 式中, x_1 和 y_1 为从起始帧中提取的前车中心坐标, c_i 为各车道线的横坐标, b 为判断前车处于变道行为的阈值, 取 40 个像素点, x_2 和 y_2 为处于判别范围内的后车中心坐标, 1 为前后车中心点沿车道线方向相差距离, d 为前后车中心点沿垂直车道线方向相差距离。

[0100] 35) 计算冲突指标 TTC: 对处理列表中存放的可能产生交通冲突的前后车进行冲突指标 TTC 的计算。

[0101] 追尾冲突: 以 d 中危险研判中的时刻帧数为起始帧, 获得该帧之后的前后车数据信息, 包括前后车中心坐标等, 计算速度和加速度随时间的变化, 以后车加速度最大对应的时刻作为冲突点, 计算该时刻前后车以各自对应速度行驶直到追及所需时间作为冲突量化指

标TTC。

$$[0102] \quad v_1 = \frac{\Delta x_1}{t} \quad (8)$$

$$[0103] \quad v_2 = \frac{\Delta x_2}{t} \quad (9)$$

$$[0104] \quad a_2 = \frac{\Delta v_2}{t} \quad (10)$$

$$[0105] \quad TTC = \frac{x_{2m} - x_{1m}}{v_{2m} - v_{1m}} \quad (11)$$

[0106] 式中 Δx_1 和 Δx_2 分别为提取的相邻两个前后车中心点坐标沿行车方向的差, t 为提取的相邻两个车辆坐标之间的帧数差, v_1 和 v_2 为根据前后车中心点坐标计算得出的前后车各帧速度, a_2 为后车各帧计算所得加速度, $x_{2m} - x_{1m}$ 为后车加速度最大时后车和前车的相对距离, v_{1m} 和 v_{2m} 为前后车加速度最大时后车的速度。

[0107] 侧击冲突: 由于发生侧击冲突的两车行驶方向不同, 其冲突轨迹交叉只有一个点, 错过交叉点就不发生碰撞, 采用如下方法计算TTC:

$$[0108] \quad TTC_A = S_A / V_A \quad (12)$$

$$[0109] \quad TTC_B = S_B / V_B \quad (13)$$

$$[0110] \quad TTC = \begin{cases} TTC_A & \begin{cases} TTC_A \geq TTC_B \\ TTC_A \leq TTC_B + L_B / V_B \end{cases} \\ TTC_B & \begin{cases} TTC_A < TTC_B \\ TTC_B \leq TTC_A + L_A / V_A \end{cases} \end{cases} \quad (14)$$

[0111] 式中, V_A 、 V_B 分别是两车速度, L_A 、 L_B 分别是两车车长, S_A 、 S_B 分别是两车距碰撞点的距离, TTC_A 和 TTC_B 分别是两车行驶至碰撞点的时间。

[0112] 根据冲突风险车辆数据, 利用相邻帧之间的坐标差求取行驶方向上前后两车的速度, 制作车辆的速度列表。当列表中存在后车速度大于前车速度, 则两车存在追尾冲突。利用车辆的速度列表中相邻速度求取后车加速度列表, 取后车加速度最大时的帧数, 并利用该帧内前后两车的运动数据计算处冲突指标TTC。如图2所示, 当同一车道上的两辆车后车 $V_B > V_A$ 时, 后车为避免碰撞而采取制动措施, 取减速时的加速度 A_B 最大时刻 T 为冲突避险时刻, 此时追尾冲突 $TTC = L_{AB} / (V_B + V_A)$ 。得到起始帧的追尾冲突的侧击冲突的冲突指标TTC后, 删除起始帧, 将下一帧作为起始帧再次判断并计算TTC, 同时限制两车之间只产生一次交通冲突, 对每一帧的冲突类型和TTC进行计数统计, 直到视频所有帧处理结束, 获得整个视频的全部交通冲突信息。

[0113] 4) 进行道路安全的实时评价: 获得道路上车辆的冲突指标TTC后, 针对不同的冲突类型设置不同的权重, 获取道路安全指数, 并对道路安全状况进行实时评价。

[0114] 根据计算出的交通冲突数据, 获取道路安全评价指数, 具体如下:

[0115] 41) 统计道路上车辆交通冲突的类型和数量, 对于追尾冲突, 当 $TTC \leq 2.7$, 则判定为严重追尾冲突, 否则为一般追尾冲突; 对于侧击冲突, 当 $TTC \leq 3.1$ 时, 则判定为严重侧击冲突, 否则为一般侧击冲突。根据视频数据, 判定显示追尾冲突数为0, 严重侧击冲突个数为14, 一般侧击冲突个数为5

[0116] 42) 计算道路安全指数RI, 根据结果对道路进行实时安全评价。其计算公如下:

$$[0117] \quad RI = (R_1 * a_1 + R_2 * a_2) * c_1 + (S_1 * b_1 + S_2 * b_2) * c_2 \quad (15)$$

[0118] 式中, R_1 是严重追尾冲突TTC均值, R_2 是一般追尾冲突TTC均值, S_1 是严重侧击冲突TTC均值; S_2 是一般侧击冲突TTC均值; a_1 和 a_2 分别为严重追尾冲突和一般追尾冲突的权重; b_1 和 b_2 分别是严重侧击冲突和一般侧击冲突的权重; c_1 和 c_2 分别是追尾冲突与侧击冲突的权重。以上权重根据路段的各类冲突造成财产损失的历史数据占比所得, 取 a_1 为 0.65、 a_2 为 0.35、 b_1 为 0.61、 b_2 为 0.39、 c_1 为 0.54、 c_2 为 0.46。例如 R_1 取 0, R_2 取 0, S_1 取 1.38, S_2 取 3.66, 相应计算得到检测时段内的安全指数 RI 为 1.04。此外, 在视频检测的时段内, 实时的 RI 指数在 0.76 ~ 1.24 的范围内浮动。以上信息表明监测视频中显示的横向车流较多, 道路行车安全较差, 需小心驾驶。

[0119] 实施例2

[0120] 在东莞市西城一期路口往东部快线方向的路口监控进行检测, 具体的透视变换矩阵根据实际的路口勘测和CAD图纸进行确定, 经过与上例类似的检测计算后得到路段的 RI 指数在 0.6 ~ 0.94 之间的范围内浮动, 可以得出该路口的安全水平相对较好, 但仍然需要注意对向行车安全。

[0121] 实施例3

[0122] 最后在中心小学路口往新四村方向的路口监控进行检测, 具体的透视变换矩阵也根据实际的路口勘测和CAD图纸进行确定, 经过与例一类似的检测计算后得到路段的 RI 指数在 0.8 ~ 1.31 之间的范围内浮动, 可以得出该路口的安全水平比较一般, 相邻相位间未排空车辆产生了较多的侧击冲突, 车辆行驶中需要注意周期内未排空车辆。

[0123] 以上公开的本发明优选实施例只是用于帮助阐述本发明。优选实施例并没有详尽叙述所有的细节, 也不限制该发明仅为所述的具体实施方式。显然, 根据本说明书的内容, 可作很多的修改和变化。本说明书选取并具体描述这些实施例, 是为了更好地解释本发明的原理和实际应用, 从而使所属技术领域技术人员能很好地理解和利用本发明。本发明仅受权利要求书及其全部范围和等效物的限制。

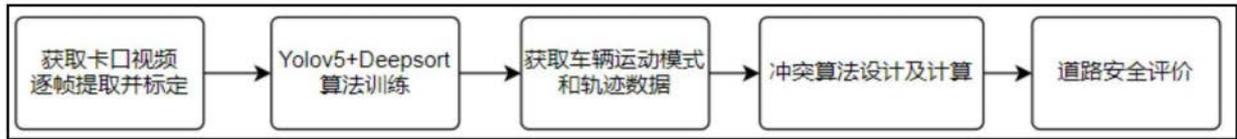


图1

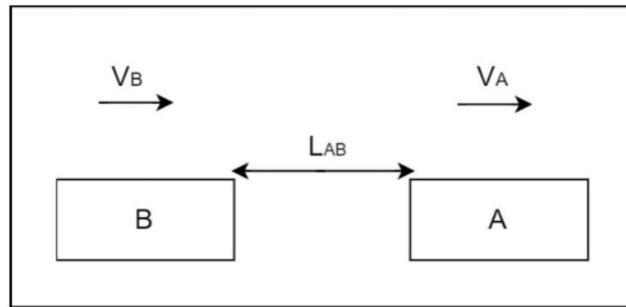


图2

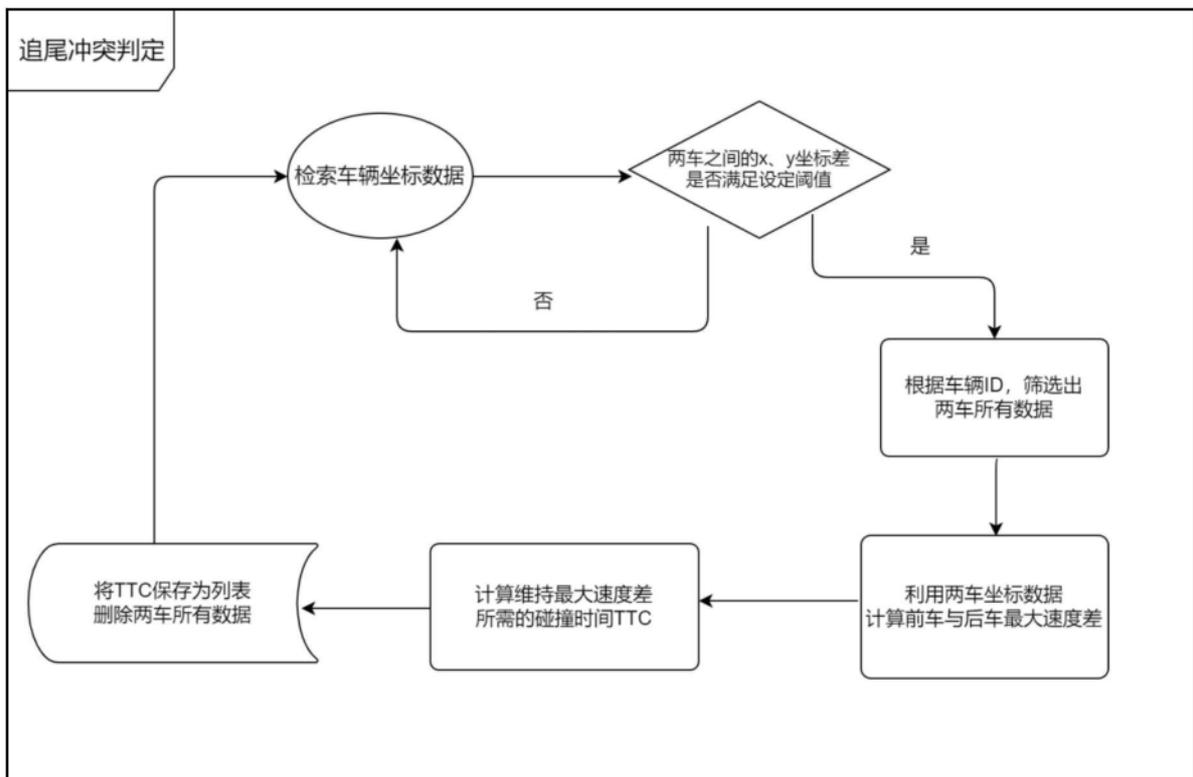


图3

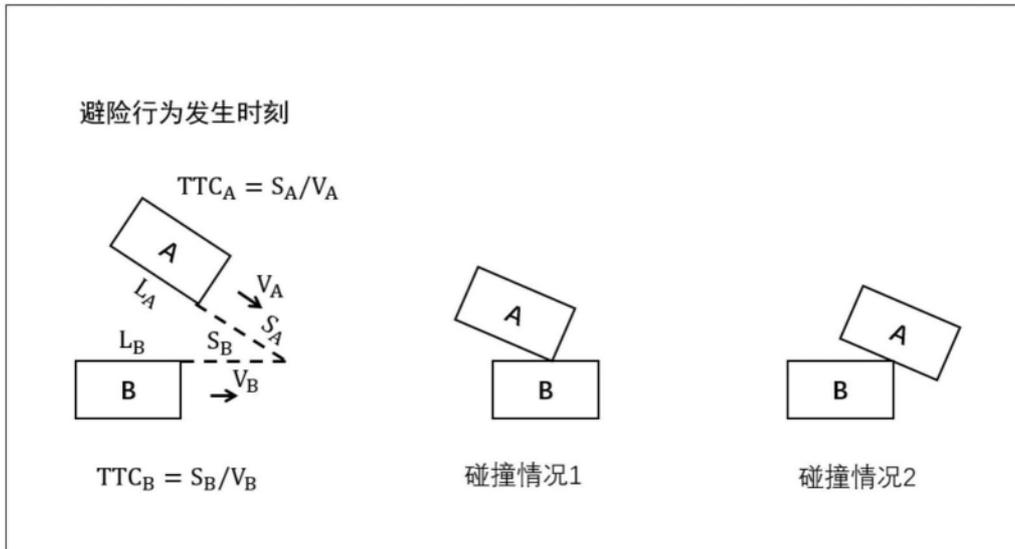


图4

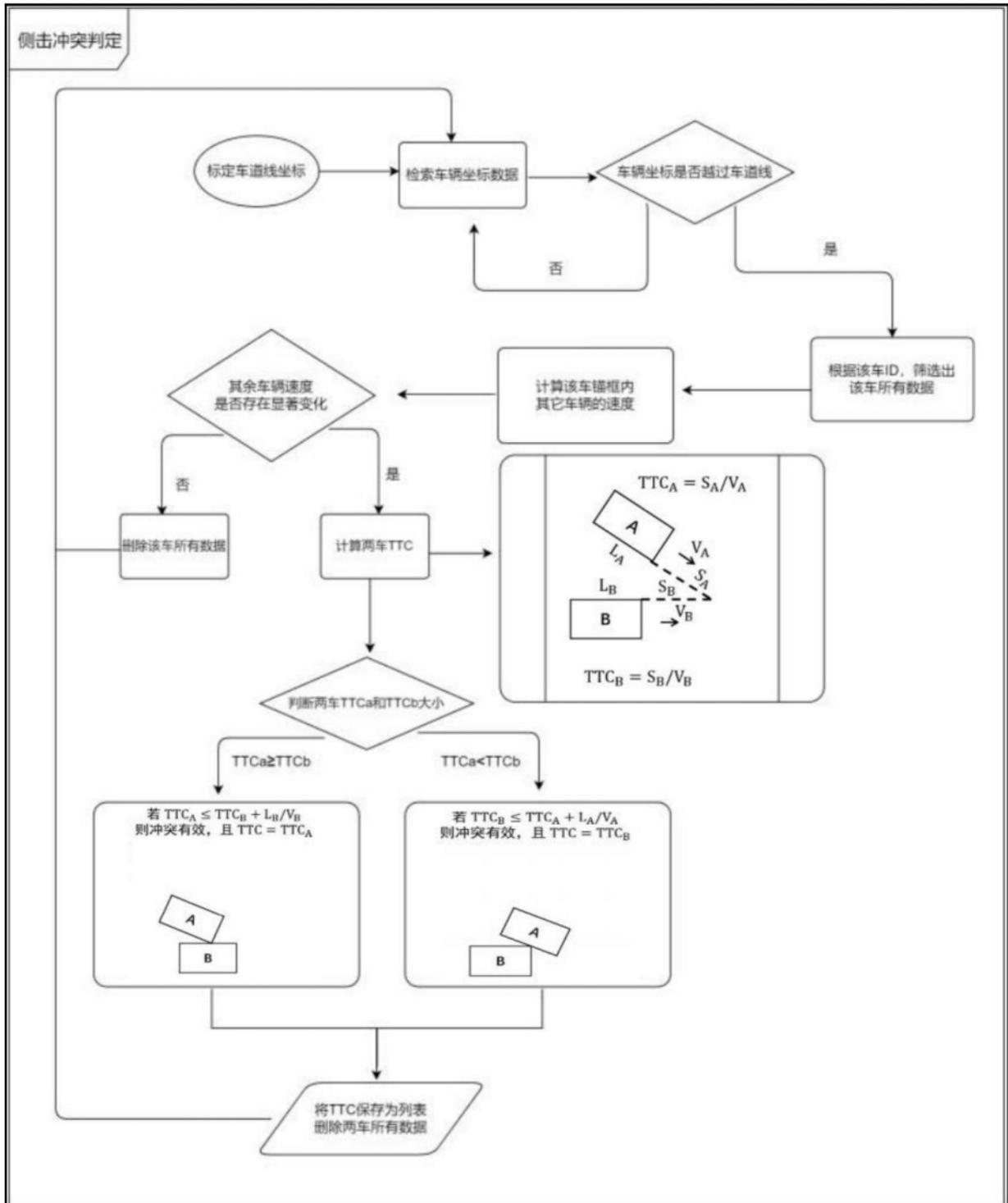


图5

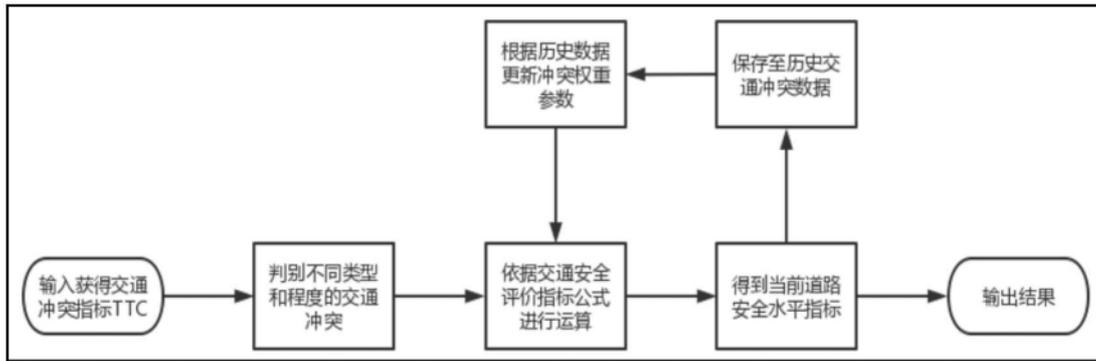


图6