

## CyberTORCS 虚拟仿真平台在线可视化调试插件 TORCS-Visual 0.1 内测版使用手册

### 兼容性说明：

本工具基于计算机视觉库 OpenCV 开发。由于 OpenCV 高度依赖于 VC 编译器版本，而 OpenCV 官方团队从 Visual Studio 2015（VC14）开始就不再对 Win32 开发环境进行主动支持而只发布 X64 开发环境下的安装文件。

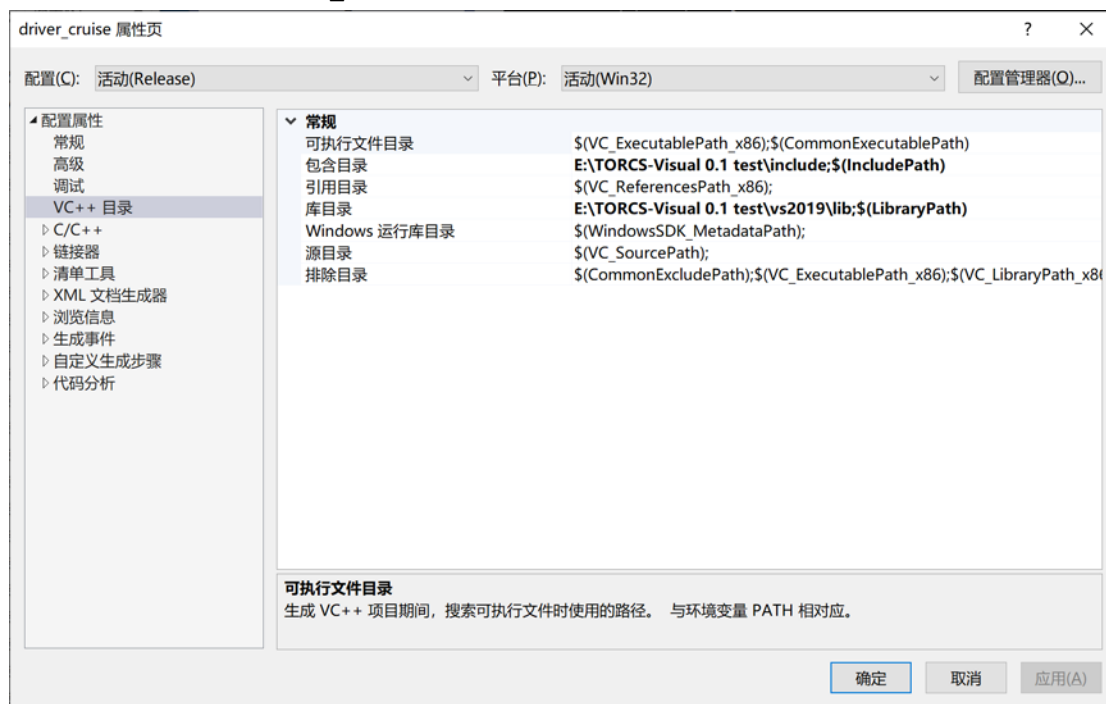
考虑到大部分同学使用的是 Visual Studio 2019，（而不是本人所使用的 CyberTORCS 的平台开发环境 Visual Studio 2010），因此在如何在 Visual Studio 2019 上兼容只在 X64 模式下运行的 OpenCV 和只在 Win32 模式下运行的 CyberTORCS，就成了平台兼容性最大的挑战。

目前该问题已通过在 Win32 模式下使用 Visual Studio 2019 对 OpenCV 进行源码编译解决。这个内测版本仅在 Visual Studio 2019（VC16）下可以使用。建议：

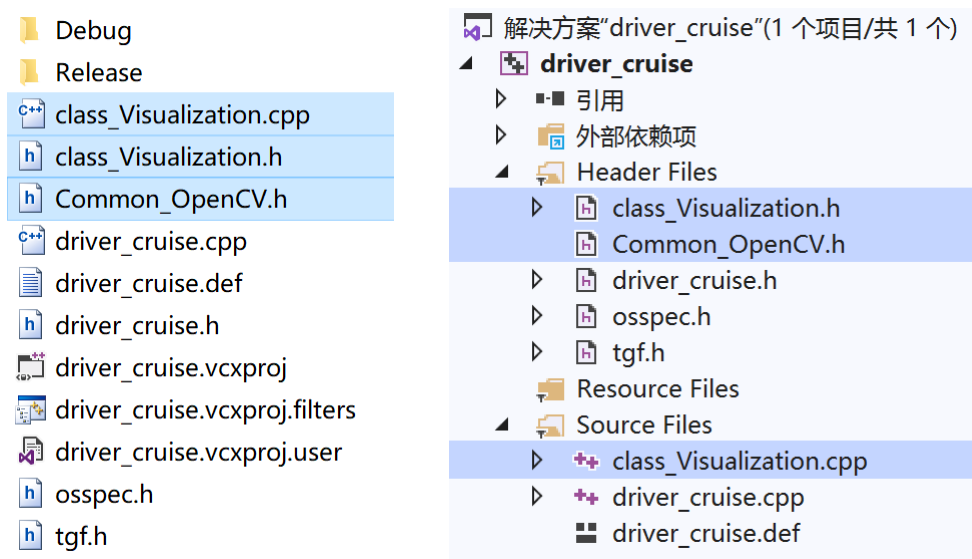
1. 使用 Visual Studio 2019（VC16）的同学直接按照本手册配置插件
2. 使用 Visual Studio 2010（VC10）的同学私下联系我索要该版本插件
3. 使用其他版本 Visual Studio 的同学建议使用交大官方资源升级至 Visual Studio 2019，如果因为硬盘空间不足或者想挑战自己对其他版本做兼容的同学，欢迎私下联系我索要兼容方法（期间可能会出现各种意想不到的问题）

### 安装说明：

#### 1. 解压插件，更改 driver\_cruise 属性，增加包含目录和库目录（以放在 E 盘根目录为例）



2. 将 3 个源文件拷贝到 **driver\_cruise** 源文件目录下，然后添加到工程中

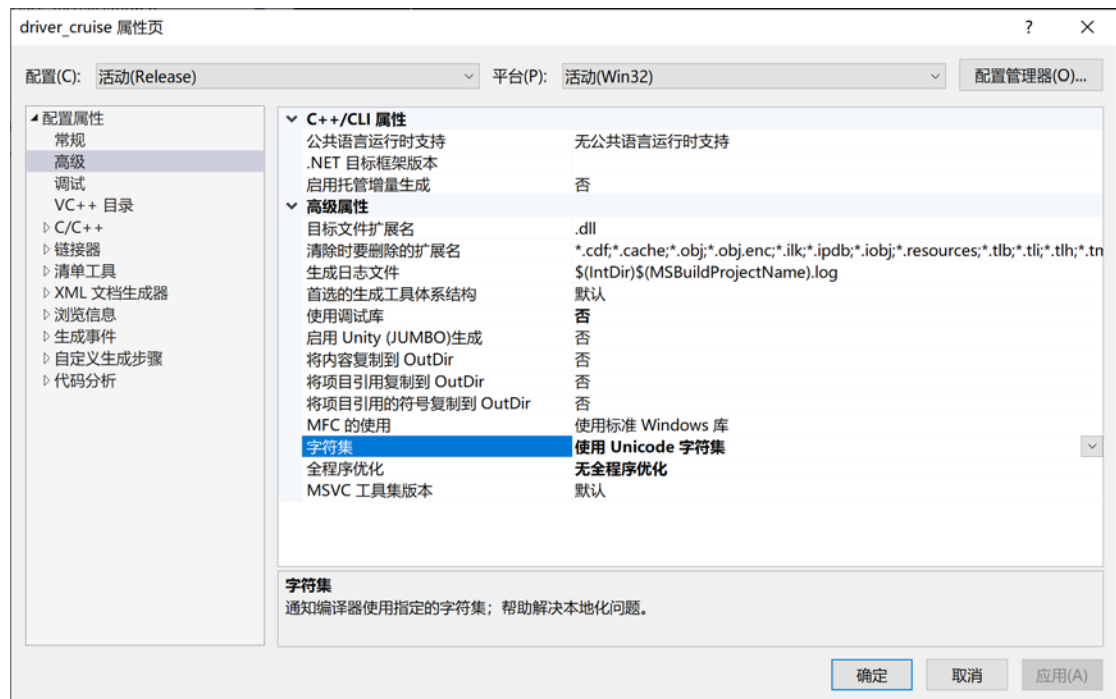


3. 打开工程中的 **osspec.h** 文件，把下图两行注释掉

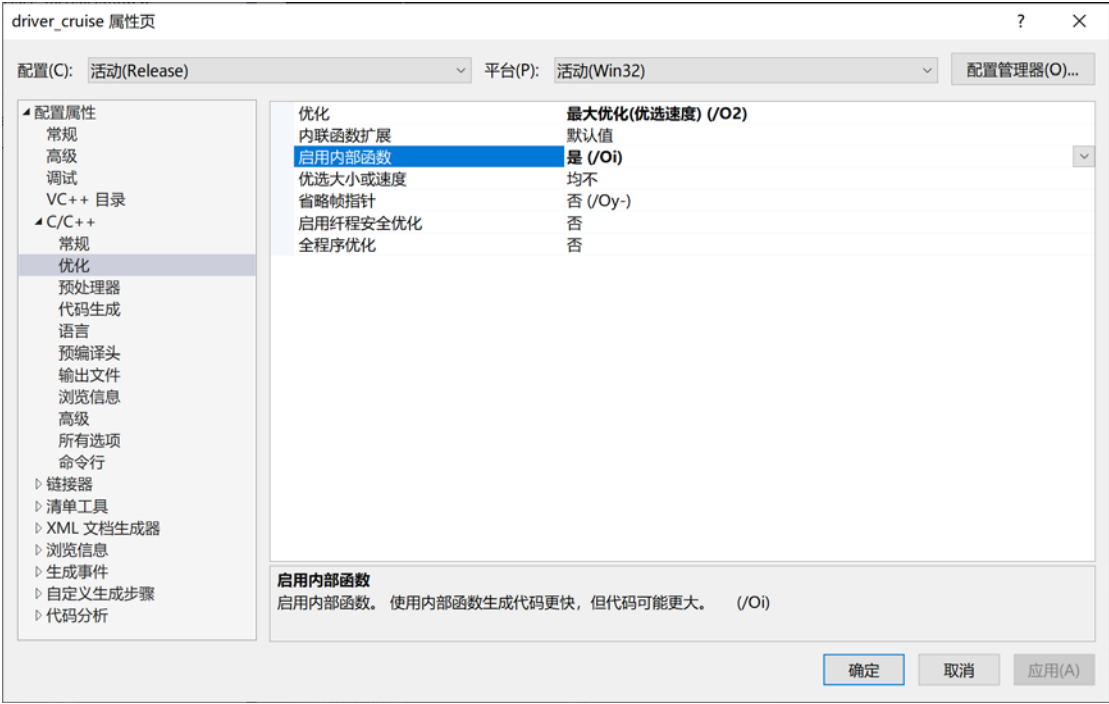
```
37 // #define uint unsigned int
38 // #define uchar unsigned char
```

4. 更改 **driver\_cruise** 属性

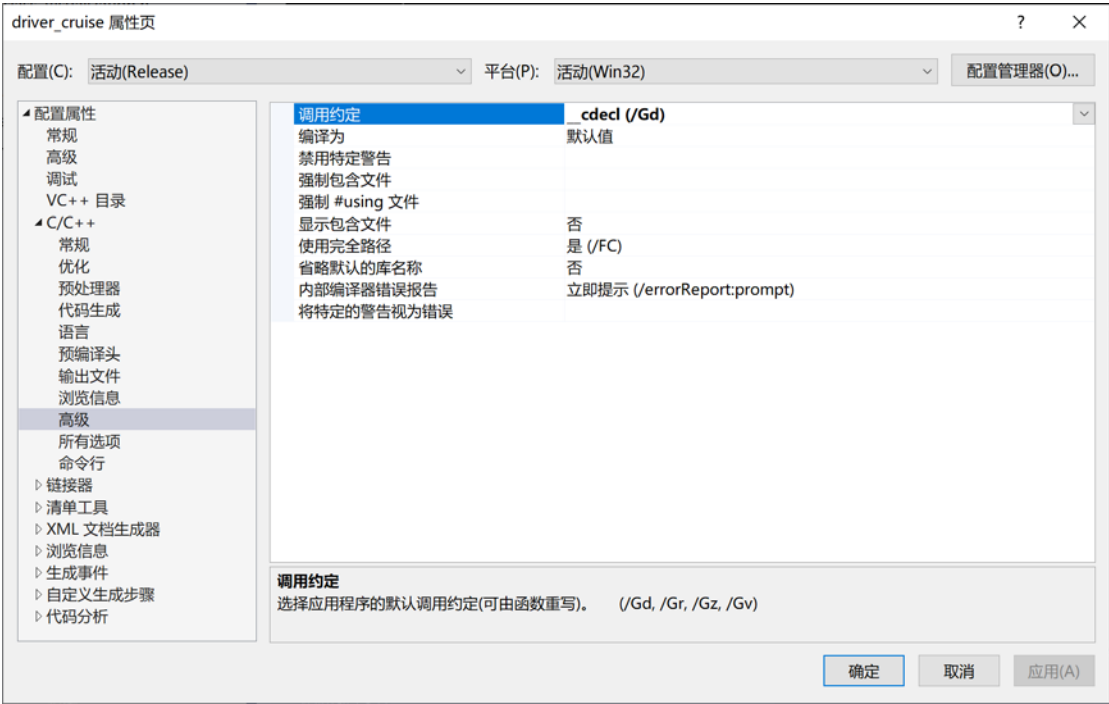
在高级->字符集处，改为：使用 Unicode 字符集



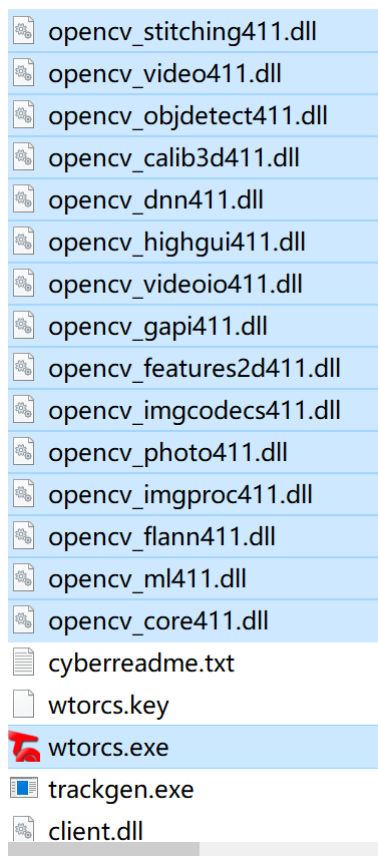
在 C/C++ -> 优化->启用内部函数处，改为：是 (/Oi)



在 C/C++->高级->调用约定处，改为：\_cdecl(/Gd)



5. 将本插件 vs2019/bin 下的 dll 拷贝到 wtorcs.exe 所在文件夹内



使用说明:

样例程序文件夹中对应了目前开发的使用本插件的 cruise.cpp 样例

首先, 需要引用本插件的 class\_Visualization.h

```
11 #include "class_Visualization.h"
```

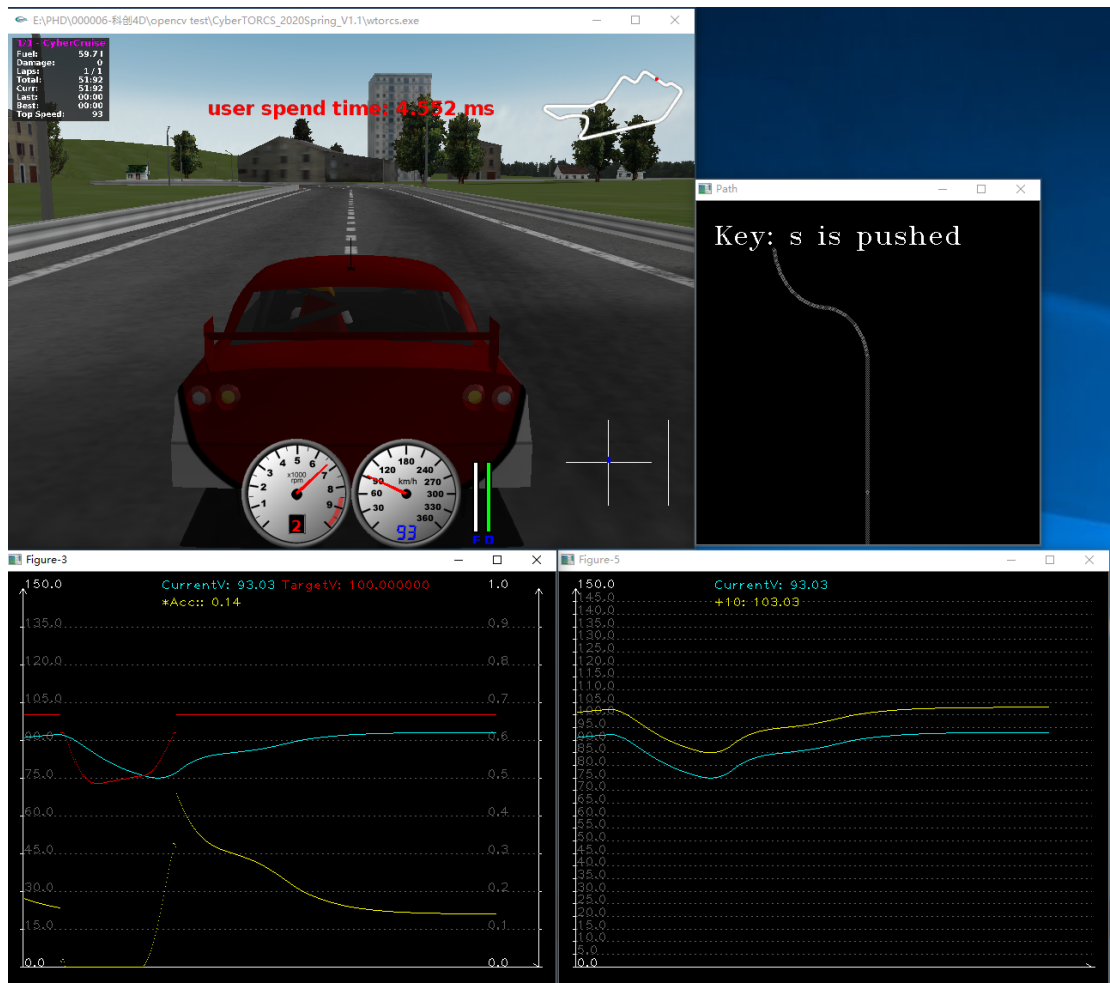
然后在任何全局位置需要声明本插件的类 cls\_VISUAL 以及其他任何需要的全局变量。推荐大家放在 userDriverSetParam 上面。

```
127 //***** Global Variables for OpenCV Visualization *****/
128 // Shiyi Wu 2020.03.17
129 // Under Development
130 cls_VISUAL cls_visual;
131 int nKey = 0;
132 char cKeyName[512];
133 //*****
134
135 static void userDriverSetParam(float* cmdAcc, float* cmdBrake, float* cmdSteer, int* cmdGear) {
```

最后，在 userDriverSetParam 最后有本插件的实体样例。221 行和 222 行是单行绘图语句，别的部分针对有兴趣的同学二次开发使用。

```
213 #pragma region Visual
214 cv::Mat im1Src = cv::Mat::zeros(cv::Size(400, 400), CV_8UC1);
215
216 for (int i = 0; i < 200; i++)
217     cv::circle(im1Src, cv::Point(200 + _midline[i][0] * 2, 400 - _midline[i][1] * 2), 2, cv::Scalar(100, 100, 100));
218 sprintf_s(cKeyName, "Key: %c is pushed", nKey);
219 cv::putText(im1Src, cKeyName, cv::Point(20, 50), cv::FONT_HERSHEY_TRIPLEX, 1, cv::Scalar(255, 255, 255));
220 cv::imshow("Path", im1Src);
221 cls_visual.Fig1V(5, 0, 150, 30, "CurrentV", _speed, "+10", _speed + 10.0, "-10", _speed - 10.0);
222 cls_visual.Fig2V(3, 0, 150, 0, 1, 10, "CurrentV", _speed, "Acc", "cmdAcc", "TargetV", expectedSpeed);
223 int tempKey = cv::waitKey(1);
224 if (tempKey != -1)
225     nKey = tempKey;
226 #pragma endregion
```

开发完成后照常生成 dll 覆盖仿真平台原有位置的即可。  
样例程序效果如下，新增了键盘输入调试功能。



有任何问题欢迎在 QQ 群内找我私信。欢迎大家积极再开发。

吴士一  
2020.3.29