

# Intelligentcar2023-FZ3B

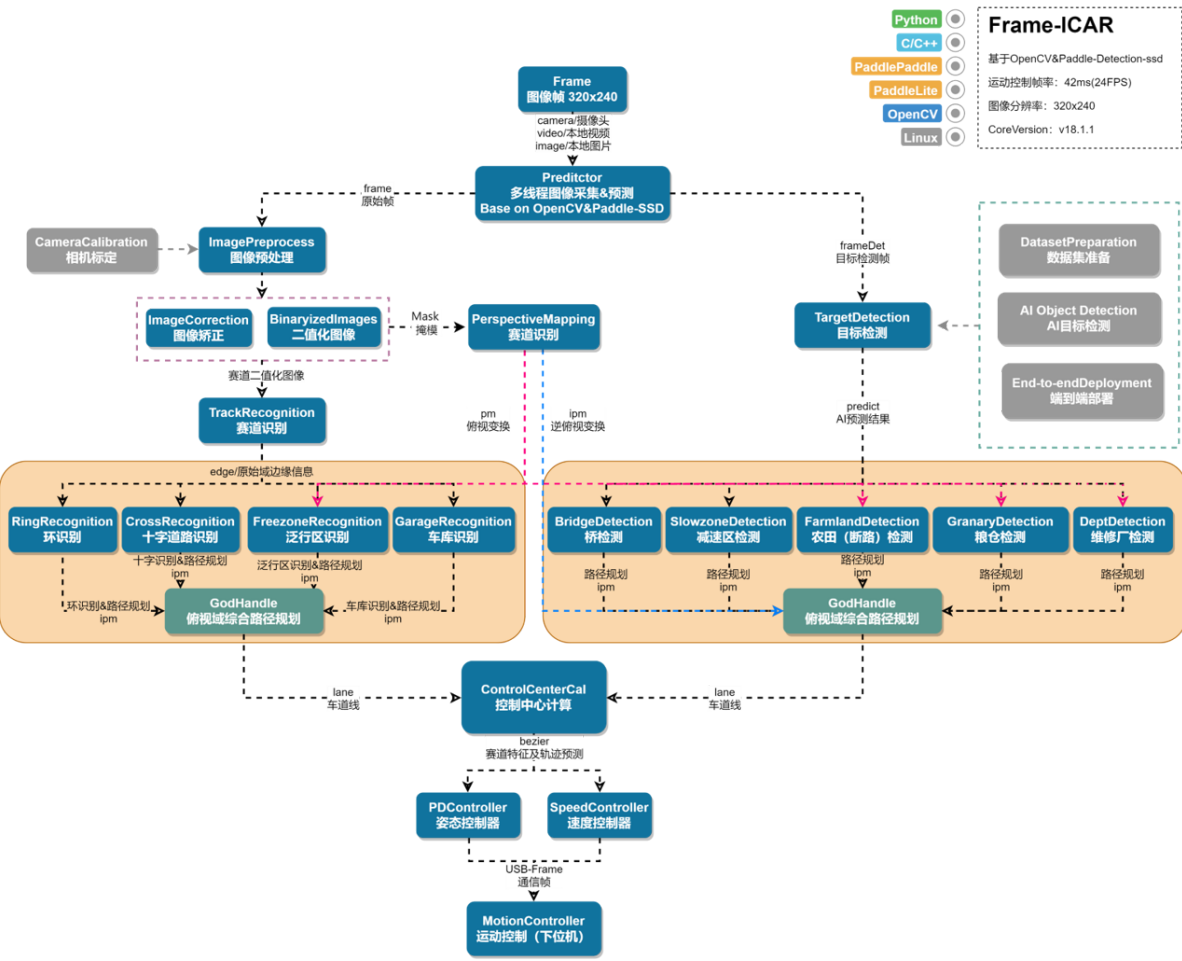
## 介绍

全国大学生智能汽车竞赛 - [赛曙科技]开源软件工程，基于百度Edgeboard-FZ3B赛事教育版边缘计算板卡开发，

严肃声明！！该软件方案仅限智能汽车竞赛学习交流，本人基于智能车爱好者维度向公众公开源代码，不涉及任何商业行为，请全体技术爱好者共同监督指导！！

此工程包含完全模型组2023年“智慧农业”主题完整代码示例，相关演示视频请查看具体链接。

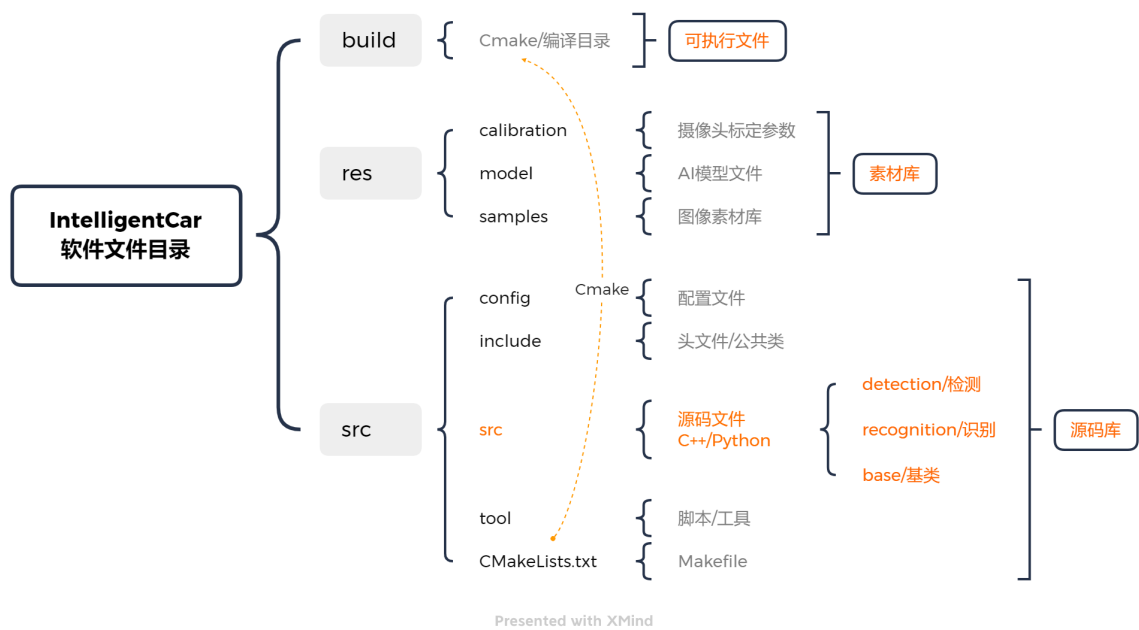
## 软件架构



## 使用说明

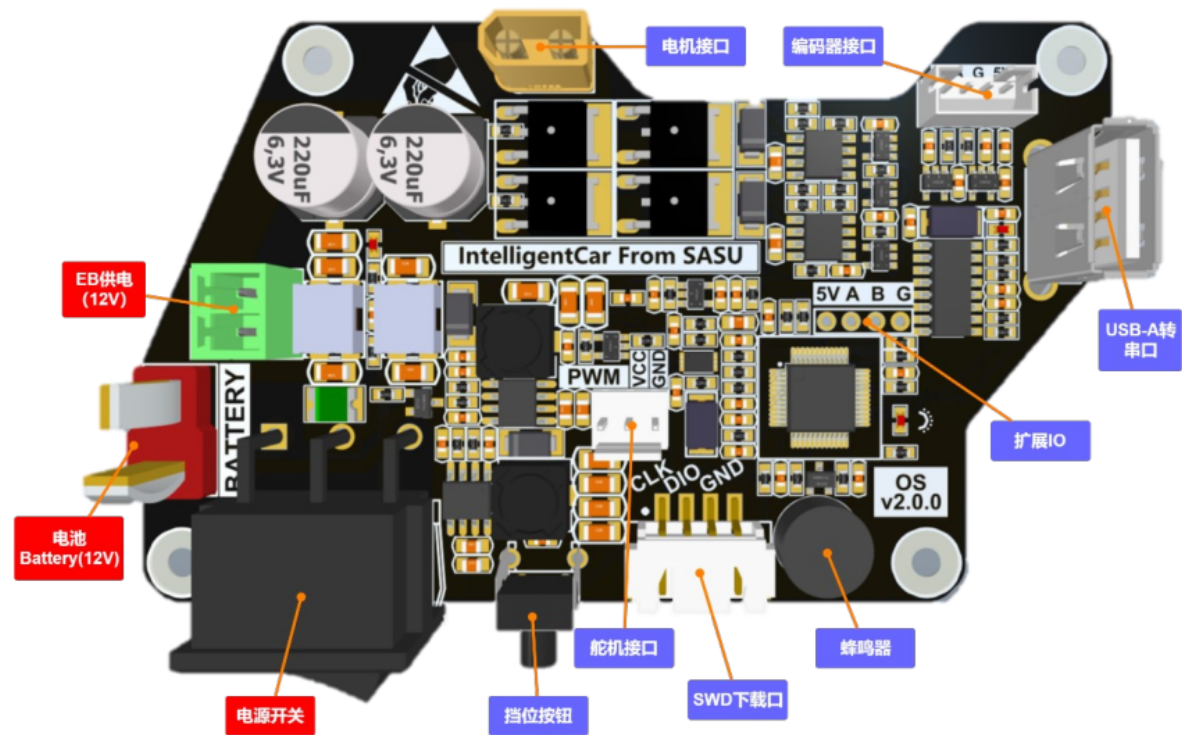
1. 该软件工程基于C++/OpenCV2/Python3.7/PaddlePaddle-Detection搭建;
2. 基于百度智能汽车-完全模型组及赛事开源Edgeboard板卡FZ3B调试开发;
3. 默认将该软件下载到EB: root@EdgeBoard:~/workspace目录下即可启用;
4. 除此之外, 该软件支持Debug仿真调试(视频)/比赛模式两种方式运行;

# 文件目录



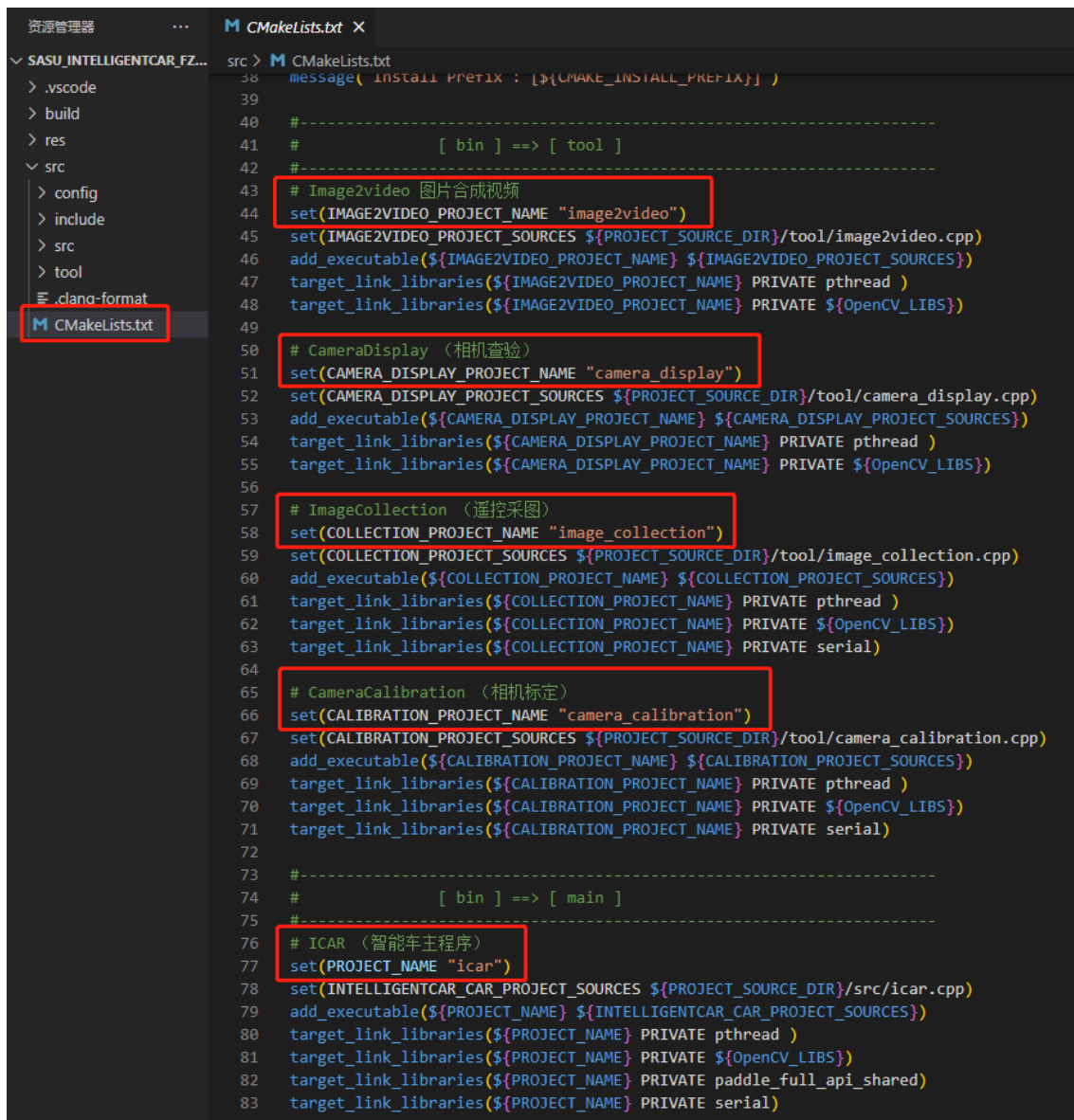
## 下位机说明

请注意：该代码工程需配合赛曙科技开源智控板（CarDo）使用，用户也可按照开源协议准备自己的下位机，通过USB转串口协议联合使用。



## 程序启动说明

1. 编译代码/Make

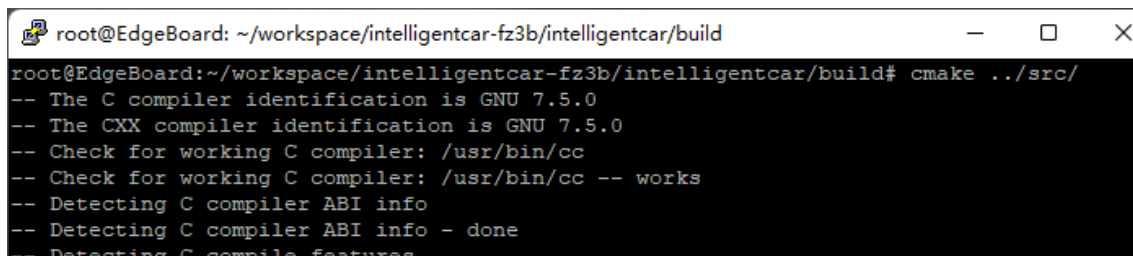


```
src > M CMakeLists.txt
38 message( INSTALL_PREFIX : [ ${CMAKE_INSTALL_PREFIX} ] )
39
40 #-----
41 # [ bin ] ==> [ tool ]
42 #-----
43 # Image2video 图片合成视频
44 set(IMAGE2VIDEO_PROJECT_NAME "image2video")
45 set(IMAGE2VIDEO_PROJECT_SOURCES ${PROJECT_SOURCE_DIR}/tool/image2video.cpp)
46 add_executable(${IMAGE2VIDEO_PROJECT_NAME} ${IMAGE2VIDEO_PROJECT_SOURCES})
47 target_link_libraries(${IMAGE2VIDEO_PROJECT_NAME} PRIVATE pthread )
48 target_link_libraries(${IMAGE2VIDEO_PROJECT_NAME} PRIVATE ${OpenCV_LIBS})
49
50 # CameraDisplay (相机查验)
51 set(CAMERA_DISPLAY_PROJECT_NAME "camera_display")
52 set(CAMERA_DISPLAY_PROJECT_SOURCES ${PROJECT_SOURCE_DIR}/tool/camera_display.cpp)
53 add_executable(${CAMERA_DISPLAY_PROJECT_NAME} ${CAMERA_DISPLAY_PROJECT_SOURCES})
54 target_link_libraries(${CAMERA_DISPLAY_PROJECT_NAME} PRIVATE pthread )
55 target_link_libraries(${CAMERA_DISPLAY_PROJECT_NAME} PRIVATE ${OpenCV_LIBS})
56
57 # ImageCollection (遥控采图)
58 set(COLLECTION_PROJECT_NAME "image_collection")
59 set(COLLECTION_PROJECT_SOURCES ${PROJECT_SOURCE_DIR}/tool/image_collection.cpp)
60 add_executable(${COLLECTION_PROJECT_NAME} ${COLLECTION_PROJECT_SOURCES})
61 target_link_libraries(${COLLECTION_PROJECT_NAME} PRIVATE pthread )
62 target_link_libraries(${COLLECTION_PROJECT_NAME} PRIVATE ${OpenCV_LIBS})
63 target_link_libraries(${COLLECTION_PROJECT_NAME} PRIVATE serial)
64
65 # CameraCalibration (相机标定)
66 set(CALIBRATION_PROJECT_NAME "camera_calibration")
67 set(CALIBRATION_PROJECT_SOURCES ${PROJECT_SOURCE_DIR}/tool/camera_calibration.cpp)
68 add_executable(${CALIBRATION_PROJECT_NAME} ${CALIBRATION_PROJECT_SOURCES})
69 target_link_libraries(${CALIBRATION_PROJECT_NAME} PRIVATE pthread )
70 target_link_libraries(${CALIBRATION_PROJECT_NAME} PRIVATE ${OpenCV_LIBS})
71 target_link_libraries(${CALIBRATION_PROJECT_NAME} PRIVATE serial)
72
73 #-----
74 # [ bin ] ==> [ main ]
75 #-----
76 # ICAR (智能车主程序)
77 set(PROJECT_NAME "icar")
78 set(INTELLIGENTCAR_CAR_PROJECT_SOURCES ${PROJECT_SOURCE_DIR}/src/icar.cpp)
79 add_executable(${PROJECT_NAME} ${INTELLIGENTCAR_CAR_PROJECT_SOURCES})
80 target_link_libraries(${PROJECT_NAME} PRIVATE pthread )
81 target_link_libraries(${PROJECT_NAME} PRIVATE ${OpenCV_LIBS})
82 target_link_libraries(${PROJECT_NAME} PRIVATE paddle_full_api_shared)
83 target_link_libraries(${PROJECT_NAME} PRIVATE serial)
```

如上图所示CMakeLists.txt文件：本工程包含智能车主程序和其它可被编译运行的程序，用户编译后执行，编译步骤如下：

cd到工程build路径下，执行

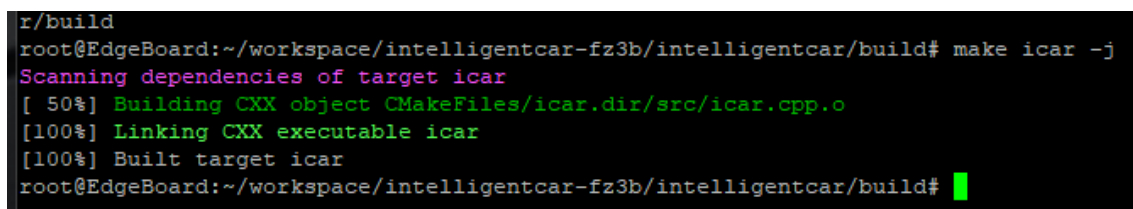
```
sudo cmake ../src/
```



```
root@EdgeBoard: ~/workspace/intelligentcar-fz3b/intelligentcar/build
root@EdgeBoard:~/workspace/intelligentcar-fz3b/intelligentcar/build# cmake ../src/
-- The C compiler identification is GNU 7.5.0
-- The CXX compiler identification is GNU 7.5.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
```

编译代码：

```
sudo make icar -j
```



```
r/build
root@EdgeBoard:~/workspace/intelligentcar-fz3b/intelligentcar/build# make icar -j
Scanning dependencies of target icar
[ 50%] Building CXX object CMakeFiles/icar.dir/src/icar.cpp.o
[100%] Linking CXX executable icar
[100%] Built target icar
root@EdgeBoard:~/workspace/intelligentcar-fz3b/intelligentcar/build#
```

## 2. 配置文件 (config) 修改:

方便比赛调试, 本工程摘取重要参数关联至motion.json文件, 用户在调试过程直接修改配置文件, 无需编译代码即可运行。

其中, 可通过debug标志启动仿真(跑视频)模式和比赛(竞速)模式, 仿真模式下默认调取本地视频 ("pathVideo": "../res/samples/sample.mp4"), 并且通过远程桌面观察实时图像和运算结果; 比赛模式将关闭图像显示和图形绘制(省时10ms)。

另外, 包括智能车速度、姿态模型参数、赛道元素使能等关键参数, 用户按照配置文件详细说明修改。

```
1  {
2      "speedLow": 0.8,
3      "speedHigh": 0.8,
4      "speedDown": 0.8,
5      "speedBridge": 1.2,
6      "speedSlowzone": 0.8,
7      "speedGarage": 0.8,
8      "runP1": 1.7,
9      "runP2": 0.014,
10     "runP3": 0.01,
11     "turnP": 3.5,
12     "turnD": 3.5,
13     "debug": false,
14     "saveImage": false,
15     "rowCutUp": 20,
16     "rowCutBottom": 10,
17     "disGarageEntry": 0.35,
18     "GarageEnable": true,
19     "BridgeEnable": true,
20     "FreezoneEnable": false,
21     "RingEnable": true,
22     "CrossEnable": true,
23     "GranaryEnable": true,
24     "DepotEnable": true,
25     "FarmlandEnable": true,
26     "SlowzoneEnable": true,
27     "circles": 2,
28     "pathVideo": "../res/samples/sample.mp4",
29     "record": [
30         {
31             "#speedLow": "智能车最低速",
32             "#speedHigh": "智能车最高速",
33             "#speedDown": "特殊元素减速速度",
34             "#speedBridge": "坡道(桥)速度",
35             "#speedSlowzone": "慢行区行驶速度",
36             "#speedGarage": "出入车库速度",
37             "#runP1": "一阶比例系数: 直线控制量",
38             "#runP2": "二阶比例系数: 弯道控制量",
39             "#runP3": "三阶比例系数: 弯道控制量",
40             "#turnP": "一阶比例系数: 转弯控制量",
41             "#turnD": "一阶微分系数: 转弯控制量",
42             "#debug": "调试模式使能(存图|看图)",
```

修改motion.json文件后, 不必再次编译即可运行, 方便后期调试。可通过debug标志启动仿真(跑视频)模式和比赛(竞速)模式, 仿真模式下默认调取本地视频, 并且通过VNC观察实时图像运算结果; 比赛模式将关闭图像显示和图形绘制(省时间/10ms)。

## 3. 连接下位机, 启动程序

运行icar程序需要连接下位机，否则程序退出，插入ch340/cp2102的USB串口即可（或注释串口程序）。

```
root@EdgeBoard: /workspace/intelligentcar-fz3b/intelligentcar/build# ./icar
--- runP1:1.4 | runP2:0.015 | runP3:0
--- turnP:3.5 | turnD:3.5
--- speedLow:1.1m/s | speedHigh:1.4m/s
Camera Param: frame rate = 23.976 width = 320 height = 240
Config:{"format":"RGB","input_height":300,"input_width":300,"labels_file_name":"label
list.txt","mean":[127.5,127.5,127.5],"model_file_name":"mobilenet-ssd-model","params_
file_name":"mobilenet-ssd-params","scale":[0.007843,0.007843,0.007843],"threshold":0.3
Model Config Init Success !!!
Predictor Init Success !!!
run frame time : 0ms
run frame time : 805ms
run frame time : 63ms
run frame time : 47ms
run frame time : 59ms
run frame time : 57ms
run frame time : 50ms
run frame time : 58ms
run frame time : 66ms
```

配置文件信息

AI模型信息

控制帧率/调试模式

#### 4. 查看运行结果 (debug模式)



#### 5. 开始比赛 (竞赛模式)

脱机运行：小车在赛道上奔跑时可通过 "nohup" 命令启动程序，此时拔掉网线继续运行。

```
root@EdgeBoard: ~/workspace/intelligentcar-fz3b/intelligentcar/build
root@EdgeBoard:~/workspace/intelligentcar-fz3b/intelligentcar/build# nohup ./icar
nohup: ignoring input and appending output to 'nohup.out'
```

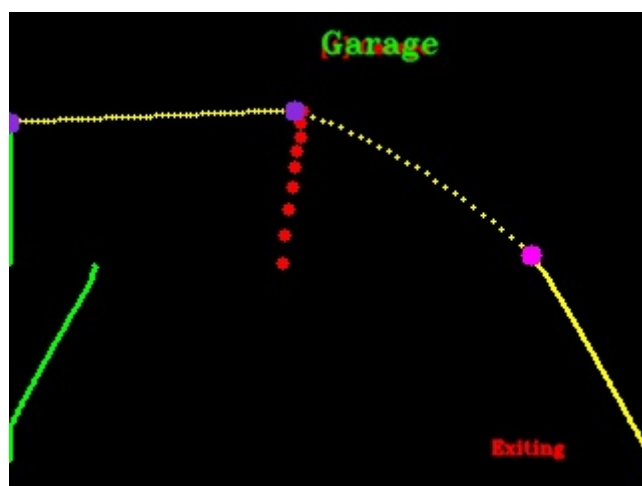
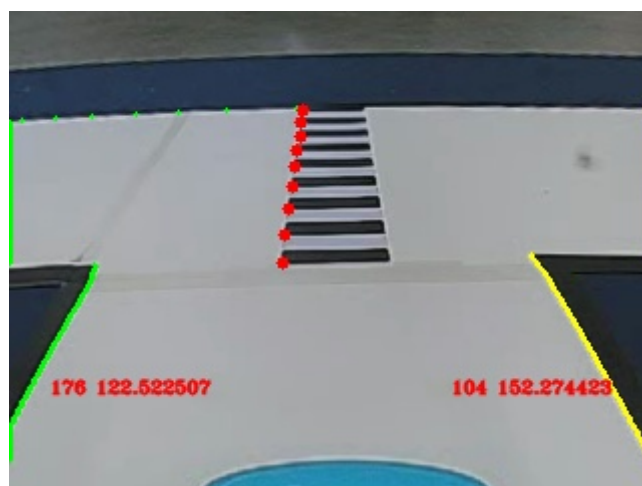
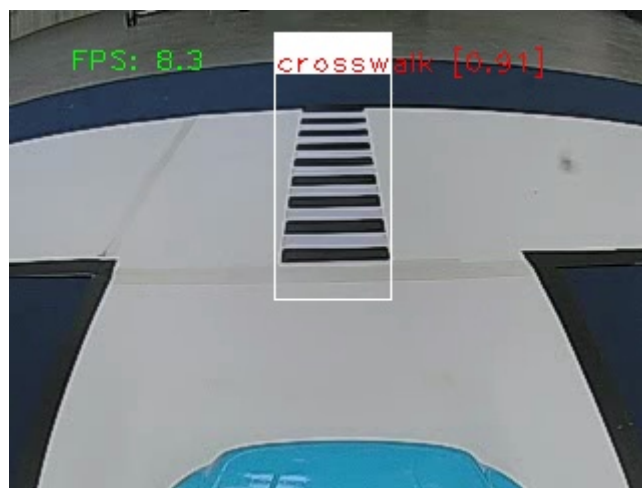
若程序无法自行退出，待重新插上网线后，通过Ctrl+C终止进程，或者查询当前进程ID：

```
ps -ef|grep ./icar
```

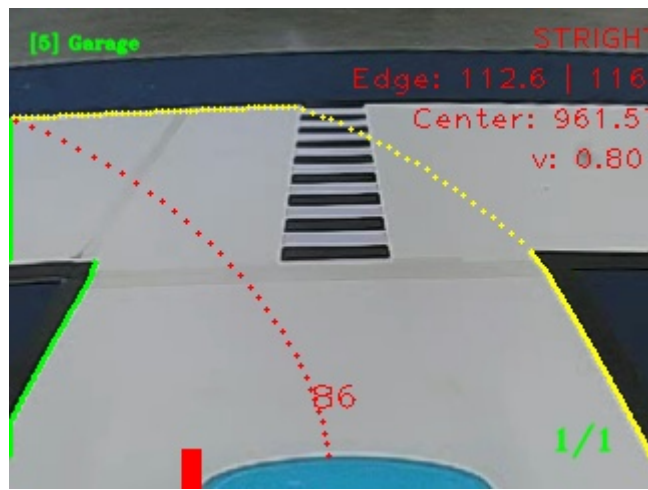
然后杀掉当前进程 (kill -9 ID) 。

## 图像处理思路

相机采图 → 图像校正 → AI预处理 → 赛道识别 (Track) → 特殊元素识别 (Rec/Det) → 路径拟合 → 运动控制模型计算 → 下位机通信：

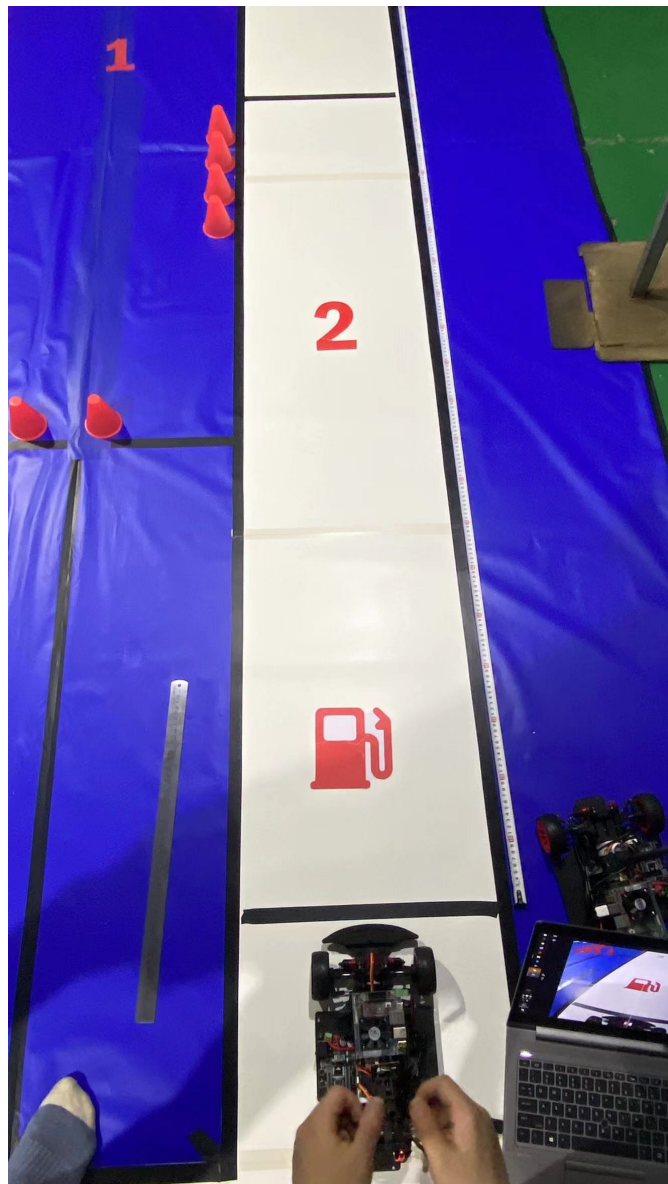


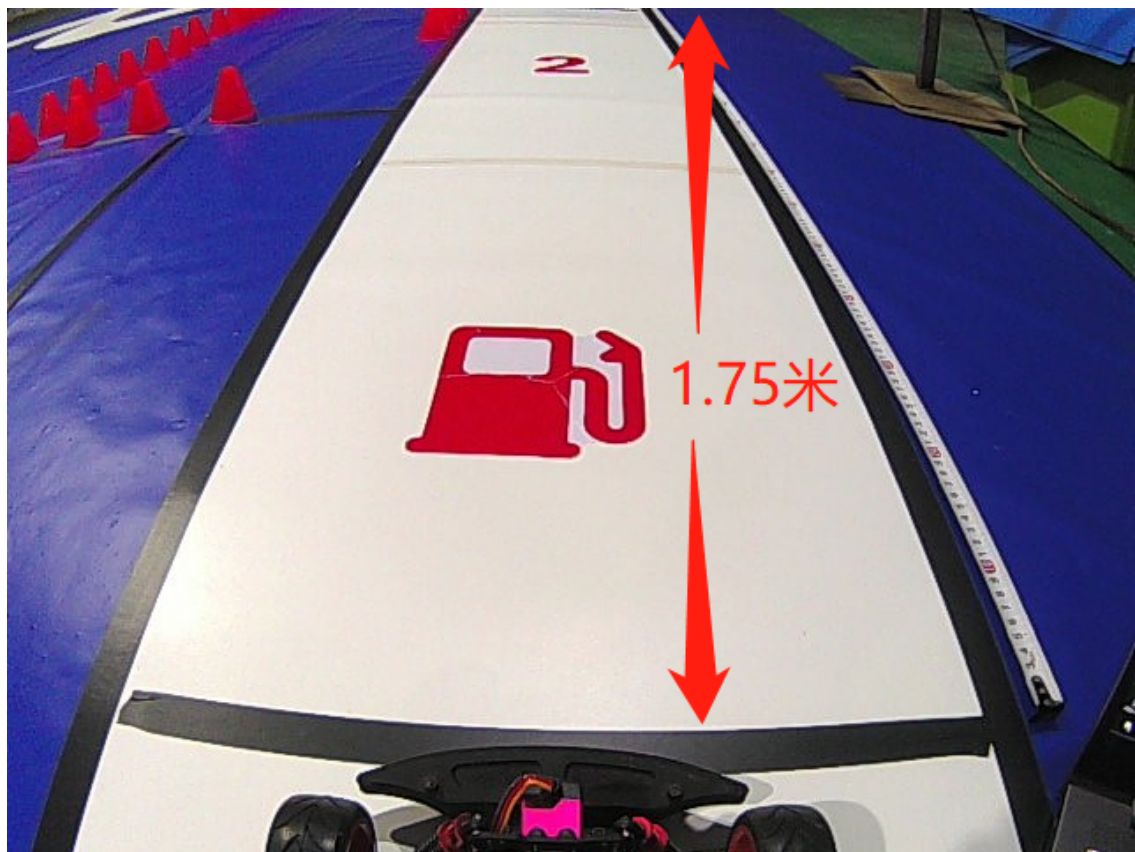




## 车模配置

1. 摄像头前瞻距离：1.6~2.2米（推荐1.75m）





2. 下位机设计PID模型/电机模型-控速，控制单位：m/s (float)
3. 下位机舵机控制，TIM/定时器计数上限20000，频率50Hz，占空比控制阈值[500, 2500]。上位机理论控制阈值[1100, 1500, 1900] / 左|中|右，舵机中值及左右阈值在实际的标定值上偏移。