# 前言

本文主要内容为V1Pro相机定位算法模块使用示例说明,本示例基于ROS平台,经测试在Ubuntu16.04、Ubuntu18.04、Ubuntu20.04等Linux系统上编译运行,通过调用相机SDK实现建图(采图)、定位等功能。在查阅本文档前请先阅读相机SDK使用说明《LxCameraApi(C、C++)开发者指南.pdf》,同时,若运行于Linux平台请按照提示运行 install.sh 脚本,完成环境配置。

# 定位算法模块

#### 异常代码

```
1 // status
 2 // 算法返回值: 异常码
 3 Unknown = -1, //!< 重定位中
 4 Tracking
               = 0, //!< 正常定位中
 5 RelocFailed = 1, //!< 重定位失败
   slipping = 2, //!< 打滑
 7 Relocating = 6, //!< 重定位中
8 NoTopImage = 7, //!< 相机图像输入异常
9
   NoOdom
               = 8, //!< 里程计超时异常
              = 20, //!< 正常建图中, 建图需要odom,图像以及激光数据(激光数据or激光位
10 Mapping
    姿至少满足一个, 如无可发虚拟数据)
11 MappingFailed = 21, //!< 建图异常
           = 22, //!< 激光超时异常, 建图时使用
12 NoLaser
13 NoLaserPose = 23, //!< 激光位姿数据超时异常, 建图时使用
14
   // 附加返回值: 重定位异常码
15
16 -1: Unknown //! = 重定位中(若重定位位姿(x,y,yaw)全为0,会判断为误发,返回代
   码-1)
                    //!< 重定位成功
17 0: Normal

      18
      1: No image
      //!< 重定位失败, 图像输入异常</td>

      19
      2: No detection
      //!< 重定位失败, 特征检测异常</td>

20 3: No expectation //!< 重定位失败,期望地图异常
23 6: Off>30cm //!< 重定位失败,检测与地图无匹配 //!< 重定位失败,检测与地图无匹配 //!< 重定位失败,检测与地图无匹配 //!< 重定位生版 //!<
                //!< 重定位失败,特征检测、期望地图都异常
                     //!< 重定位失败, 检测与地图存在匹配, 距离超过阈值
24 7: invalid request //!< 重定位失败, 重定位请求位姿无效, 如NaN
```

### 定位置信度

在正常定位的状态(异常码为0时),可通过置信度衡量定位结果的质量,定位置信度分布于0~4,其中:

- 0: 不好
- 1: 较差
- 2-3: 一般
- 4: 较好

在正常定位的状态时,建议取置信度大于等于1时的定位结果。

### 建图

- 输入数据
  - 。 机器人里程计(绝对)的数据,类型为nav\_msgs/Odometry
  - o 机器人激光雷达的数据(辅助建图所需,部分场景可不用激光,可以不输入,请联系技术支持确认),类型为sensor\_msgs/LaserScan
  - 机器人激光位姿的数据(若需视觉地图和激光地图坐标系统一则需传入,否则可不输入),类型为geometry\_msgs/PoseStamped
- 输出数据
  - 导出相机中录制的原始数据,离线建图,然后导入视觉地图至相机

### 定位

- 输入数据
  - 。 机器人里程计(绝对)数据,类型为nav\_msgs/Odometry
- 输出数据
  - o 发送重定位请求,机器人重定位成功后,输出机器人在视觉地图中的位姿,类型为 geometry\_msgs/PoseStamped

# 编译

ROS节点创建步骤:

- 1. 进入示例程序代码文件目录 Lanxin-MRDVS/Sample/ROS/lx\_camera\_node\_ws
- 2. catkin\_make
- 3. source devel/setup.bash

## 运行

运行ROS示例前,若相机中已经部署ROS主从,需先关闭ROS主从相关设置,否则会因ROS产生相关通讯异常。若相机中有ROS环境,需上电后等待约20s左右,再运行ROS示例程序。

### 通讯

可以使用虚拟数据调通SDK通讯,确认与相机之间通讯正常。

- 1. source devel/setup.bash
- 2. roslaunch lx\_camera\_ros sensor\_sim.launch

如下图所示,该节点发出虚拟的激光数据 /sim/scan 、里程计数据 /sim/odom 和激光位姿数据 /sim/scan\_pose ,相机通过SDK去接收这些数据,联调通讯。

```
<!-- -135° -->
1
 2
      <param name="min_ang"</pre>
                                       type="double" value="-2.35619449" />
 3
      <!-- 135° -->
      <param name="max_ang"</pre>
                                       type="double" value="2.35619449" />
 4
      <param name="angle_increment" type="double" value="0.00582" />
 5
 6
      <param name="time_increment"</pre>
                                       type="double" value="0.00006167129" />
 7
      <param name="range_min"</pre>
                                       type="double" value="0.05" />
                                       type="double" value="101.0" />
      <param name="range_max"</pre>
 8
      <param name="init_range"</pre>
9
                                     type="double" value="100" />
      <!-- <remap from="scan" to="scan_back"/> -->
10
11
      <!-- 虚拟激光数据topic -->
12
      <param name="laser_frame_id"</pre>
                                            type="string" value="laser_link"
```

```
/>
14
    <!-- 虚拟里程计数据topic -->
                                   type="string" value="base" />
15
     <param name="odom_frame_id"</pre>
                                   type="string" value="/sim/odom"
16
     <param name="odom_topic_name"</pre>
17
     <!-- 虚拟激光位姿topic -->
18
     <param name="laserpose_frame_id"</pre>
                                       type="string" value="base"
19
     <param name="laserpose_topic_name"</pre>
                                       type="string"
    value="/sim/scan_pose" />
```

#### 3. roslaunch lx\_camera\_ros mapping.launch

如下图所示,该节点为接收虚拟传感器数据,然后通过调用相机SDK传入数据。

```
<launch>
1
2
       <node name="lx_localization_node" pkg="lx_camera_ros"</pre>
    type="lx_localization_node" output="screen">
 3
        <!-- 相机IP, 默认为空时按设备索引获取 -->
        <param name="ip"</pre>
4
                                                    type="string"
     value="192.168.100.82" />
        <!-- 图像显示使能 -->
 5
        <param name="is_show"</pre>
 6
                                                    type="bool"
     value="true" />
7
        <!-- 接收或发布ROS话题名 -->
8
        <param name="LxCamera_UploadScan"</pre>
                                                   type="string"
     value="/sim/scan" />
9
        <param name="LxCamera_UploadOdom"</pre>
                                                    type="string"
     value="/sim/odom" />
10
        <param name="LxCamera_UploadLaserPose"</pre>
                                                    type="string"
     value="/sim/scan_pose" />
11
        <param name="LxCamera_Error"</pre>
                                                    type="string"
     value="LxCamera_Error" />
12
        <param name="LxCamera_Command"</pre>
                                                    type="string"
     value="LxCamera_Command" />
13
        <param name="LxCamera_Mapping"</pre>
                                                    type="string"
     value="LxCamera_Mapping" />
14
        <param name="LxCamera_Location"</pre>
                                                    type="string"
     value="LxCamera_Location" />
15
        <param name="LxCamera_SetParam"</pre>
                                                    type="string"
     value="LxCamera_SetParam" />
16
        <param name="LxCamera_SwitchMap"</pre>
                                                    type="string"
     value="LxCamera_SwitchMap" />
17
        <param name="LxCamera_DownloadMap"</pre>
                                                    type="string"
     value="LxCamera_DownloadMap" />
18
        <param name="LxCamera_UploadMap"</pre>
                                                    type="string"
     value="LxCamera_UploadMap" />
19
        <param name="LxCamera_LocationResult"</pre>
                                                    type="string"
     value="LxCamera_LocationResult" />
20
        <param name="LxCamera_UploadReloc"</pre>
                                                    type="string"
     value="LxCamera_UploadReloc" />
21
        <!-- 自动曝光期望值, 范围[0-100], 默认:50 -->
                                                type="int"
22
        <param name="auto_exposure_value"</pre>
    value="50" />
23
        <!-- 建图使能, true or false -->
```

```
<param name="mapping_mode"</pre>
                                               type="bool"
24
     value="true" />
25
       <!-- 定位使能, true or false -->
        <param name="localization_mode"</pre>
26
                                              type="bool"
     value="false" />
       <!-- 重要: 定位时地图名需有效,已上传相机并存在;建图时,如果相机中无地图可默认输
27
    \lambda"example_map1" -->
28
        <param name="map_name"</pre>
                                      type="string"
    value="example_map1" />
        <!-- 相机至机器人外参,单位:米,度数,格式[x,y,z,yaw,pitch,roll] -->
29
        <rosparam param="camera_extrinsic_param"> [0.34, 0.00, 1.3, -90, 0,
30
    0] </rosparam>
        <!-- 激光雷达至机器人外参,单位:米,度数,格式[x,y,yaw] -->
31
32
        <rosparam param="laser_extrinsic_param"> [0.34, 0.11, 0.0]
    </resparam>
      </node>
33
34
35
   </launch>
```

4. 待算法返回异常码为20(Mapping)时,通讯即正常。

### 建图

- 1. 准备好建图所需的传感器数据
  - 。 机器人里程计(绝对)的topic,类型为nav\_msgs/Odometry
  - 机器人激光雷达的topic(部分场景可不用激光,请联系技术支持确认,此时可输入虚拟激光数据),类型为sensor\_msgs/LaserScan
  - 机器人激光位姿的topic(若需视觉地图和激光地图坐标系统一则需传入),类型为geometry\_msgs/PoseStamped
- 2. 按照要求修改launch文件夹 mapping.launch下文件(切记请核对),修改实际传入数据的topic名,如激光数据 /scan、里程计数据 /odom 和激光位姿数据 /scan\_pose。设置已存在地图名(若初次使用或不知道已存在地图名,可设置为默认 "example\_map1" )、相机外参、激光雷达外参等

```
1 < launch>
2
       <node name="lx_localization_node" pkg="lx_camera_ros"</pre>
    type="lx_localization_node" output="screen">
 3
        <!-- 相机IP, 默认为空时按设备索引获取 -->
4
        <param name="ip"</pre>
                                                     type="string"
     value="192.168.100.82" />
 5
        <!-- 图像显示使能 -->
 6
         <param name="is_show"</pre>
                                                     type="bool"
     value="true" />
 7
        <!-- 接收或发布ROS话题名 -->
 8
         <param name="LxCamera_UploadScan"</pre>
                                                    type="string"
     value="/scan" />
9
         <param name="LxCamera_UploadOdom"</pre>
                                                    type="string"
     value="/odom" />
         <param name="LxCamera_UploadLaserPose"</pre>
10
                                                    type="string"
     value="/scan_pose" />
11
        <param name="LxCamera_Error"</pre>
                                                    type="string"
     value="LxCamera_Error" />
        <param name="LxCamera_Command"</pre>
                                                    type="string"
12
     value="LxCamera_Command" />
13
         <param name="LxCamera_Mapping"</pre>
                                                    type="string"
     value="LxCamera_Mapping" />
```

```
14
        <param name="LxCamera_Location"</pre>
                                                  type="string"
     value="LxCamera_Location" />
15
        <param name="LxCamera_SetParam"</pre>
                                                  type="string"
     value="LxCamera_SetParam" />
16
        <param name="LxCamera_SwitchMap"</pre>
                                                  type="string"
     value="LxCamera_SwitchMap" />
17
        <param name="LxCamera_DownloadMap"</pre>
                                                  type="string"
     value="LxCamera_DownloadMap" />
        <param name="LxCamera_UploadMap"</pre>
                                                  type="string"
18
     value="LxCamera_UploadMap" />
        <param name="LxCamera_LocationResult"</pre>
19
                                                  type="string"
     value="LxCamera_LocationResult" />
20
        <param name="LxCamera_UploadReloc"</pre>
                                                  type="string"
     value="LxCamera_UploadReloc" />
21
        <!-- 自动曝光期望值,范围[0-100],默认:50 -->
        <param name="auto_exposure_value"</pre>
                                                  type="int"
22
    value="50" />
23
        <!-- 建图使能, true or false -->
        <param name="mapping_mode"</pre>
24
                                                  type="bool"
     value="true" />
        <!-- 定位使能, true or false -->
25
        <param name="localization_mode"</pre>
26
                                                 type="bool"
     value="false" />
        <!-- 相机至机器人外参,单位:米,度数,格式[x,y,z,yaw,pitch,roll] -->
27
28
        <rosparam param="camera_extrinsic_param"> [0.34, 0.00, 1.3, -90, 0,
29
        <!-- 激光雷达至机器人外参,单位:米,度数,格式[x,y,yaw] -->
30
        <rosparam param="laser_extrinsic_param"> [0.34, 0.11, 0.0]
    </resparam>
31
       </node>
32
33
    </launch>
```

- 3. source devel/setup.bash
- 4. roslaunch lx\_camera\_ros mapping.launch
- 5. 待算法返回异常码为20(Mapping)时,可以推动机器人录制数据
- 6. 导出录制数据。通过SDK接口实现该功能,**需先关闭建图使能**,然后导出。或者通过示例的 rostopic pub的形式导出数据,示例如下:

先关闭建图使能:

```
rostopic pub -1 /lx_localization_node/LxCamera_Mapping std_msgs/String "data: '0'"
```

然后,请根据实际情况修改路径,示例:

```
rostopic pub -1 /lx_localization_node/LxCamera_DownloadMap
std_msgs/String "data: 'your_dir/Lanxin-
MRDVS/Sample/ROS/lx_camera_node_ws/src/lx_camera_ros/map/download_map'"
```

同时通过 /lx\_localization\_node/LxCamera\_Message 话题查看下发情况:

```
1 rostopic echo /lx_localization_node/LxCamera_Message | grep DownloadMap
```

若返回结果如下,且在则导出相应路径中可以看到导出的地图,例如在 /home/fr1511b/v1pro/Lanxin-MRDVS/Sample/ros-v1pro/map 路径下生成 download\_map.zip 文件,则导出数据成功。**然后请联系MRDVS相关人员技术支持,并提供建图数据**。

```
data: "{\"cmd\":\"LxCamera_DownloadMap\",\"result\":0}
```

7. 建图完成后,需先关闭建图使能,然后将地图包(20240826之前的SDK版本,地图名后缀为.zip;20240826及之后的SDK版本,地图名后缀更改为.bin)导入至相机。通过SDK接口实现该功能,或者通过示例的rostopic pub的形式导入数据,示例如下:

```
rostopic pub -1 /lx_localization_node/LxCamera_UploadMap std_msgs/String "data: '/home/fr1511b/v1pro/Lanxin-MRDVS/Sample/ros-v1pro/map/xz9_231216.bin'"
```

同时通过 /lx\_localization\_node/LxCamera\_Message 话题查看下发情况:

```
1 rostopic echo /lx_localization_node/LxCamera_Message | grep UploadMap
```

若返回结果如下,则导入地图成功。

```
1 data: "{\"cmd\":\"LxCamera_UploadMap\",\"result\":0}
```

#### 定位

- 1. 准备好建图所需的传感器数据
  - 。 机器人里程计的topic, 类型为nav msgs/Odometry
- 2. 按照要求修改launch文件夹下 localization.launch (切记请核对) ,设置地图名(已有地图 名,如刚才导入的地图)、相机内参等

```
1 < launch>
       <node name="lx_localization_node" pkg="lx_camera_ros"</pre>
2
    type="lx_localization_node" output="screen">
 3
        <!-- 相机IP, 默认为空时按设备索引获取 -->
        <param name="ip"</pre>
 4
                                                    type="string"
     value="192.168.100.82" />
 5
        <!-- 图像显示使能 -->
                                                    type="bool"
 6
        <param name="is_show"</pre>
     value="true" />
 7
        <!-- 接收或发布ROS话题名 -->
        <param name="LxCamera_UploadScan"</pre>
 8
                                                    type="string"
     value="/scan" />
        <param name="LxCamera_UploadOdom"</pre>
 9
                                                    type="string"
     value="/odom" />
10
        <param name="LxCamera_UploadLaserPose"</pre>
                                                    type="string"
     value="/scan_pose" />
        <param name="LxCamera_Error"</pre>
11
                                                    type="string"
     value="LxCamera_Error" />
12
        <param name="LxCamera_Command"</pre>
                                                    type="string"
     value="LxCamera_Command" />
                                                    type="string"
13
        <param name="LxCamera_Mapping"</pre>
     value="LxCamera_Mapping" />
```

```
<param name="LxCamera_Location"</pre>
14
                                                  type="string"
     value="LxCamera_Location" />
15
        <param name="LxCamera_SetParam"</pre>
                                                  type="string"
     value="LxCamera_SetParam" />
16
        <param name="LxCamera_SwitchMap"</pre>
                                                  type="string"
     value="LxCamera_SwitchMap" />
17
        <param name="LxCamera_DownloadMap"</pre>
                                                  type="string"
     value="LxCamera_DownloadMap" />
        <param name="LxCamera_UploadMap"</pre>
                                                  type="string"
18
     value="LxCamera_UploadMap" />
        <param name="LxCamera_LocationResult"</pre>
19
                                                  type="string"
     value="LxCamera_LocationResult" />
20
        <param name="LxCamera_UploadReloc"</pre>
                                                  type="string"
     value="LxCamera_UploadReloc" />
        <!-- 自动曝光期望值, 范围[0-100], 默认:50 -->
21
        <param name="auto_exposure_value"</pre>
22
                                                 type="int"
    value="50" />
23
        <!-- 建图使能, true or false -->
        <param name="mapping_mode"</pre>
                                                  type="bool"
24
     value="false" />
25
        <!-- 定位使能, true or false -->
        <param name="localization_mode"</pre>
26
                                                type="bool"
     value="true" />
        <!-- 重要: 定位时地图名需有效,已上传相机并存在; 建图时,如果相机中无地图可默认输
27
    \lambda"example_map1" -->
        <param name="map_name"</pre>
28
                                        type="string"
     value="xz9_231216" />
29
        <!-- 相机至机器人外参,单位:米,度数,格式[x,y,z,yaw,pitch,roll] -->
        <rosparam param="camera_extrinsic_param"> [0.34, 0.00, 1.3, -90, 0,
30
    0] </rosparam>
        <!-- 激光雷达至机器人外参,单位:米,度数,格式[x,y,yaw] -->
31
32
        <rosparam param="laser_extrinsic_param"> [0.34, 0.11, 0.0]
    </resparam>
33
       </node>
34
35
    </launch>
```

#### 3. source devel/setup.bash

4. 发送重定位请求。将机器人移动至指定地点,发送该点位在视觉地图中的坐标即可(一般地,录图起始点坐标为(x=0, y=0, yaw=0))(注意:直接发重定位请求位姿(x=0, y=0, yaw=0)会被当成误发,可以规避坐标零点,如重定位请求位姿(x=0.01, y=0.00, yaw=0)),此部分可以通过SDK接口实现,或在示例的rostopic pub的形式发布重定位请求:

```
rostopic pub -1 /lx_localization_node/LxCamera_UploadReloc
   geometry_msgs/PoseWithCovarianceStamped "header:
2
    seq: 0
3
    stamp:
4
     secs: 0
5
     nsecs: 0
6
    frame_id: ''
7
  pose:
8
    pose:
      position: {x: 0.1, y: -0.1, z: 0.0}
9
10
     orientation: {x: 0.0, y: 0.0, z: 0.0, w: 1.0}
    11
   0.0, 0.0, 0.0,
12
     0.0, 0.0, 0.0,
13
     0.0, 0.0, 0.0, 0.0, 0.0, 0.0]"
```

同时通过 /1x\_localization\_node/LxCamera\_Message 话题查看下发情况:

1 | rostopic echo /lx\_localization\_node/LxCamera\_Message | grep UploadReloc

若返回结果如下,则导入地图成功。

```
1 data: "{\"cmd\":\"LxCamera_UploadReloc\",\"result\":0}
```

5. 待算法返回异常码为0(Tracking)时,并且可以订阅 到 /lx\_localization\_node/LxCamera\_LocationResult 即定位OK。此时,相机返回的为机器人 在视觉地图中的位姿。

# 常见问题

运行该示例出现问题时,可通过相机SDK日志快速排查,常见问题如下:

- 1. 若通过ROS示例会出现卡顿、数据通讯异常,考虑ROS主从问题,去掉相关设置即可
- 2. 若关闭ROS示例程序后重新运行失败,考虑相机流占用问题,请耐心等待10-20s,再重新打开
- 3. 若出现SDK库加载失败、网络频繁丢包等问题,请确认配置脚本 install.sh 是否运行生效
- 4. 重定位成功时程序崩溃,考虑ROS相机内部配置问题,请联系相关人员解决

# 参考

- LxCameraApi(C、C++)开发者指南.pdf
- LxCameraViewer使用说明书.pdf
- V1Pro用户使用手册.pdf