

知荐 | ADAS系统中的弯道巡航控制算法（二）

知荐 | ADAS系统中的弯道巡航控制算法（一）

ADAS系统开发中不仅要考虑好的控制算法，更要考虑好的感知融合算法，因为对于目标识别来讲，不精准的目标识别往往意味着不精准的甚至错误的控制输出。现有的研究中，大部分的目标识别算法基本都针对ADAS系统自车与前方目标车处于直道或者较小曲率半径路段的情况，而实际情况时，对于大曲率半径弯道，系统却经常出现误识别或漏识别，这对车辆的安全运行存在不利影响。

针对上述需求，并考虑前文所述中 ADAS系统自身加减速控制的响应算法已较成熟，本文中将重点研究弯道目标感知算法，即系统如何处理感知的前方车辆行驶状态及对弯道跟车控制的输出问题。

好的弯道目标跟踪与识别算法需要综合考虑如下三个方面：

- 1、如何利用自车的车速、横摆角速度、转向角度、横向加速度计算自车行驶轨迹；
- 2、如何利用传感器探测到的前车的车速、横向加速度、前车与自车的相对角度、相对距离预估前车行驶轨迹；
- 3、如何利用前车的轨迹斜率及其变化特征识别前车的换道和进出弯道行为；
- 4、如何更好地区分弯道边沿、护栏、草坪等信息。

以上场景对于弯道目标控制中起着决定性的作用，下面分别阐述相应的实现方案。

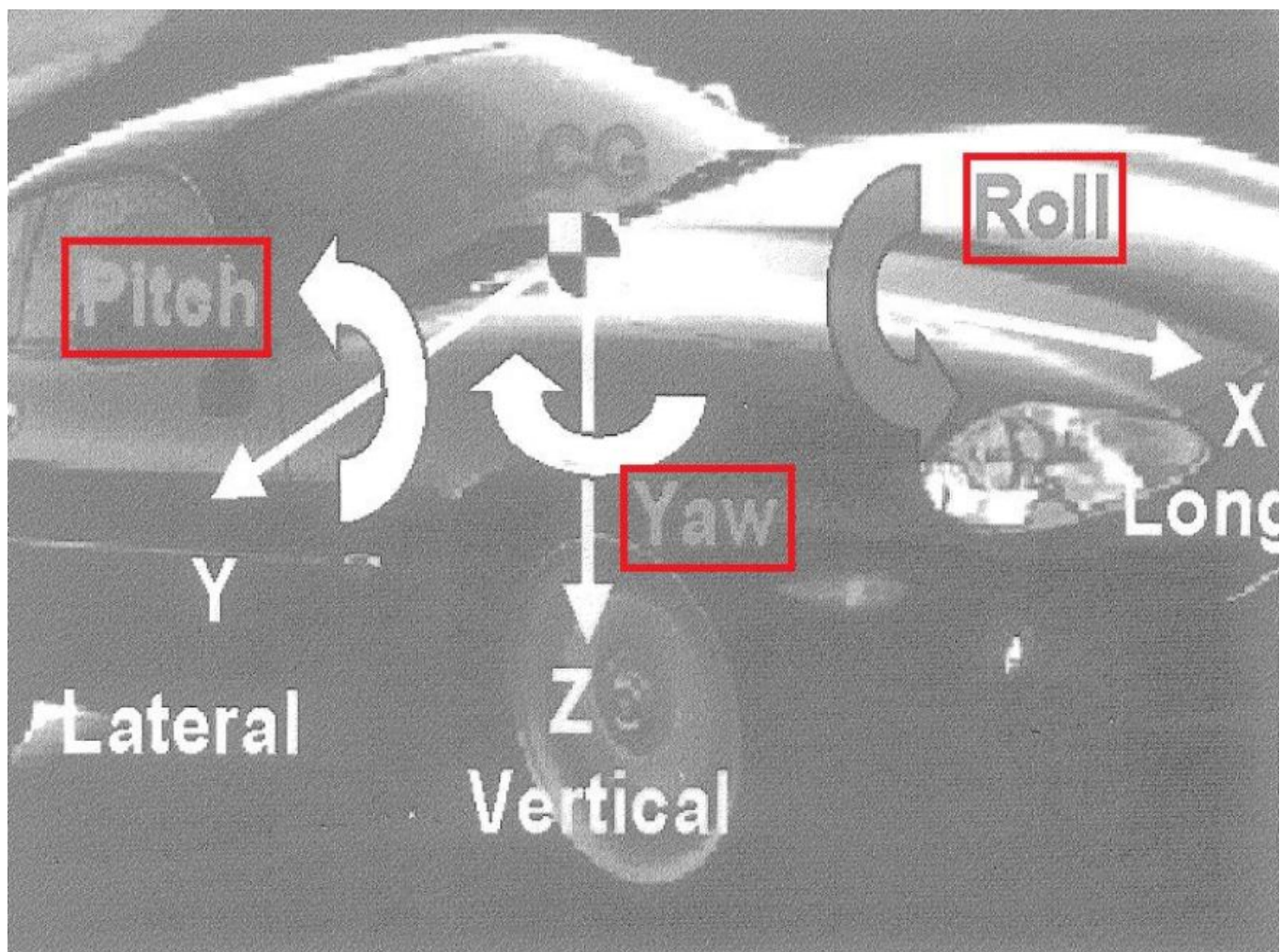
车辆行驶轨迹计算

- 自车基础坐标系定义

本文中首先建立了建立如下基础坐标系，采用美国机动车工程师学会(SAE)的设定，该坐标系以自车车辆中心为原点，建立如下坐标系：

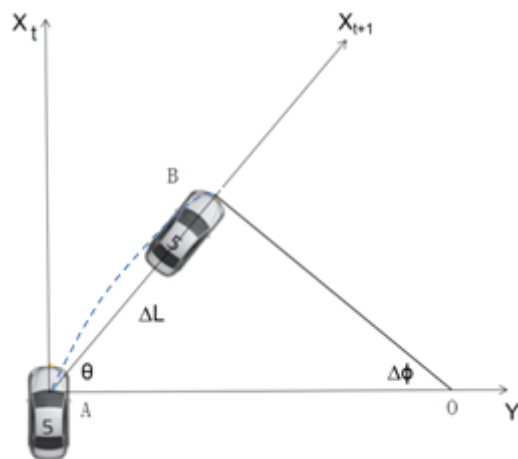
- 1) 以车辆行驶方向的前后为 x 轴，且向前为正；
- 2) 以车辆行驶方向的两侧为 y 轴，且向右为正；
- 3) 以车辆行驶方向的上下方向为 z 轴，且向上为正。

其示意如下图：



• 车辆入弯行驶定义

车辆行驶过程中，其轨迹表征为不规则的曲线段组成，而将各曲线段按照足够小的步长进行分割，则每个小的步长内曲线段又可以近似看为直线。其中每一小分割按照坐标系表示如下图：假设车辆一定速度经过以O为圆心的弯道，车辆行驶轨迹为一定弧长，但是在划分为足够小的步长后，该车辆在较小轨迹中可看成一定直线。如下图表示了车辆在0时刻的坐标关系图：



其中 ΔL 表示为每个小分割的步长， $\Delta L = V_x * \Delta t$ ， $\Delta \phi$ 表示在一定时间 t 内，车辆过弯旋转的角度， $\Delta \phi = W_t * \Delta t$ 。以上所有值均可测量， V_x 和 W_s 可以直接通过底盘ESP传感器获取。

其中，相应行驶的角度可以表示为

$$\theta = \frac{\pi - \Delta\varphi}{2}$$

以此类推，车辆在第t个坐标关系为如下：

$$\begin{pmatrix} y_t \\ x_t \end{pmatrix} = \begin{pmatrix} \Delta L \cos \theta \\ \Delta L \sin \theta \end{pmatrix} = v\Delta t * \begin{pmatrix} \cos((\pi - w\Delta t) / 2) \\ \sin((\pi - w\Delta t) / 2) \end{pmatrix} = v\Delta t * \begin{pmatrix} \sin(\frac{w_t\Delta t}{2}) \\ \cos(\frac{w_t\Delta t}{2}) \end{pmatrix}$$

再由 $\Delta L = V_x * \Delta t$ 及在第t个坐标系的关系可类推车辆在t+1个坐标系的关系如下：

$$\begin{pmatrix} y_{t+1} \\ x_{t+1} \end{pmatrix} = v\Delta t * \begin{pmatrix} \sin(\frac{w_{t+1}\Delta t}{2}) \\ \cos(\frac{w_{t+1}\Delta t}{2}) \end{pmatrix} + \begin{pmatrix} y_t \\ x_t \end{pmatrix}$$

如上推论可利用自车的速度和横摆角速度计算不同时刻自车在基础坐标系中的位置，以此得到自车行驶轨迹。然后利用雷达数据计算前车相对于自车的位置关系，得到前车行驶轨迹。

相应的自车及前车参数输入要求如下：

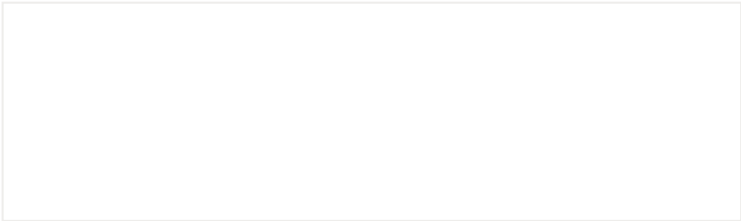
参数	参数名称	参数来源
自车参数	车速 V_{ego}	ESP
	横摆角速度 W_s	ESP
	转向角速度 θ	EPS/SAS
	横向加速度 a_{ego}	ESP
前车参数	前车速度 V_{front}	Radar
	前车距离 L	Radar
	前车横向加速度 a_{front}	Radar
	前车与本车相对角度 Φ	Radar/Camera

道路实际情况预估

对于前车目标的运动实际情况需要综合考虑车道实际弯曲度变化情况，比如在弯道内换道，自车探测到前车的横向加速度时，实际是要减去因道路弯曲而产生的部分过弯横向加速度，此时才是真正的变道横向加速度值。

本文所指的道路实际情况预估也即道路曲率估算（暂不考虑因路面不平、减速带及噪声影响），自车行驶的当前道路曲率可以通过自车数据预估道路当前曲率及实际路面情况。假设在划分的当前一小段时间之内其速度和横摆率满足线性关系时，直接通过如下公式计算得出： $K_s = V_x / W_s$

前车轨迹曲率参照采集到的一系列车辆行驶轨迹中的离散点及通过车辆轨迹中预估的点进行拟合得到的实际曲率，根据以上表示的坐标点关系，相应的横纵向关系可以表示为 $x=f(y)$ ，对于某一个固定轨迹点的曲率方程表示如下：

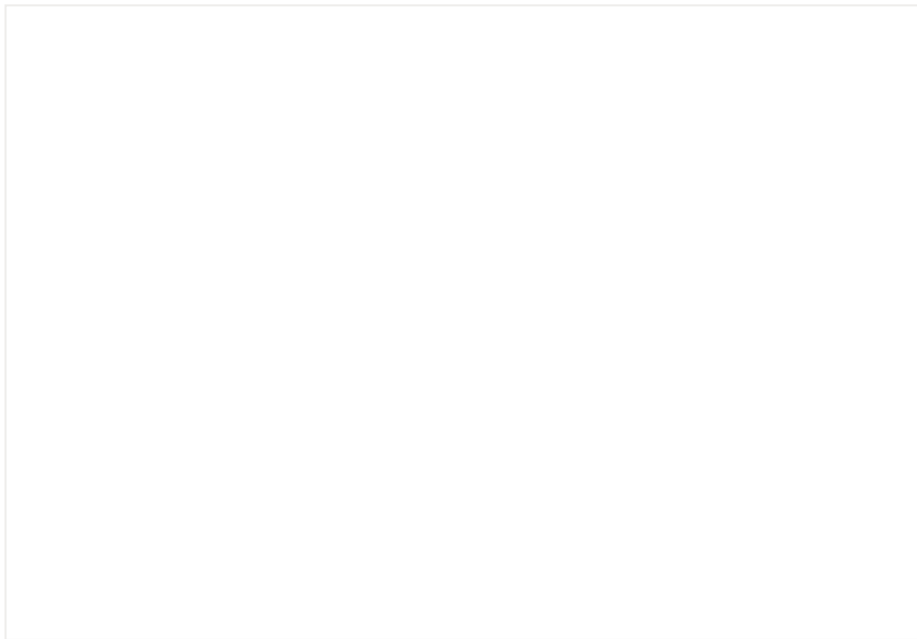


一般地，在我们划分的极小的两段时刻中的两段曲率不会发生突变，故可以利用当前时刻的实际计算的曲率通过一定的预瞄算法估算出下一段曲率。也即



前车出弯与换道模型识别

本算法重点在与区分前车相对于本车的运动状态，包括前车出弯或是换道，单纯从传感器层面是无法区分出前车换道或是出弯状态的，因为先前即有文章分析过，单雷达探测前车的原理即是通过探测前车与本车的重叠量是否大于某个阈值来判定前方目标是否还作为本车的跟踪目标，出弯和换道均满足该条件，故是没有办法区分两种情况的。本算法通过接受传感器探测的前车数据（包括方位角、横向加速度、距离、车速等）元素，按照一定的算法为前车预估一条轨迹曲线，在该曲线下判断前车当前的行为，从而区分换道和出弯。



一般的本车和前车在弯道内的行驶情况可分为如下几种情况：

1、 自车处于直道，前车入弯，如上图（a）；

此时本车位置处弯道曲率变化不大，若前车处于本车道内入弯的状态，则估算的前车行驶轨迹斜率K和由小逐渐变大，且该斜率变化规律为一致性的。若前车处于换道过程时，则估算的前车轨迹斜率先逐渐变大在变小；

2、 自车处于弯道，前车也处于弯道，如上图（b）；

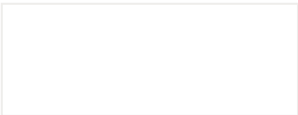
此时，本车位置处弯道曲率变化为一定值，可通过对本车斜率进行预估得到相应的横纵向坐标轨迹点变化规律，此时前车也处于弯道中时，我们需要估算本车与前车在一定位置处的斜率差，当该斜率差由小逐渐变大，且该斜率变化规律为一致性，则判断为前车在本弯道内消失于本车探测视野，当该斜率差先逐渐变大在变小时，则判断为前车处于换道过程中；

3、 自车处于弯道，前车出弯，如上图（c）；

此时本车位置处弯道曲率变化较大，若前车处于本车道内出弯的状态，则估算的前车行驶轨迹斜率K变化不大。若前车处于换道过程时，则估算的前车轨迹斜率先逐渐变大在变小；

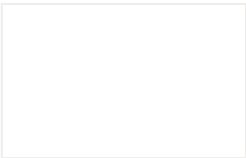
以上表示的前车行驶轨迹斜率变化表示为如下：

取极小的一段时间内的行驶轨迹对应的斜率为



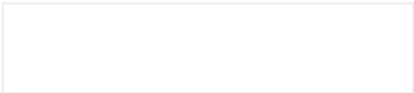
其中Y表示横向偏移，X表示纵向偏移。

则通过如下的方程式可以表示前车轨迹的斜率变化率为



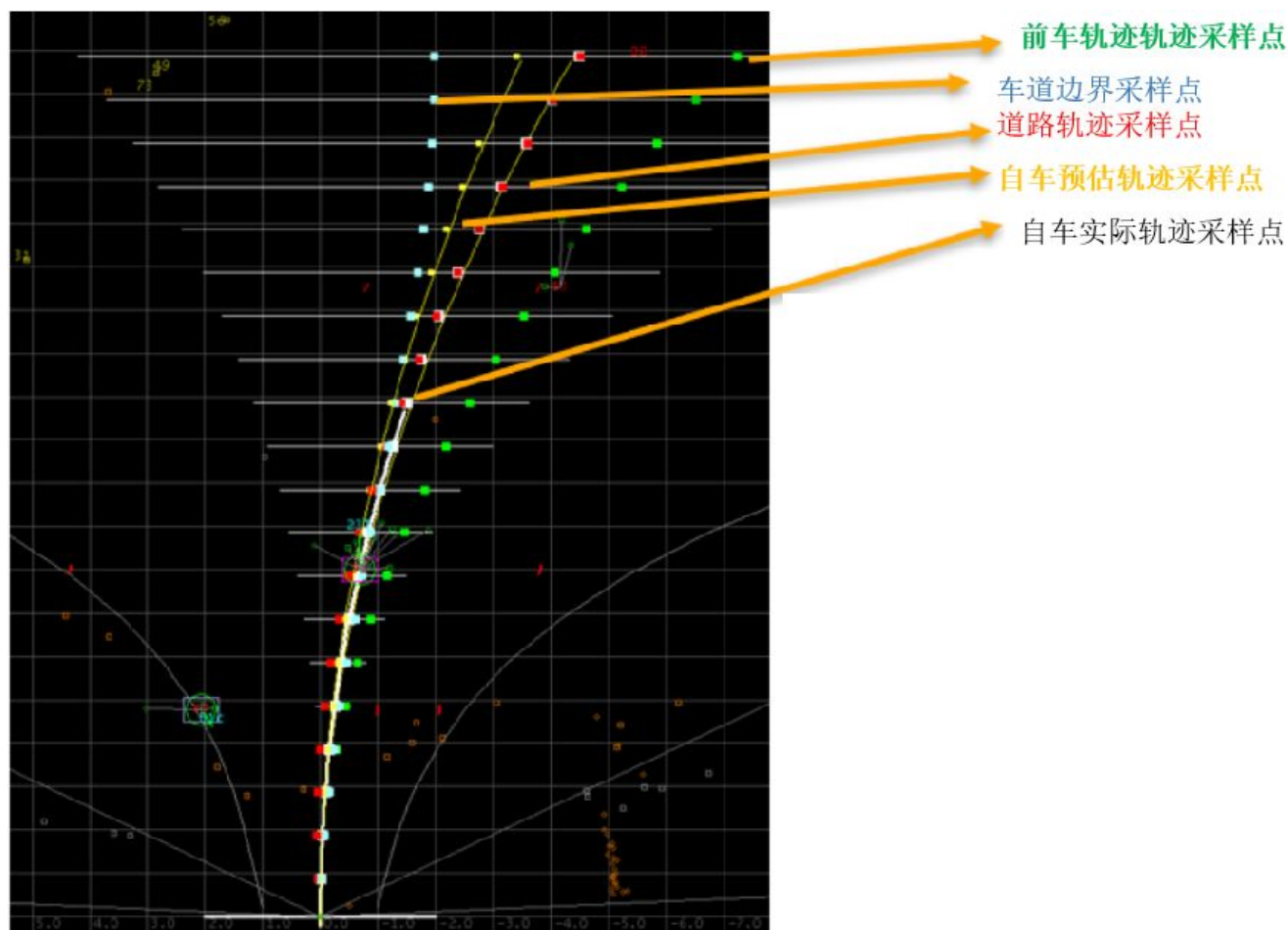
要求

时满足判断要求。



如下图表示了在实际试验过程中采集到本车直道，前车入弯的一组本车轨迹点，与传感器探测到的前车轨迹点，并通过一定的算法拟合出的轨迹曲线图。





总结

本文详细描述了如何在传感器受限情况下提升弯道内对于前车目标探测的精确度，当前对于弯道这一极限工况下的目标探测仍然存在很多性能提升方向，弯道目标行驶状态的精准探测对于弯道决策控制起着决定性的作用，当弯道目标错误探测后，比如前车出弯，本车未能准确判断，可能出现错误的加速，甚至与出弯的前车有碰撞风险，另外如果前车在弯道内换道，本车错误判断为出弯，此时仍然保持为本车一定的速度不增加，则可能无法按照正常的自适应巡航方式进行加速，也容易引起用户抱怨，故寻找两者的最优探测方案正是本文要说明的问题。

当然，后续随着传感器的配置不断增加，探测能力不断升级，可以结合对于环境周边物体，如车道线、护栏等信息，并利用车辆网V2X、5G技术等可以更加完美的实现目标行驶状态探测功能。