# 工程端部署环视功能问题Lists

# 1.全景俯视视图

**问题 1：环视拼接结果图纹理混乱**

**原因：**

环视拼接模块输入数据格式要求为NV12，目前相机输出数据格式就是NV12，无需再进行数据格式转换；

**解决办法：**

去掉yuv转nv12操作；ZSV\_RunPipeline 接口函数输入数据为当前相机原数据；

**问题2：环视拼接模块耗时异常（600-700ms）**

**原因：**

ZSV\_RunPipeline 接口函数输入数据指针指向了跨线程跨CPU的buffer，导致数据访问速度慢；

**解决办法：**

(1) 初始化时为环视模块输入的四路相机图像数据指针预分配内存，如pt\_mat->pv\_data = malloc(img\_yuv420sp\_bytesize)；

(2) 调用ZSV\_RunPipeline之前，通过memcpy的方式把四路相机图像数据copy到预分配的内存中；

(3) 增加对预分配内参的释放；

# 2.车轮视图

**问题 1：车轮视图纹理混乱**

**原因：**

目前相机输出数据格式就是NV12，无需再进行数据格式转换；

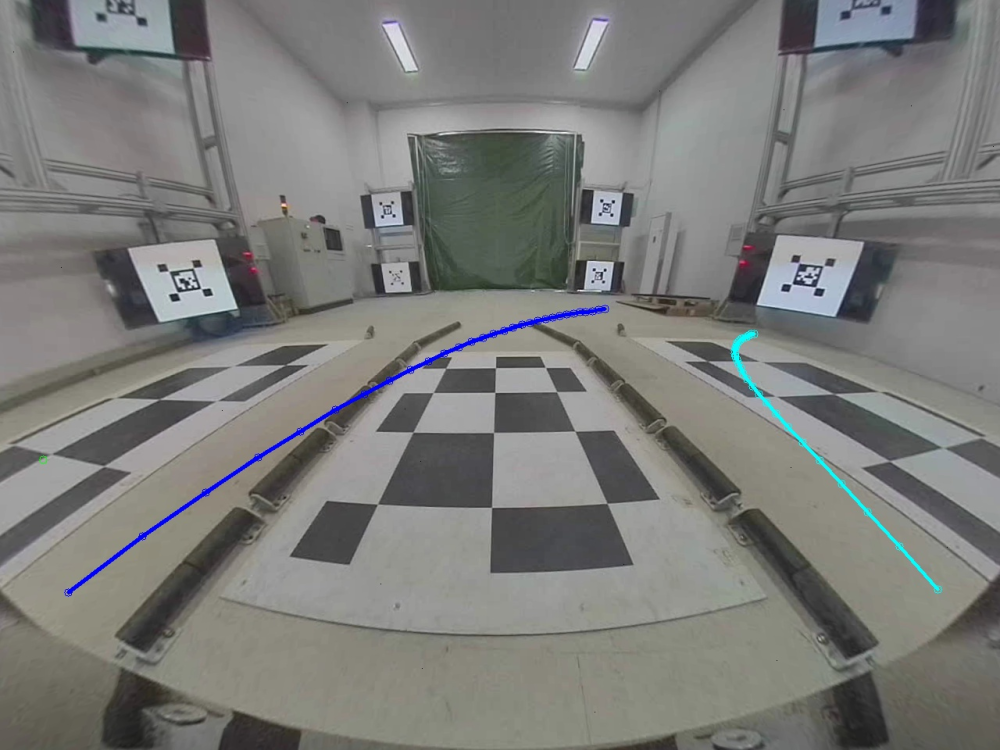
**解决办法：**

进行纹理映射前，去掉yuv转nv12操作；

同时，为保证访问速度，创建buffer，通过memcpy的方式把当前相机数据copy到buffer中，buffer作为纹理映射输入参数；

# 3.轨迹线效果问题

算法输出的是离散的点，之后需要工程侧将其连接为光滑连续的曲线。在单视图对应的轨迹线离散点会出现末端的点越发集中的现象（原理性限制），如下图所示圆圈代表算法产生的点。

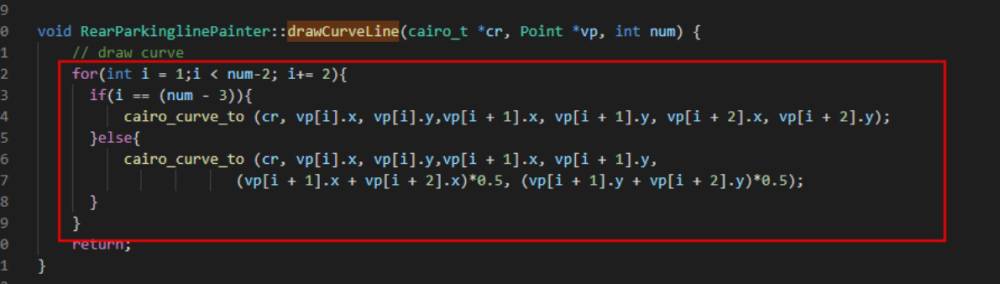


工程侧目前使用cario进行轨迹线绘制，为了使得工程侧绘制的轨迹线获得较好视觉效果，需要对算法产生的点做如下处理。

1）使用后处理策略避免末端的点过于集中，一种简单的策略如下，dis\_threshold用于控制点之间的最小距离。

int num = traj\_lines.lines[n].shape\_points.size();  
  
 int dis\_threshold = 0;  
 std::vector<zsv::Point3f> new\_points;  
  
 for (int i = 0; i < num; i++)  
 {  
 if (i == 0 || i == num - 1)  
 {  
 new\_points.push\_back(traj\_lines.lines[n].shape\_points[i]);  
 continue;  
 }  
 zsv::Point3f last\_pt = new\_points[new\_points.size() - 1];  
 zsv::Point3f cur\_pt = traj\_lines.lines[n].shape\_points[i];  
  
 float dis = sqrt((last\_pt.x - cur\_pt.x)\*(last\_pt.x - cur\_pt.x)  
 + (last\_pt.y - cur\_pt.y)\*(last\_pt.y - cur\_pt.y));  
 if (dis > dis\_threshold)  
 {  
 new\_points.push\_back(cur\_pt);  
 }  
 }

2）将点的数量补足为2n+4的形式（若不够则将最后一个点做复制操作来补充），之后工程侧使用cario的cairo\_curve\_to接口进行线段的绘制，核心操作如下图所示，具体可参考EP33L部分代码。



# 4.车模绘制

2D/3D拼接得到的结果图中存在底部黑色盲区，工程侧会覆盖上车模，一般车模会做一定的放大操作以遮盖住黑色盲区。

标定模块中会输出car\_corners\_id.txt，内容如下所示。其中的第二行数据格式为黑色盲区左上角x世界坐标（单位mm，下同） 左上角y世界坐标 右下角x世界坐标 右下角y世界坐标。由此定义了黑色盲区在世界坐标系下的范围，工程据此结合显示的世界范围，可计算得到车模的缩放比例进行盲区的遮盖

car corners id: 1371 1331 1431 1391  
car box in millimeters: -2641 1018 2641 -1038

# 5.单视图

一般环视中会要求工程侧把**单视图中的后视图做左右镜像处理**，这样人眼观察时符合习惯，即单视图中显示在左侧的东西也位于驾驶员的左侧；

后视图中一般要绘制轨迹线，但是**算法模块输出的轨迹线坐标是以原始视图为基准，不能直接用于镜像后的单视图**

处理流程为：单视图处理模块（如果存在）——绘制轨迹线——单视图+轨迹线整体左右镜像；或者单视图处理模块（如果存在）——单视图左右镜像——镜像绘制轨迹线

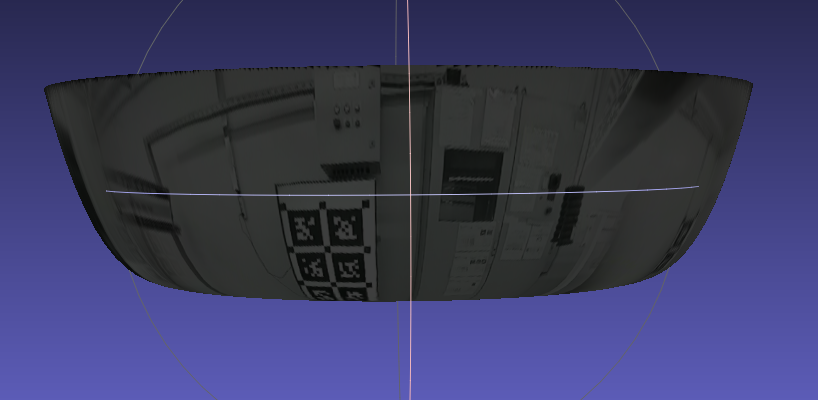
# 6.3D 视图适配

AS33 3D 视图适配至UI时，3D视图默认视角存在黑边；

**措施1：**

修改：zsv/examples/test\_create\_ply.cc ，loop 从37修改为39：

bowl\_mesh\_config.loop = 39;//37;  
  
bowl\_seperate\_mesh\_config.loop = 39;//37;



**措施2：**

修改Bowl高度后，保证3D视图默认视角不出现黑边，但是交互拖动时依旧出现黑边，针对该问题，工程侧限制交互旋转角度；

# 7.广角视图

**需求：**

前后广角的尺寸：

主驾位置，不显示APA时：1500x756

主驾位置，显示APA时：2000x756

副驾位置，不显示APA时：1800x756

副驾位置，显示APA时：不显示广角视图

**措施：**

算法按照1800x756分辨率制作广角PLY模型，工程侧适配时，通过缩放适配2000x756、1500x756；

# 8.环视标定

**问题1**：工程模式下，环视自动标定模块耗时较久，约1min；

**原因**：srvcalib 这个进程是标定进程，没有在sysmonitord里面报备过，没有报备过的进程的话，sysmonitord 会对它进行限制，防止影响其他的进程；

**解决办法**：不进行限制，耗时约9s；

**问题2**：**环视标定相关路径**

* **动态读取的内参的保存路径：**

/etc/adas/srv/res/zsv/oem/default/1280x960/fisheye/camera\_model/

* **环视标定结果的保存路径**

/keyinfo/adas/srv/cfg/res/zsv/alg/

注意：在环视标定模块，需要工程侧在输出目录下创建camera\_model文件夹；标定成功后，工程侧转存标定结果至/keyinfo/adas/srv/cfg/res/zsv/alg/；

* **环视拼接模块的输入文件路径**：

工程部署时，若车机从未登录工程模式执行过环视标定，则环视拼接模块的输入文件路径在：

/etc/adas/srv/res/zsv/oem/SAIC/SMPV\_ROEWE\_AS33/AS33/1280x960/alg/

* 若车机登录工程模式执行过环视标定，则环视拼接模块的输入文件路径在：

/keyinfo/adas/srv/cfg/res/zsv/alg/

工程侧部署环视拼接模块，会先检查/keyinfo/adas/srv/cfg/res/zsv/alg/目录，如果没则使用/etc/adas/srv/res/zsv/oem/SAIC/SMPV\_ROEWE\_AS33/AS33/1280x960/alg/

**优化1：**

环视标定增加了Ceres优化模块后，设置ROI区域降低耗时，工程模式下环视标定耗时约9s；

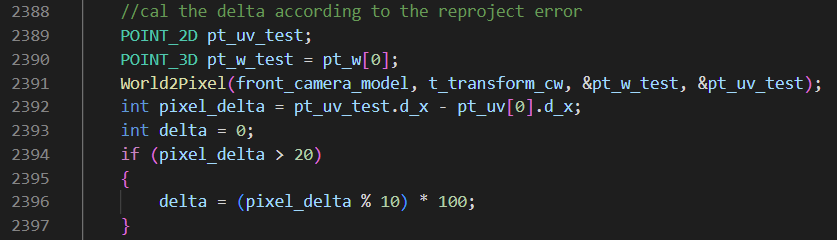
如下图，前视ROI：



**优化2**：

由于IPM 视图默认尺寸的限制，导致侧边棋盘格区域不完整；根据中间角点的重投影误差，设置delta值，动态补偿IPM的宽度；

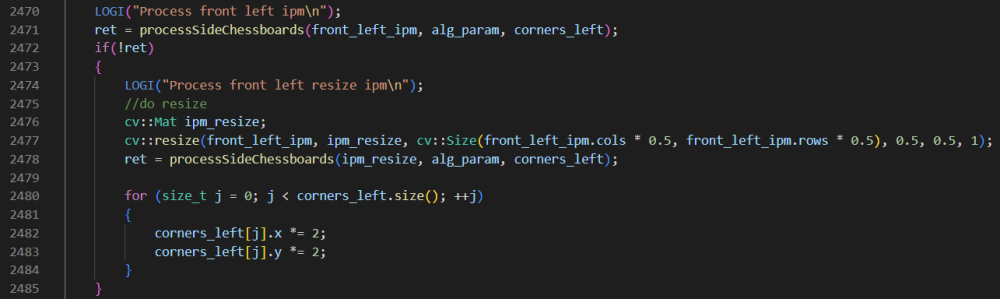
其中，delta值的计算如下：



**优化3：**

由于IPM视图中黑白棋盘格间隔较大，导致角点提取失败；对ipm 图像上left/right 区域进行resize处理，之后再进行角点提取操作；

其中，前视左侧区域处理示例如下：



**优化4：**

调整 car\_box\_margin.txt，避免环视拼接图露出车体部分；

根据相机安装说明图，前视车体遮挡约19cm，后视车体遮挡约7cm，车长4655cm，车宽度1890cm；

根据理论计算，分别为：

0.0408 0 0.015 0

根据减小黑框的原则以及实测的效果，实际设置为：

0.0415 -0.05 0.015 -0.05

**优化5：**

针对此场景左后、右后的清晰度问题，融合区域车尾部淡化消失问题：

首先颜色和清晰度的差异是因为该区域为融合区域、右图和后视图中此区域亮度和清晰度存在差异，调整融合区域使其以后视图为主，提高草坪的清晰的，但会导致该车旁边的车的车尾上部淡化消失（后视图看不到车尾上部，针对该问题之前调测过融合区域），目前只能折中取一套参数，左后、右后以两侧的视图为主，清晰度由两侧的视图清晰度决定，稍微提高下左后、右后的整体亮度，无法保证和中间区域清晰度一样；

修改blending\_parameters.txt为：（AS33项目所用参数）

blending\_width\_3d: 0.3 0.3 0.3 0.3

blending\_angle\_3d: 135 30 -30 -135

blending\_width\_2d: 0.3 0.3 0.3 0.3

blending\_angle\_2d: 135 45 -45 -135