

障碍物横向跳动/漂移优化

策略

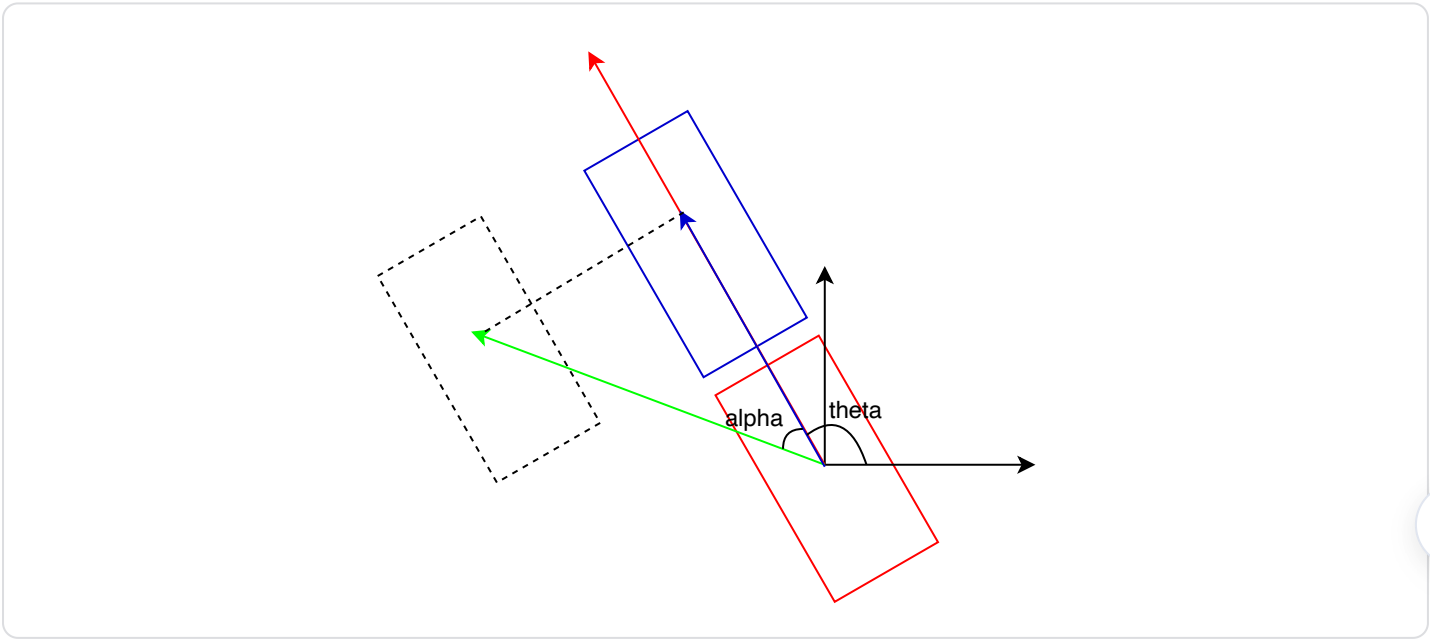
障碍物运动方向应与朝向保持一致，若不一致则会出现位置漂移，跳动等不合理的情况。障碍物速度是由位置变化推算出来的，感知障碍物位置不准确的跳变会产生一个错误的速度（方向），因此障碍物的位置不能完全由速度（方向）来决定。

朝向过于平滑时不符合预期!!! 需结合障碍物朝向与障碍物速度方向!

核心思路

取障碍物在朝向方向（投影）的位置变化更新障碍物的位置，即只考虑朝向方向上的位移。

示意图



说明：

- 红色框为障碍物上帧位置，黑色虚线框为障碍物当前帧观测（位置发生漂移），蓝色框为修正位置后障碍物的最新位置。
- 红色箭头为障碍物平均朝向向量（图中所示假设朝向障碍物前后不发生变化）（实际是归一化向量，这里方便展示特意画长），绿色箭头为障碍物位移向量，蓝色箭头为优化后最终的位移向量。
- alpha为位移向量与平均朝向向量的夹角，theta为障碍物平均朝向角

公式：

平均朝向向量: $\overrightarrow{dir} = 0.5 * (\overrightarrow{dir_0} + \overrightarrow{dir_1})$ 其中, $\overrightarrow{dir_0}$ 为上一帧朝向, $\overrightarrow{dir_1}$ 为当前帧观测朝向

位移向量: $\overrightarrow{P} = (\overrightarrow{P_1} - \overrightarrow{P_0})$ 其中, $\overrightarrow{P_0}$ 为上一帧位置, $\overrightarrow{P_1}$ 为当前帧观测位置

位移向量在平均朝向上投影长度: $ProjectNorm = \overrightarrow{P} * \cos(\alpha)$

最终位移向量: $\overrightarrow{CorrectP} = (ProjectNorm * \overrightarrow{dir}[0], ProjectNorm * \overrightarrow{dir}[1], 0)$ 其中,
 $\overrightarrow{dir}[0] = \cos(\theta)$, $\overrightarrow{dir}[1] = \sin(\theta)$

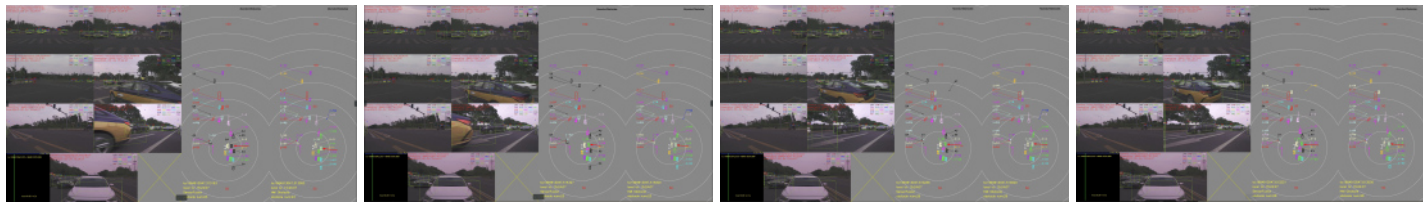
最终位置: $\overrightarrow{Pout} = \overrightarrow{P_0} + \overrightarrow{CorrectP}$

case回归效果

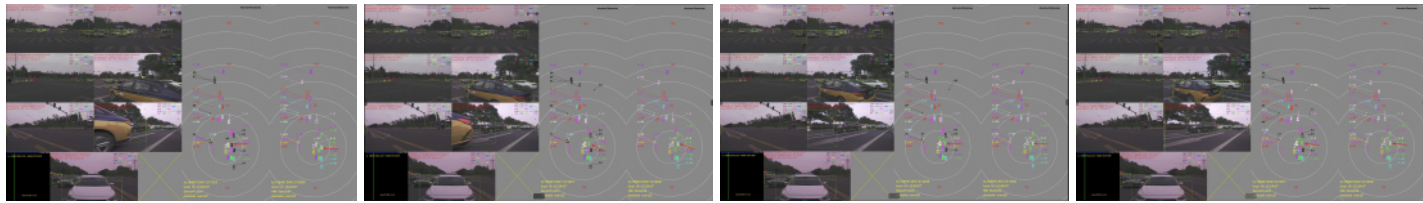
数据：cyberecord_ARCF506_default_0_20230524152402_20230524152430

case1

线上结果：



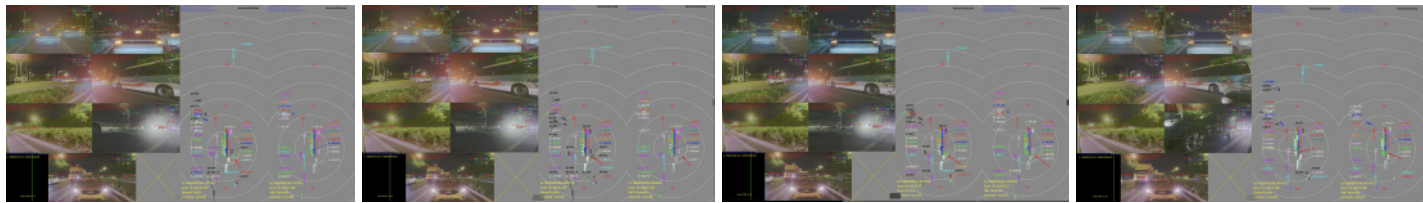
优化效果：



数据：https://yf.baidu-int.com/adp/datafinder/taskdetailinfo?taskid=ARCF011_20230606195148&namespace=apollo_ap%2Fauto_car

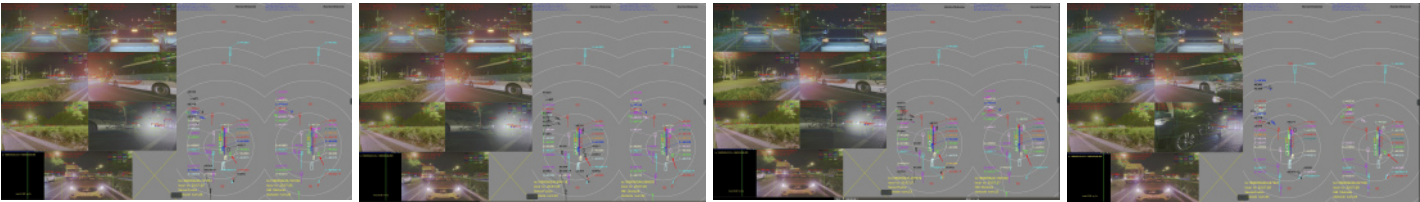
case2

线上结果：



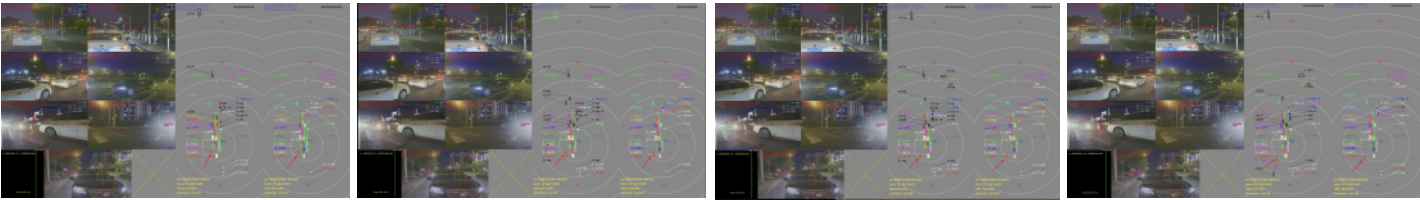
优化效果：



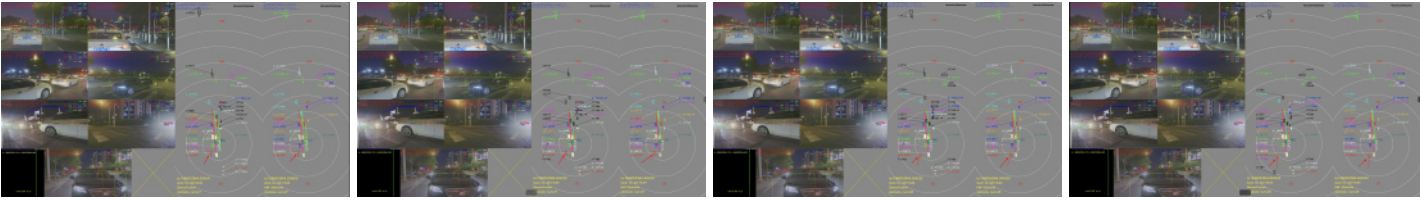


case3

线上结果：

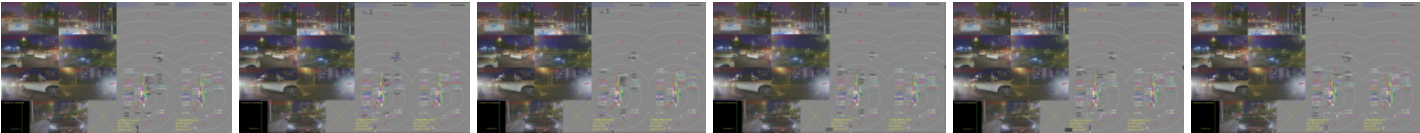


优化效果：



case4

线上结果：



优化效果：

