Coagent 动态轨迹设计说明书

修订记录

版本	修改记录	作者	时间
V1.0.0	第一次发布	mjzhang	2017-02-16

目录

1. 引言

1.1 编写目的

本文档旨在 说明动态轨迹 技术的设计原理、数据模型解析、以及软件转换思路。

1.2 背景说明

倒车轨迹是近两年部分汽车导航设备上新出现的一个功能，其借助方向盘转角信息将汽车可能的后退路线叠加到后视摄像头的输出上并标注出距离，以直观形象化的形式协助驾驶人员调整选择倒车路线，减少驾驶人员特别是新手的误判断，对使用者是一个不错的实用功能。倒车轨迹在智能倒车领域内属于辅助倒车系统中的一种，虽然其还无法达到智能化倒车，但是其实用性和辅助性上对汽车智能化单元技术方面是一个有效的补充。

当前我司倒车轨迹技术的实现依赖于外部供应商，一般采用集成轨迹的摄像头，或者将摄像头输入轨迹盒子进行转换，生成倒车轨迹信号。外部方案存在以下问题：

1、成本高，采购成本高

2、集成周期长，需要供应商配合调试比较麻烦

3、适应能力差，难以根据车辆差异进行校准

因此有必要自主研发倒车轨迹技术。我司将此技术命名为动态轨迹。

2. 技术原理

2.1 概念说明：

如图 1

1、纵轴距 Dwb —— 前轴中心与后轴中心的距离

2、转向轴距 Dsa —— 前轮转动时，左轮偏转的垂直轴线 与 右轮偏转的垂直轴线 的距离。这二条轴线位于前轮轴线上，对应轮的内侧。

3、内轮角 Ai —— 内侧轮胎转向角

4、外轮角 Ao —— 外侧轮胎转向角

5、舵角 Aw —— 方向盘转角

6、轨迹圆心 O(x,y) —— 内轮角线 外轮角线 与后轴线 的相交点，该 3 条线一定会相交于一点，这是由汽车厂设计 与 转向调教需要保证的，否则转向过程中轮胎就会侧滑拖拽导致磨损。

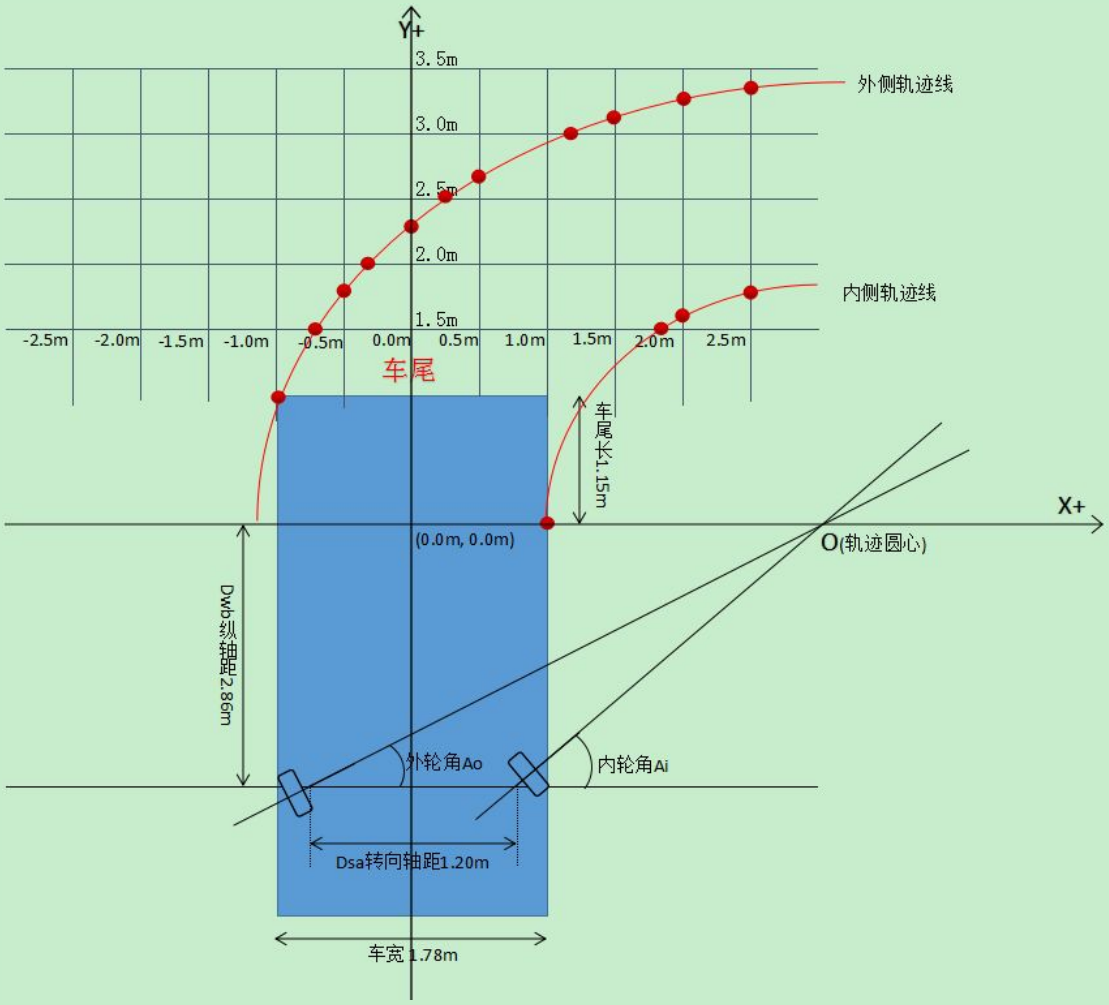


图 1 实际坐标系下的轨迹模型

2.2 实际坐标下的模型分析

首先我们建立实际坐标系的轨迹模型，如图 1。根据倒车摄像头的特性，从车头往车尾俯视汽车，车尾延伸方向为 Y+，后轮的轴线为 X 轴。坐标绘制范围(x:-2.5m ~ 2.5m, y:1.0m ~ 3.5m)基本可以覆盖倒车安全范围。故我们的目标就是得到内外两条轨迹线与坐标网格的相交点，并据此绘制曲线到倒车视频上。

2.2.1 获得轮角

如图 1，汽车设计完成后，它的内轮角 Ai,外轮角 Ao 与方向盘舵角 Aw 是一个由转向臂决定的固定对应关系。但并不是一个固定比值，这个关系表可由汽车设计单位给出。如下图 2 表所示。

方向盘舵角	轮胎角度 （ 外）	轮胎角度（ 内）
0	0	0
10	0.51	0.69
20	1.11	1.29
30	1.71	1.89
40	2.31	2.48
50	2.92	3.08
60	3.52	3.67
70	4.13	4.26
80	4.74	4.86
90	5.35	5.45

图 2 舵角 与 轮角对应表

2.2.2 计算转弯圆心

如图 1，汽车设计时，为了避免转弯过程中车轮拖拽现象发生，转弯过程中会让四轮以后轮轴线上的一个点 O(x,y)为圆心绕行。当然会有误差，在此我们暂不计，仅以理想状态计算。

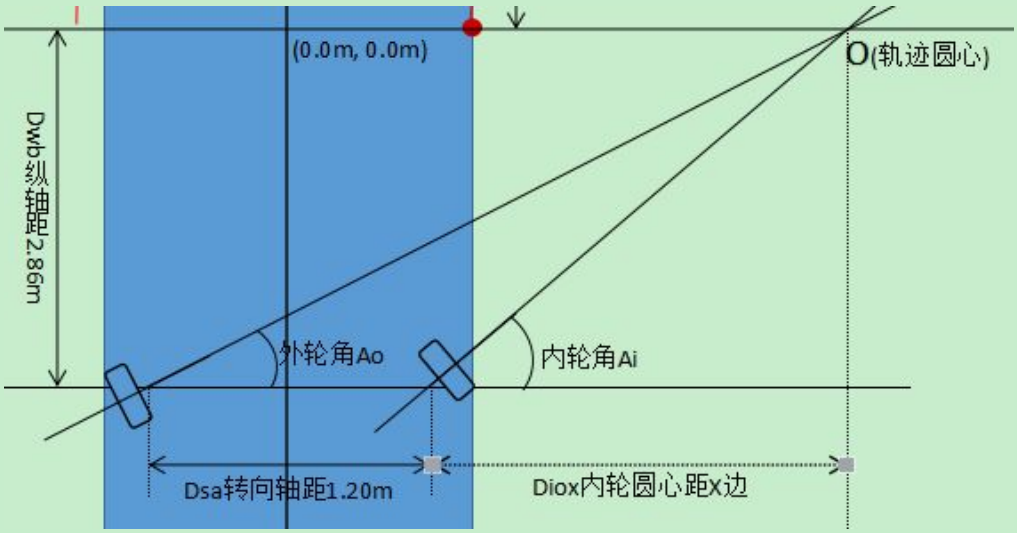


图 3 圆心计算示意图

O(x,y)在 X 轴线上，故 Oy = 0.0m 。

如图 3，$Oy = \frac{Dsa}{2} + Diox$。而 $Diox = \frac{Dwb}{\tan(Ai)}$，因此 $Oy = \frac{Dsa}{2} + \frac{Dwb}{\tan(Ai)}$。

2.2.3 轨迹方程 与 轨迹点求解

在计算轨迹之前，须首先确定使用汽车的哪个位置来计算轨迹合适。在此我们以倒车时车尾最远点 Pf 和最近点 Pn 来计算。示意图中我们是左转，因此这里最远点坐标为(-车宽/2, 车尾长)，从图 1 可知为 (-0.89m, 1.15m)。最近点(车宽/2, 0),即(0.89m, 0.0m)。

已知圆心 和 圆上一点，即可推算圆周方程。

先计算外侧轨迹圆半径 $Ro=\sqrt{(Pfx-Ox)^2+(Pfy-Oy)^2}$，故而外侧轨迹圆方程为 $(x-Ox)^2+(y-Oy)^2=Ro^2$。

同理内侧轨迹圆半径 $Ri=\sqrt{(Pnx-Ox)^2+(Pny-Oy)^2}$，内侧轨迹圆方程为 $(x-Ox)^2+(y-Oy)^2=Ri^2$。

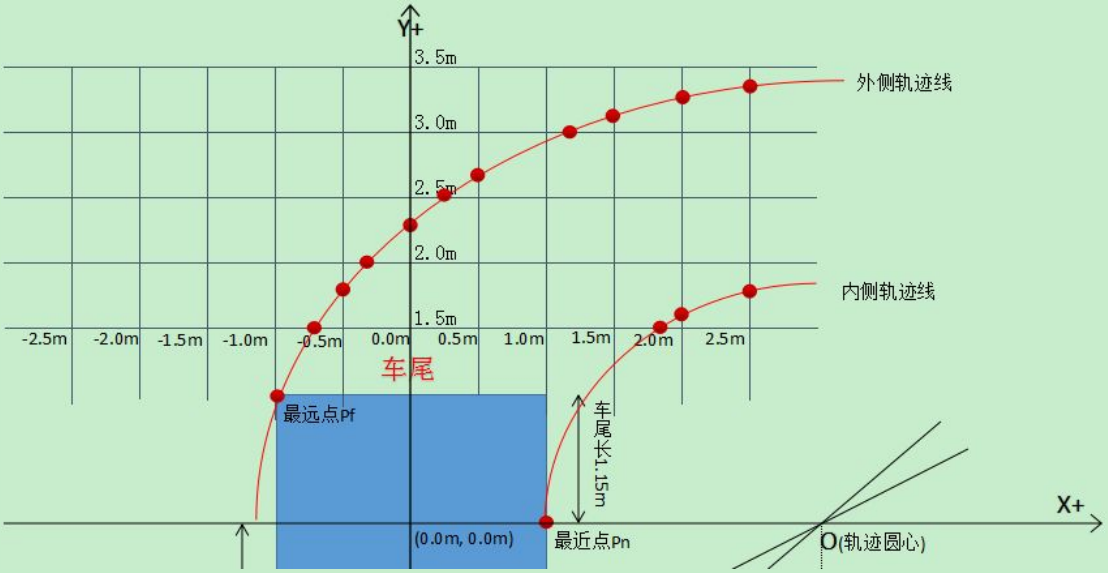


图 4 最远点 与 最近点 的轨迹

接下来只要将圆与坐标网格的相交点计算成一个坐标数组 Physics[]即可。

这里计算圆与直线的相交点需要解一元二次方程，可将方程转换后，根据下面这个模型进行求解。

方程 $ax^2+bx+c=0(a\neq 0)$ 的解为 $x=\dfrac{-b\pm\sqrt{b^2-4ac}}{2a}$

在求解方程之前，应先判断圆与线是否相交，若不相交，上述方程无解(解为虚数)。

2.3 像素坐标系捕获

2.3.1 如何纠正摄像头畸变

前面经过一系列计算，得到了轨迹点数据，但是实际坐标系下的点是无法显示到屏幕中的。因为实际坐标系经过摄像头捕获到屏幕上会产生畸变，就不在是简单的比例关系了。必须考虑摄像头的畸变因素。那么就需要一个与实际坐标系对应的像素坐标系，而畸变因素因各家摄像头制造的不同，变化多端难以统计。

如果可以找到一种方法来自动纠正这种畸变，那就消除了这种导致误差的最大的因素。在此我们做一个猜想，假如我们程序内保存了所有的像素点对应的地面坐标，那么把我们前面计算得到的实际坐标系相交点 Physics[]，通过查表得到对应的像素坐标点 Pixel[]，将 Pixel[]连贯绘制，那不就是我们想要在屏幕上呈现的轨迹线吗。

2.3.2 如何捕获像素坐标系

那下一步就是如何得到这个 Physics 与 Pixel 的对应关系表。我们可以在地上将所有点的坐标标记出来，然后通过摄像头抓取图片，在图上捕获标记出来的实际坐标点对应的像素坐标点。然而，我们不可能所有实际坐标点，都捕获对应的像素坐标点。一张 800x480 的图片就有 38.4 万像素，要做完，那就呵呵了。更何况未来屏幕分辨率越来越高。

通过实验，我们发现只要实际坐标系每隔 0.5m 捕获一个点，则每 2 个点之间的线段畸变曲率就已经可以忽略了。如图 5。鉴于我们倒车时，对于倒车正前方的障碍物是较敏感的，所以不必捕获太远，到后车轴 3.5m 的距离足够。而侧面存在盲区，因此左右各捕获 2.5m，即-2.5m ~ 2.5m。因此我们需要捕获一个 11 x 5 的矩阵。如图 6。实际左右边缘有些点是不可能到达的，可酌情忽略。

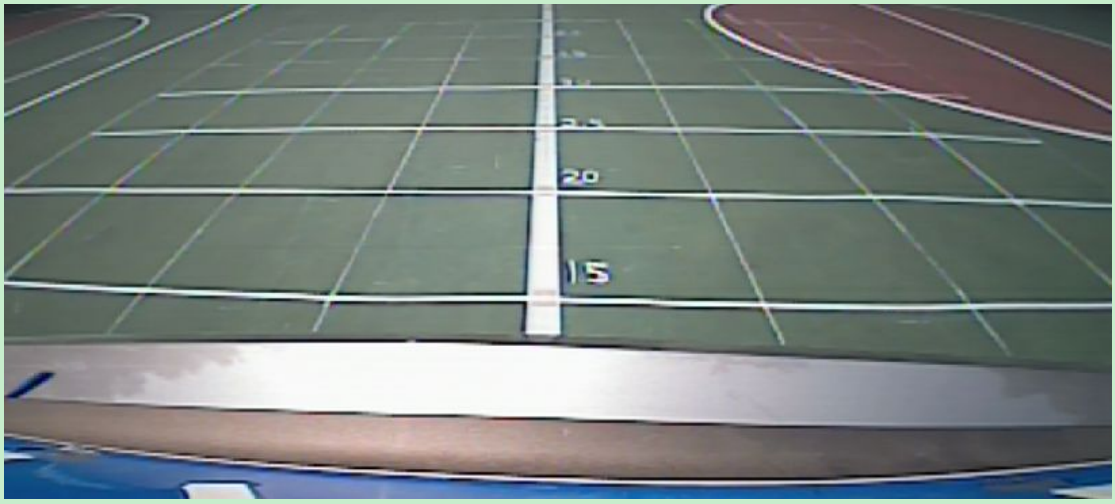


图 5 地面实际坐标系通过摄像头在屏幕上的呈现

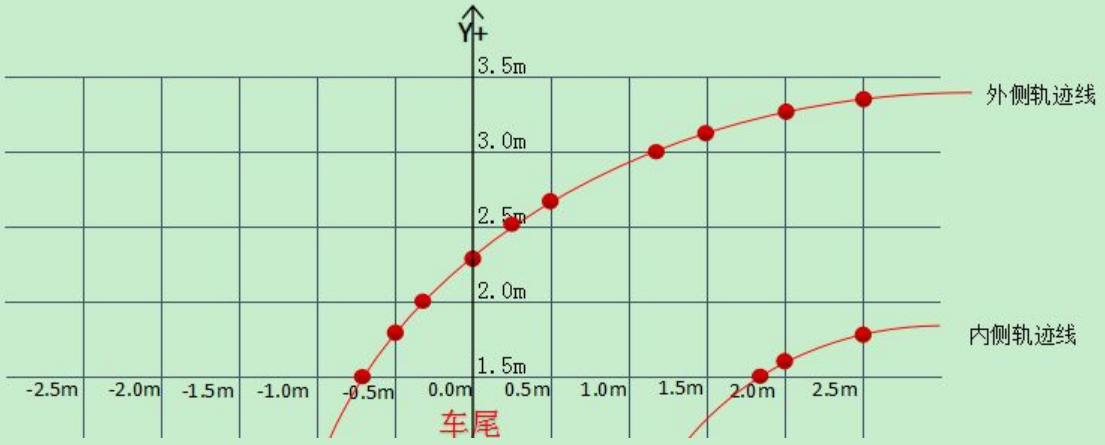


图 6 实际坐标系中的捕获矩阵

另外，需要再捕获几个车尾边缘的点对车库线进行修正，故前述 11 x 5 的矩阵实际我们改为 11 x 6 的矩阵。如图 7。

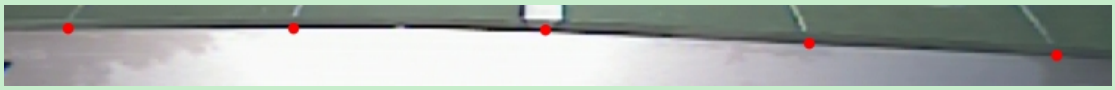


图 7 车尾边缘 5 个车库线修正点

到此我们就得到了一个完整的像素坐标系表 CoorPixel。

2.4 将实际坐标系的轨迹相交点 转换成 像素坐标系中的点

2.4.1 转换像素坐标点

前面我们已经得到实际坐标系中的轨迹相交点 Physics 和 像素坐标系 CoorPixel，接下来只要将 Physics 根据 CoorPixel 转化为 Pixel 即可进行绘制了。

因为按 0.5m 一段来捕获坐标系时，坐标系的每一段都可近似为直线段。所以我们可以根据 Physics 点落在哪一条线段上，来近似计算它对应的 Pixel 点。这只要查表后，按等比例折算即可，不再赘述。

2.5 绘制

2.5.1 绘制曲线

由于实际坐标系中的线通过摄像头后都变成了曲线，我们这里要先实现绘制曲线的方法，这里可以根据贝塞尔曲线函数来绘制每 2 个点之间的线段。要注意以下几点：

(1)、因为要保证这些线段连起来后是一条连贯(连续可导)的曲线，而不是折线。因此前一段曲线的结束和后一段曲线的开始必须保持一次导(即切线的斜率)相等。这可以通过关联 2 段曲线的控制点的方式来实现。

(2)、要注意 3 段曲线的曲率若非单调变化时，中间的曲线就无法通过一个控制点来保持曲线的连续可导。此时可以将中间的曲线分为 2 段，并且 2 段曲线的控制点方向相反的方式来保持曲线的连续可导。

2.5.2 绘制车库线

车库宽度固定为 2.0m，故将 CoorPixel 中的点按照规则绘制出来即可。

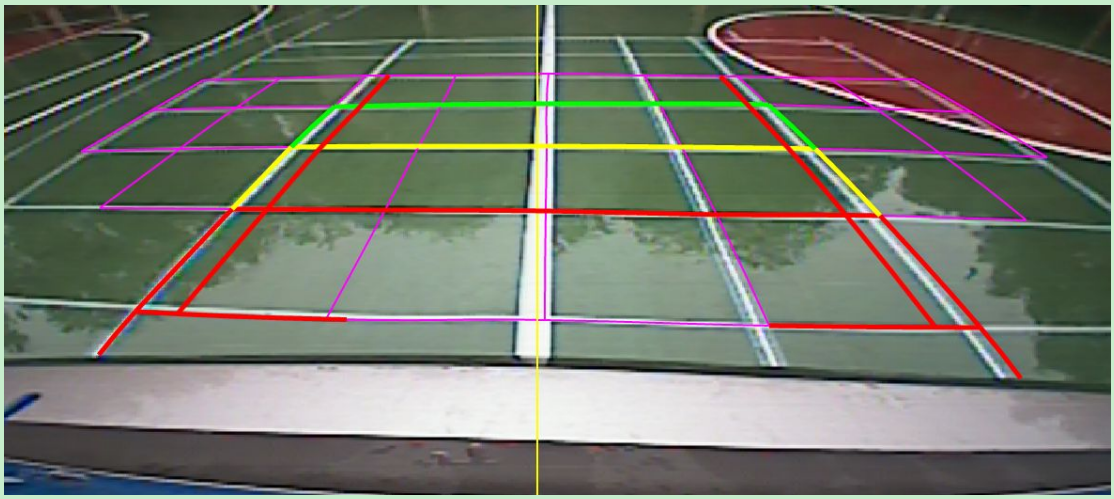
2.5.3 绘制像素坐标系

将捕获的 CoorPixel 绘制为可见的网格，以便核验校准效果。

2.5.4 绘制轨迹

将前面计算并转换好的 Pixel 绘制为曲线，并根据输入角度变化。

会之后的效果图如图 8。



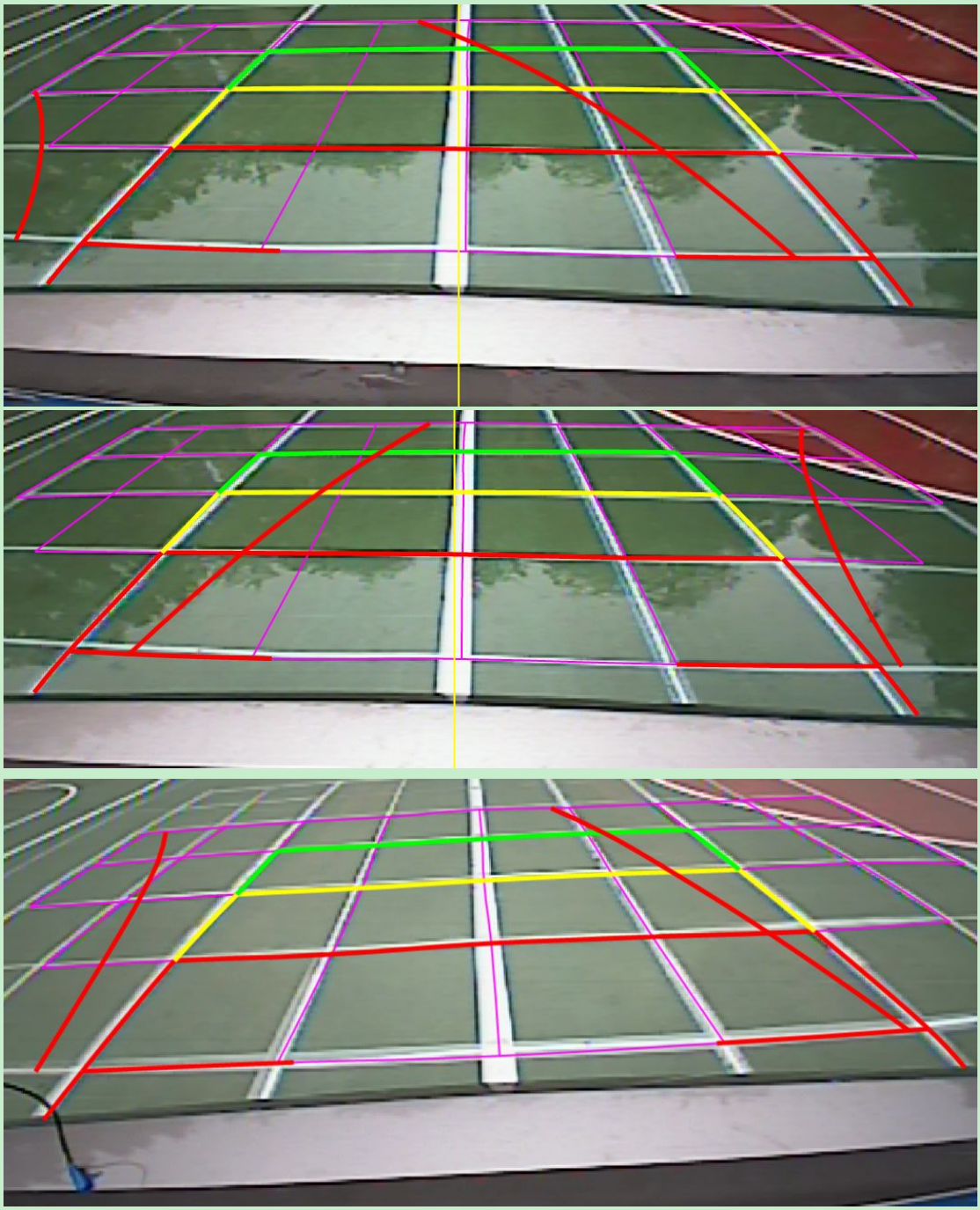


图 8 绘制车库 轨迹 坐标网格 效果图

2.6 校准轨迹

校准轨迹，即是校准像素坐标系，根据前面 2.3 所述原理，设计界面供客户操作。

如图 9。支持像素坐标系 11 x 6 的校准，同时还支持车身参数手动修正。这个设计可以让用户很方便简单的调整相关参数。

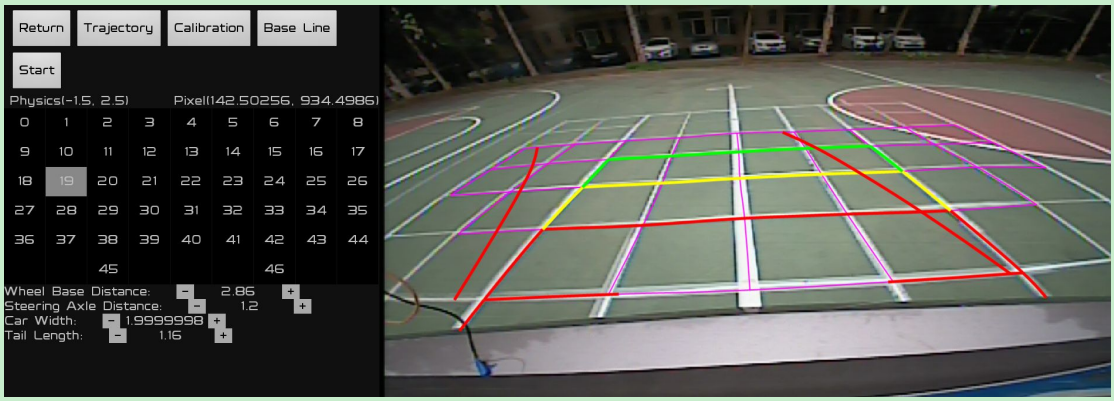


图 9 轨迹校准基本界面

3. 验证结果

经过初步的实车验证，误差范围、校准效果可以很好地满足当前客户需求，并且支持校准。比之前外购的方案有更好的体验，又节省成本。目前在 Geely 项目已投入使用，众泰项目也计划采用该技术。之前所有外购方案，一个方案就要 8 万以上的成本。若替换 10 个项目的话，每年可为公司节省 80 万以上的成本。