


车道级数量滤波

目录

- 背景：
- 现有车道生成方案
- 优化方案：
- 车道几何构建

背景：

当前问题：因模型远端车道线检测不稳定、车道宽度在阈值附近跳动引起补线策略间歇性失效等问题出现Segment帧间车道线数量变化，对车道拓扑、属性投影模块产生影响，从而引发实车画龙问题。  [车道数量不稳定聚类](#)

优化方案：接入LD信息作为先验，结合模型观测，通过滤波方式稳定segment车道数量。

现有车道生成方案

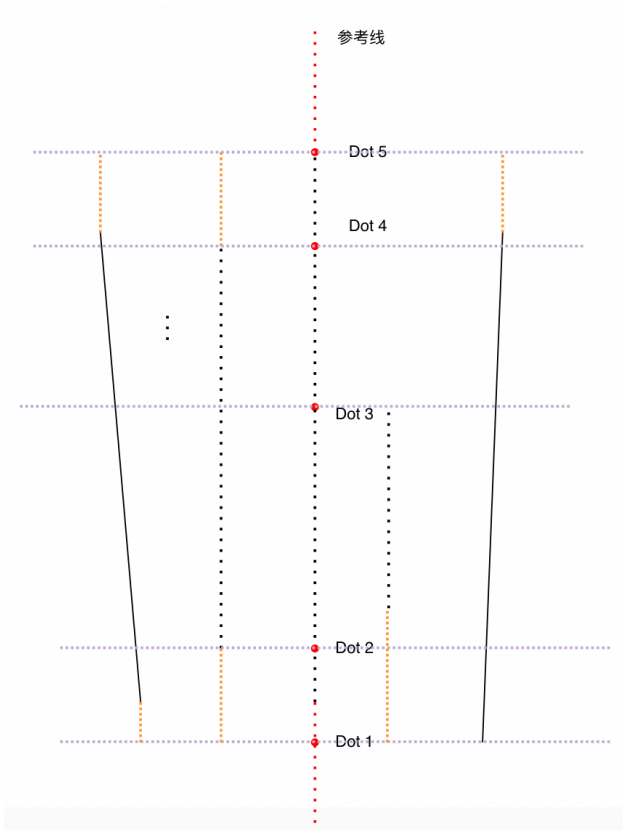
参考文档  [元素（Segment/Lane/Line）生成逻辑梳理](#)

车道生成：

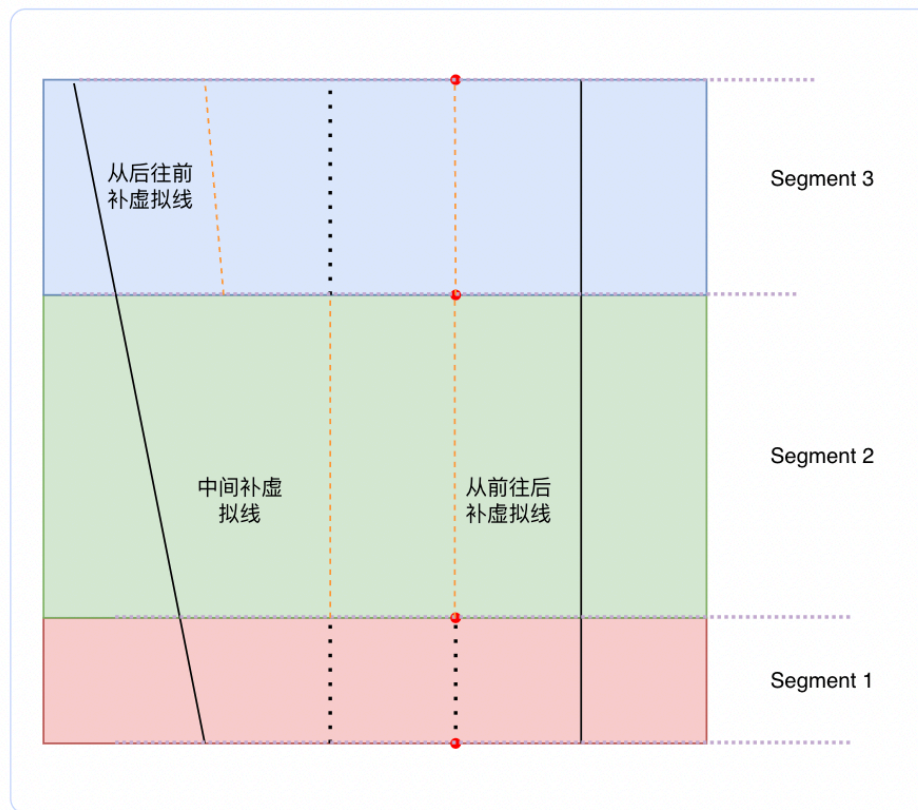
- 要点：segment之间进行车道线延长和补线策略，segment内主要查找合适的左右车道线生成策略

https://console.cloud.baidu-int.com/devops/icode/repos/baidu/adu-lab/andes/blob/master/lib/ultra_map_fusion/lib/topo_mapping/generator/segment_generator.cc#L1015

https://console.cloud.baidu-int.com/devops/icode/repos/baidu/adu-lab/andes/blob/master/lib/ultra_map_fusion/lib/topo_mapping/generator/lane_generator.cc#L31



延长策略示意图



布线策略示意图

车道关联：

- 先找相同id（应该没奇效），再基于贪心找iou最大

https://console.cloud.baidu-int.com/devops/icode/repos/baidu/adu-lab/andes/blob/master/lib/ultra_map_fusion/lib/topo_mapping/association/lane_matcher.cc#L39

车道更新：

- 未匹配观测生成新的tracker，匹配的观测target直接更新tracker

https://console.cloud.baidu-int.com/devops/icode/repos/baidu/adu-lab/andes/blob/master/lib/ultra_map_fusion/lib/topo_mapping/base/lane_track_target.cc#L63

优化方案：

在车道更新时，增加数量滤波对车道数量进行帧间平滑。因车道数量是一种离散的观测量，考虑使用HMM或者非线性滤波器。

状态空间

定义 S 为车道数量，假设最多有7条车道 $S = \{1, 2, 3, 4, 5, 6, 7\}$

状态变量

每种车道数量的概率 π ，初始值 π_0 可参考LD中车道数量通过离散高斯模型进行概率计算

状态转移概率矩阵

假设在segment宽度稳定且和LD宽度接近情况下，车道数量只存在三种变化可能： P 不变、 $0.5 - P/2$ 增加一条、 $0.5 - P/2$ 减少一条。假设宽度多帧稳定，不变的概率最大。

另外可参考LD前后segment信息判断是否存在分流场景，进行状态转移矩阵设计

不存在分流： $A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$

存在分流： $A = \begin{pmatrix} 0.9 & 0.1 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0.8 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0.8 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.8 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0.1 & 0.8 & 0.1 \\ 0 & 0 & 0 & 0 & 0 & 0.1 & 0.8 \\ 0 & 0 & 0 & 0 & 0 & 0.1 & 0.9 \end{pmatrix}$

观测概率矩阵

考虑三个方面可观测segment车道数量：LD先验车道数量、基于segment宽度推理车道数量、基于模型观测车道数量

$$a_{ij} = \alpha a_{ij}^{LD} + \beta a_{ij}^{Width} + \gamma a_{ij}^{Perception}$$

此外可以添加距离因子，距离越远高斯峰值越低

近距离时：
$$B = \begin{pmatrix} 0.8 & 0.2 & 0 & 0 & 0 & 0 & 0 \\ 0.15 & 0.7 & 0.15 & 0 & 0 & 0 & 0 \\ 0 & 0.15 & 0.7 & 0.15 & 0 & 0 & 0 \\ 0 & 0 & 0.15 & 0.7 & 0.15 & 0 & 0 \\ 0 & 0 & 0 & 0.15 & 0.7 & 0.15 & 0 \\ 0 & 0 & 0 & 0 & 0.15 & 0.7 & 0.15 \\ 0 & 0 & 0 & 0 & 0.2 & 0 & 8 \end{pmatrix}$$

远距离时：
$$B = \begin{pmatrix} 0.6 & 0.2 & 0.2 & 0 & 0 & 0 & 0 \\ 0.2 & 0.5 & 0.2 & 0.1 & 0 & 0 & 0 \\ 0.05 & 0.2 & 0.5 & 0.2 & 0.05 & 0 & 0 \\ 0 & 0.05 & 0.2 & 0.5 & 0.2 & 0.05 & 0 \\ 0 & 0 & 0.05 & 0.2 & 0.5 & 0.2 & 0.05 \\ 0 & 0 & 0 & 0.1 & 0.2 & 0.5 & 0.2 \\ 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.6 \end{pmatrix}$$

观测状态

定义 O 为观测的车道数量，观测空间 $O = \{1, 2, 3, 4, 5, 6, 7\}$

前向算法 (Forward Algorithm) 求解最优估计

对于 t 时刻，每个状态概率为： $\alpha_t(i) = \sum_j \alpha_{t-1}(j) a_{ji} b_i(o_t)$ ，最大概率对应的车道数量为 $P_t = argmax(\alpha_t(i))$

车道几何构建

1. 滤波车道数 == 感知车道数，不处理几何
2. 滤波车道量 != 感知车道数 && 滤波车道量 == LD车道数
- 方案A：如果都是非分合流车道，且存在宽车道或者窄车道进行切分或合并；
- 方案B：作为车道线是否要进行延长和补线判断依据