

# Apollo&SVL联合仿真（三）建图与定位

## 1 NDT建图与定位

### 1.1 建立NDT定位地图

#### 1. 数据包录制

Note：为防止在数据采集过程中数据包过大，对图像数据进行屏蔽，具体录制cyber时可能用到的命令，可在终端输入 `cyber_record record -h` 进行查看

```
cyber_recorder record -a -i 600 -k
/apollo/sensor/camera/front_12mm/image/compressed
/apollo/sensor/camera/front_6mm/image/compressed
```

将数据包保存到 `/apollo/data/bag/lgsvl` 文件夹下。

#### 2. 建立NDT地图

拷贝 `scripts/msf_simple_map_creator.sh` 文件，重命名为 `ndt_simple_map_creator.sh`，并对文件做出以下修改：

```
## lin25 : 修改OUT_MAP_FOLDER
OUT_MAP_FOLDER="$4/ndt_map"

## 创建create_ndt_map()函数
function create_ndt_map() {
    /apollo/bazel-bin/modules/localization/ndt/map_creation/ndt_map_creator \
        --pcd_folders $1 \
        --pose_files $2 \
        --resolution_type single \
        --resolution 1 \
        --resolution_z 1 \
        --map_folder $OUT_MAP_FOLDER \
        --zone_id $ZONE_ID
}

## 将create_lossless_map "${DIR_NAME}/pcd"
"${DIR_NAME}/pcd/corrected_poses.txt"进行替换
# create_lossless_map "${DIR_NAME}/pcd" "${DIR_NAME}/pcd/corrected_poses.txt"
create_ndt_map "${DIR_NAME}/pcd" "${DIR_NAME}/pcd/corrected_poses.txt"

## 注释掉lossy_map
# create_lossy_map
```

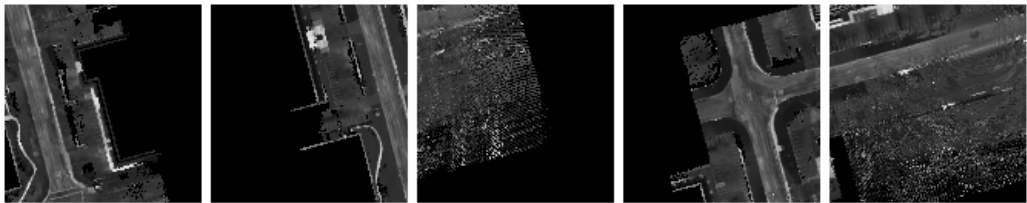
注意：resolution表示地图分辨率。对于ndt算法而言，并不需要过于精细的分辨率，一般而言，选择分辨率为1是一个相对比较好的选择。

执行建图程序，"Usage: `ndt_simple_map_creator.sh [records folder] [extrinsic_file] [zone_id] [map folder] [lidar_type]`"

```
bash ./scripts/ndt_simple_map_creator.sh /apollo/data/bag/sv1
/apollo/modules/calibration/data/Lincoln2017MKZ/velodyne_params/velodyne128_n
ovatel_extrinsics_example.yaml 10 /apollo/modules/map/data/sv1_map lidar128
```

在程序执行过程中，会依次遍历采集数据文件夹内的cyber包，先后对数据包进行**数据解析**、**位姿插值**、**地图生成**这三个步骤

- 解包处理，将数据包的点云数据，以pcd文件格式进行保存，将 `odometry` 以及 `localization_pose` 保存成txt文件，方便后续进行插值处理。
- 位姿插值，根据时间戳来进行插值来获取对应Lidar点云时刻的位姿。
- 地图创建成功后会在将地图保存在我们指定的 `map_folder` 中，点开查看 `image` 中的地图缩略图



### 3. 生成定位过程中的可视化地图

Note：在进行定位的可视化运行的是“MSF Visulizer”，需用通过 `./scripts/ndt_simple_map_creator.sh` 来生成所需要的地图，该地图同样也被MSF定位所使用

```
bash ./scripts/msf_simple_map_creator.sh /apollo/data/bag/sv1
/apollo/modules/calibration/data/Lincoln2017MKZ/velodyne_params/velodyne128_n
ovatel_extrinsics_example.yaml 10 /apollo/modules/map/data/sv1_map lidar128
```



## 1.2 运行NDT定位

1. 修改 `modules/localization/conf/localization.conf` 配置文件

```
# 指定地图位置
--map_dir=/apollo/modules/map/data/svl_map
--local_map_name=local_map
```

## 2. 启动NDT定位模块

```
cyber_launch start modules/localization/launch/ndt_localization.launch
```

## 3. 地图可视化

```
cyber_launch start modules/localization/launch/msf_visualizer.launch
```

