

数据采集工具使用介绍

摄像头数据采集工具

环境

python3+opencv

工具位置

http://192.168.5.10:9980/wangzhongju/lidar_proj/tree/master/tools/camera

使用方法

切换到 tools/camera 文件夹下，终端输入以下命令：

```
python3 get_video.py -ip 192.168.5.30 -u admin -p aa123456
```

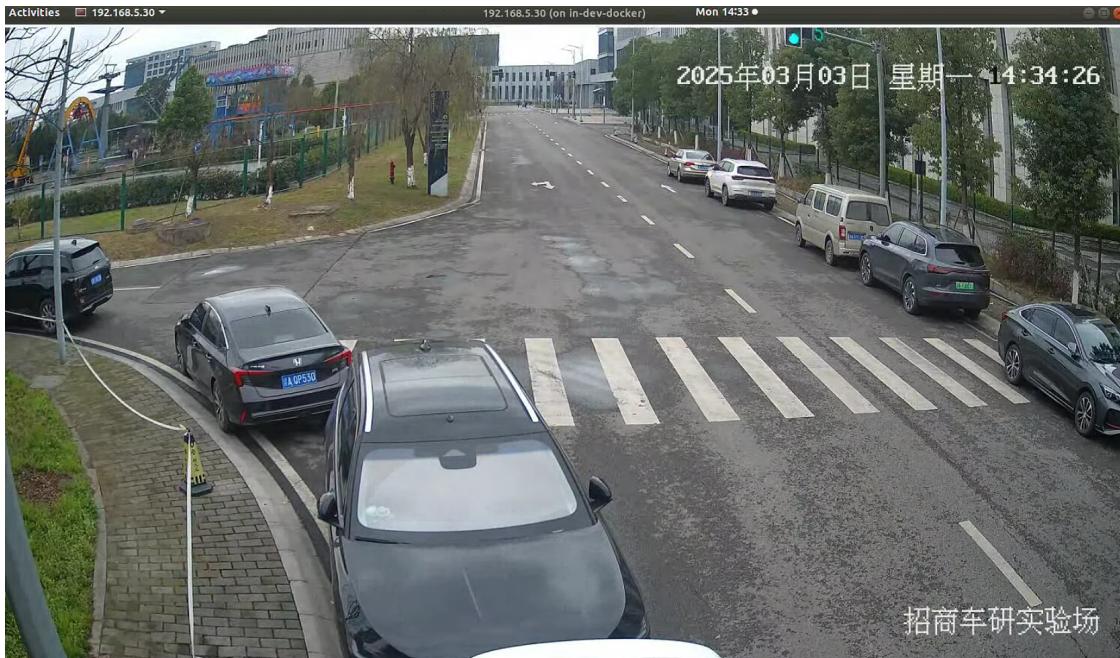
参数说明：

-ip : 替换为实际需要采集数据的摄像头设备 ip 地址

-u: Username, 摄像机用户

-p: password, 摄像机密码

1. 输入上述命令后，界面弹出对应 ip 地址的相机画面



终端显示：

```
cjy@tn-dev-docker:/workspace/tools/camera$ python3 get_video.py --ip 192.168.5.30
width: 1920
width: 1080
OpenCV: FFmpeg: tag 0x44495658/'XVID' is not supported with codec id 12 and format 'mp4 / MP4 (MPEG-4 Part 14)'
OpenCV: FFmpeg: fallback to use tag 0x7634706d/'mp4v'
```

2. 采集视频数据，鼠标放置到相机画面上，英文输入模式下，键盘点击 'p' 键，终端滚动如下数据表示视频录制中：

再次点击 'p' 键视频录制暂停，终端停止信息滚动并显示：

3. 采集图像数据，鼠标放置到相机画面上，键盘点击 's' 键，终端输出图片保存地址：

```
recording ...
Recording paused.
Saved frame as ./img/192.168.5.30/20250303_144103.jpg
```

每点击一次 's'，保存一帧图片数据。

4. 以上所有的键盘信号输入需要将鼠标指针放置到视频画面上，且输入模式为英文输入，视频录制与暂停点击 'p'，图片保存点击 's'，退出点击 'Esc'。

点云数据采集工具

环境

ros1

工具位置

http://192.168.5.10:9980/wangzhongju/lidar_proj/tree/master/tools/lidar

使用方法

1. 修改配置

配置文件路径: tools/lidar/src/my_rtsp_streamer/configstreams.yaml

```
1   streams:
2     - name: camera_30
3       url: "rtsp://admin:aa123456@192.168.5.30"
4       # - name: camera_31
5       #   url: "rtsp://admin:aa123456@192.168.5.31"
6       #- name: camera_3
7       #   url: "/workspace/data/test5.mp4"
8
```

name: 摄像头名称, 如 camera_30, 为便于区分, 30与 ip 地址第四位对齐

url: rtsp地址, 如 "rtsp://admin:aa123456@192.168.5.30"

如果有多个设备, 可按照格式添加多对 name 与 url。

2. 编译与运行

切换到 tools/lidar 文件夹下, 终端输入以下命令:

```
chmod 777 run.sh
./run.sh
```

终端显示如下表示编译运行通过:

```
-- Found PythonInterp: /usr/bin/python3 (found version "3.8.10")
-- Found Threads: TRUE
-- Using Python nosetests3
-- catkin 0.8.10
-- BUILD_SHARED_LIBS is on
-- BUILD_SHARED_LIBS is on
-- 
-- ~~~ traversing 1 packages in topological order:
-- ~~~ - my_rtsp_streamer
-- 
-- *** processing catkin package: 'my_rtsp_streamer'
-- => add_subdirectory(my_rtsp_streamer)
CMake Deprecation Warning at my_rtsp_streamer/CMakeLists.txt:1 (cmake_minimum_required):
  Compatibility with CMake < 3.5 will be removed from a future version of
  CMake.

  Update the VERSION argument <min> value or use a ...<max> suffix to tell
  CMake that the project does not need compatibility with older versions.

-- Found OpenCV: /usr (found version "4.2.0")
-- Configuring done (1.1s)
-- Generating done (0.0s)
-- Build files have been written to: /workspace/tools/lidar/build
#####
##### Running command: "make -j16 -l16" in "/workspace/tools/lidar/build"
#####
[ 33%] Building CXX object my_rtsp_streamer/CMakeFiles/rtsp_streamer_node.dir/src/rtsp_streamer_node.cpp.o
[ 66%] Building CXX object my_rtsp_streamer/CMakeFiles/rtsp_streamer_node.dir/src/rtsp_streamer.cpp.o
[100%] Linking CXX executable /workspace/tools/lidar/devel/lib/my_rtsp_streamer/rtsp_streamer_node
[100%] Built target rtsp_streamer_node
cyy@in-dev-docker:/workspace/tools/lidar$ ... logging to /home/cyy/.ros/log/48116024-f801-11ef-83b5-a8a1596c0ab3/roslaunch-in-dev-docker-26059.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://in-dev-docker:40577/
SUMMARY
========
PARAMETERS
  * /rosdistro: noetic
  * /rosversion: 1.17.0
  * /rtsp_streamer_node/config_file: /workspace/tools/...
NODES
  /
    rtsp_streamer_node (my_rtsp_streamer/rtsp_streamer_node)
ROS_MASTER_URI=http://localhost:11311
process[rtsp_streamer_node-1]: started with pid [26097]
```

3. 数据验证

终端输入

```
source /opt/ros/noetic/setup.bash // noetic可能需要替换为设备安装的ros版本  
rostopic list  
rostopic hz /camera_30/image_raw/compressed
```

终端显示如下：

```
cjy@in-dev-docker:/workspace/tools/lidar$ rostopic list  
/camera_30/image_raw  
/camera_30/image_raw/compressed  
/camera_30/image_raw/compressed/parameter_descriptions  
/camera_30/image_raw/compressed/parameter_updates  
/camera_30/image_raw/compressedDepth  
/camera_30/compressedDepth/parameter_descriptions  
/camera_30/compressedDepth/parameter_updates  
/camera_30/image_raw/theora  
/camera_30/image_raw/theora/parameter_descriptions  
/camera_30/image_raw/theora/parameter_updates  
/rosout  
/rosout_agg  
cjy@in-dev-docker:/workspace/tools/lidar$ █
```

```
cjy@in-dev-docker:/workspace/tools/lidar$ rostopic hz /camera_30/image_raw/compressed  
subscribed to [/camera_30/image_raw/compressed]  
average rate: 19.942  
    min: 0.036s max: 0.069s std dev: 0.00906s window: 20  
average rate: 19.970  
    min: 0.030s max: 0.069s std dev: 0.01061s window: 40  
average rate: 20.005  
    min: 0.030s max: 0.069s std dev: 0.01010s window: 60  
average rate: 19.923  
    min: 0.029s max: 0.069s std dev: 0.01049s window: 80  
average rate: 19.966  
    min: 0.029s max: 0.069s std dev: 0.01035s window: 99
```

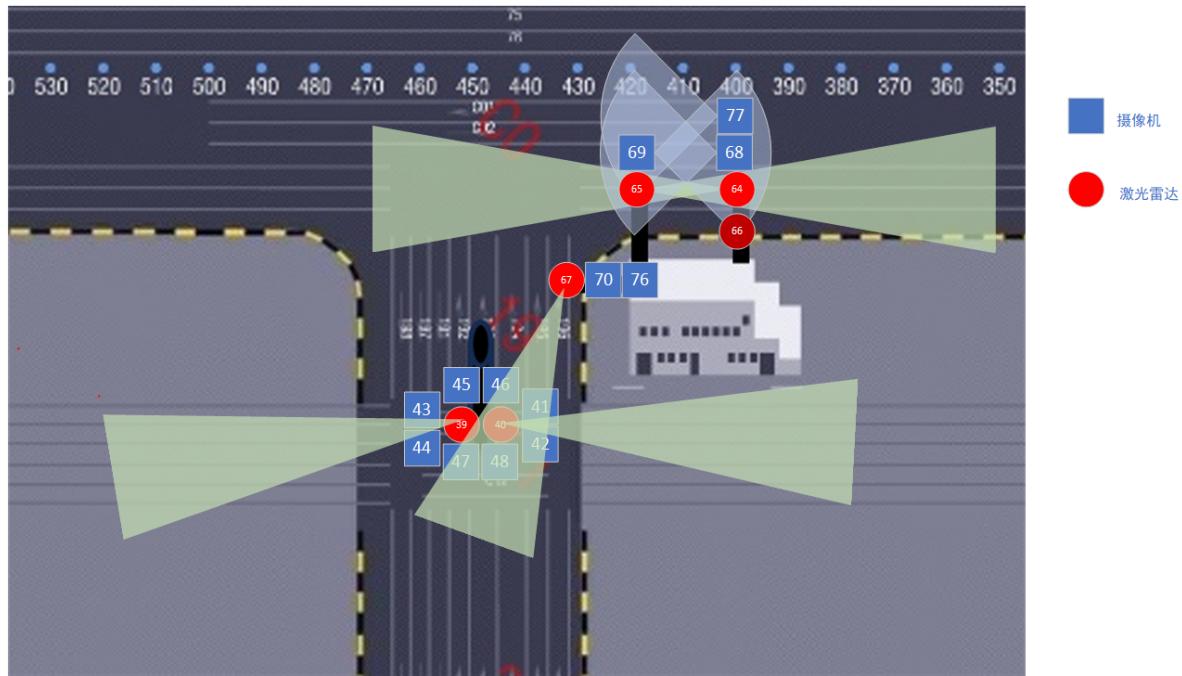
4. 雷达点云数据

找到激光雷达驱动安装设备，如速腾激光雷达驱动安装运行后，执行如下命令显示：

```
cjy@in-dev-docker:/workspace$ rostopic list  
/33/rslidar_packets  
/33/rslidar_points  
/rosout  
/rosout_agg
```

tip：上述步骤1、2、3在激光雷达驱动运行设备上执行

示例



如上，所有设备的 ip 前三位为 10.28.49.xx，66 为机械式激光雷达，64、65 为固态激光雷达，朝向如图，68、69、77 为摄像头，现在录制沿海路点云与视频数据：

1. 确认激光雷达 64、65、66 驱动安装位于哪台mec 上，假如安装与 10.28.59.70，执行：

```
rostopic list
```

终端应输出包含：

```
/64/rslidar_points  
/65/rslidar_points  
/66/rslidar_points
```

2. 将 http://192.168.5.10:9980/wangzhongju/lidar_proj/tools/lidar 拷贝到mec上用户目录

执行**使用方法**步骤 1，修改 tools/lidar/src/my_rtsp_streamer/configstreams.yaml 配置

```
streams:  
  - name: camera_68  
    url: "rtsp://admin:abc12345@10.28.49.68"  
  - name: camera_69  
    url: "rtsp://admin:abc12345@10.28.49.69"  
  - name: camera_77  
    url: "rtsp://admin:abc12345@10.28.49.77"
```

3. 执行**使用方法**步骤2

4. 数据录制

```
rosbag record /camera_68/image_raw/compressed  
/camera_69/image_raw/compressed /camera_77/image_raw/compressed  
/64/rslidar_points /65/rslidar_points /66/rslidar_points
```

录制结束后，点击 **ctrl+c** 退出数据录制，数据包名称为 <录制开始时间>.bag

tip：注意，录制之前检查磁盘空间大小，避免磁盘占满，保证rosbag保存地址至少100G以上空间。