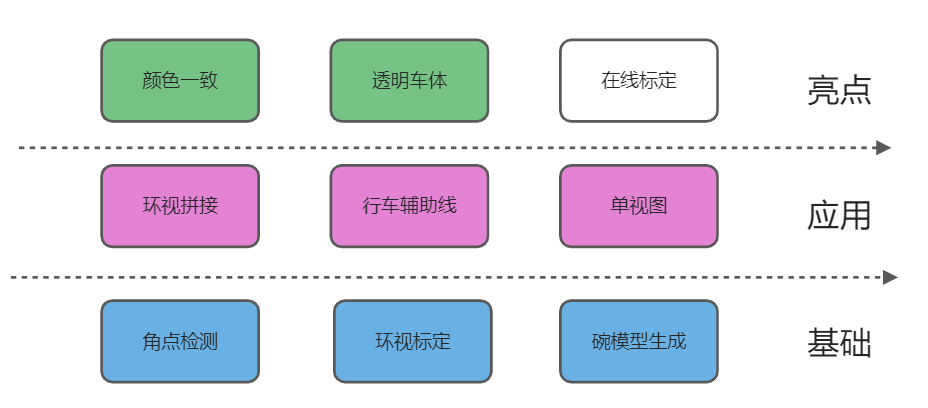
# 整体介绍

# 1.框架

 ZSV工程估计从18年左右开始启动迭代至今，为了适于多人配合开发，采取了解耦的多模块结构；算法按照功能大致分为上述几个模块。

这里又将模块分为基础、应用、亮点三部分；应用需要依赖基础模块的输出结果文件；应用与基础共同构成整体的环视应用；为了提高产品竞争力，会开发一些锦上添花的亮点应用。

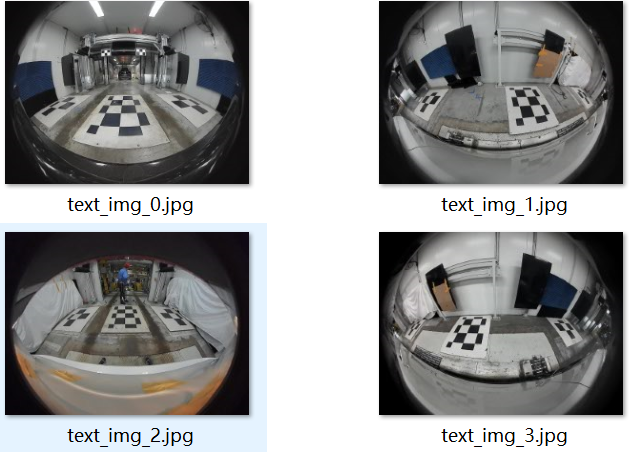
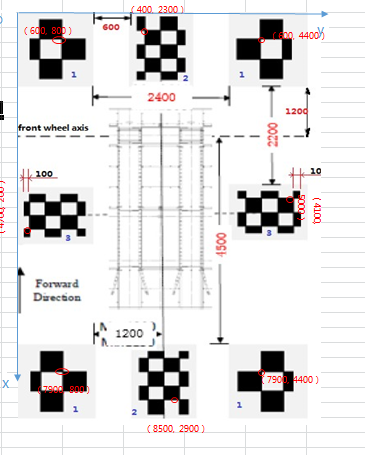
# 2.模块功能

在此之前墙裂建议先阅读学习[基础知识相关书籍/资料推荐](https://yuque.antfin-inc.com/uwk7mr/pc85xr/eq46xa)中的第一篇论文入门。

现简要介绍各个模块的功能与联系，后续篇章会针对各个模块做详细介绍。当前常用模块的调用顺序为：2D/3D模型生成——>标定——>拼接/透明车体；单视图畸变校正——>行车辅助线。

## 标定

首先需要完成环视相机的内外参标定；内参部分目前项目中要求客户提供内参/畸变参数信息，之后转换为工程支持的格式。外参部分则需要1）适配厂商的标定场地，主要体现在角点检测；2）利用1）得到的角点像素坐标、已知的角点世界坐标、相机内参，计算得到外参。左侧图是某项目中的一种标定场地，右侧图是四路相机在场地中的成像，角点检测就是要检测出成像图片中的棋盘格角点像素位置。



角点检测功能demo见test\_corner\_extraction。

之后就是标定部分，利用上述得到的角点像素坐标，和已知的角点世界坐标、相机内参，计算得到相机外参。

在得到相机外参后，我们可实现如下的功能：已知某世界坐标系中的一个点，即世界坐标已知，则可利用上述标定得到的相机外参、相机内参，计算得到其在输入图像中的像素坐标。即知道这个点长什么样，这一点非常重要，是后续拼接的基础。

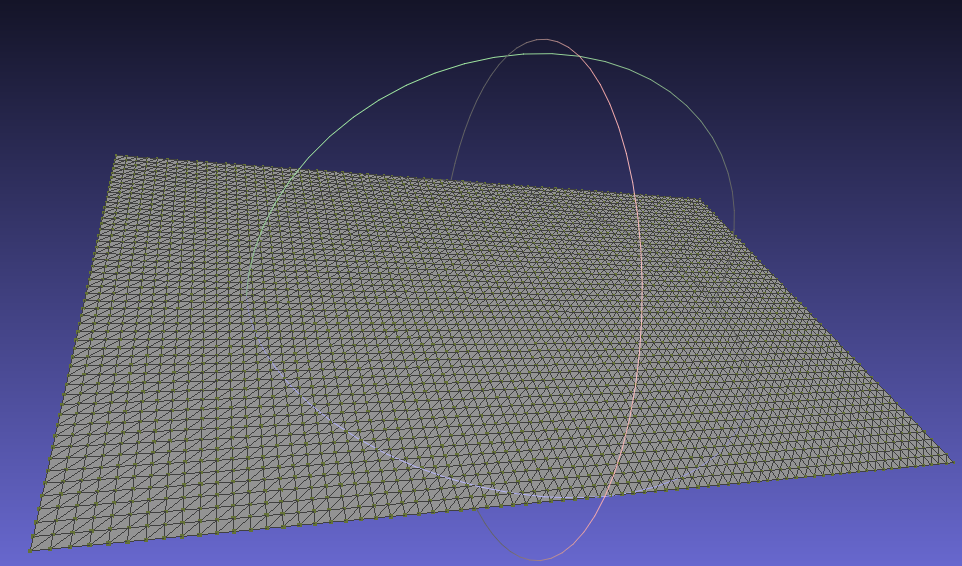
之后需要使用模型生成模块来生成我们想要展示的“场景形状”，再同步生成“场景形状”对应的“纹理”与“纹理坐标”，综合服务与工程中进行渲染。这其中会涉及到下面的模型生成模块与拼接模块。

标定demo见test\_calibration。

## 2D/3D模型生成

如上所述，我们需要先生成想要展示的“场景形状”，目前主要包括2D和3D两种。

2D即假设车辆周边一定范围内的场景都位于地面，即同一平面。此时的“场景形状”也是一个平面，如下图所示。

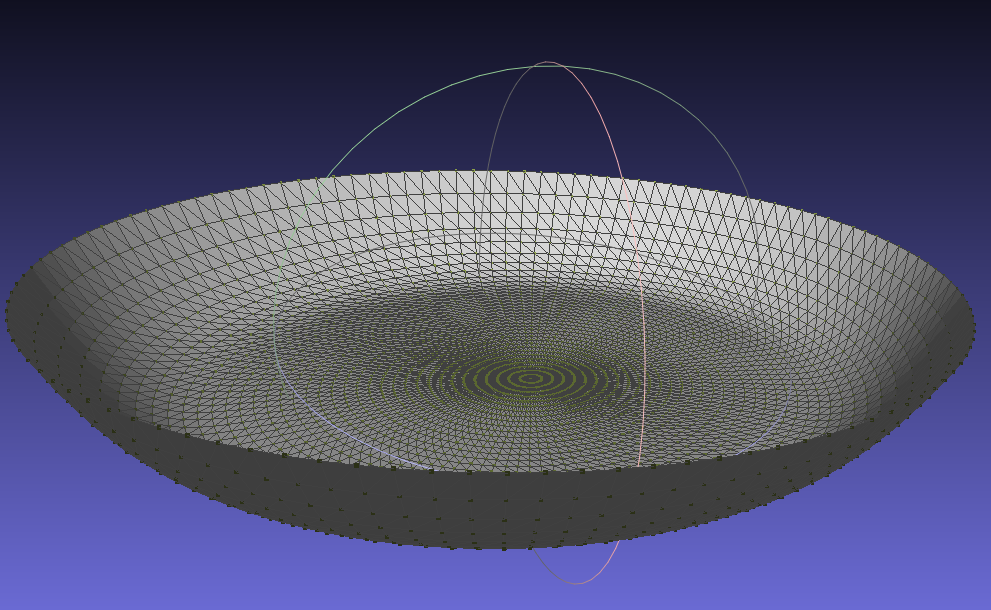


对应拼接的结果我们幅鸟瞰图，一般这样的结果工程端可直接渲染到屏幕，无其他交互操作，如下图所示。

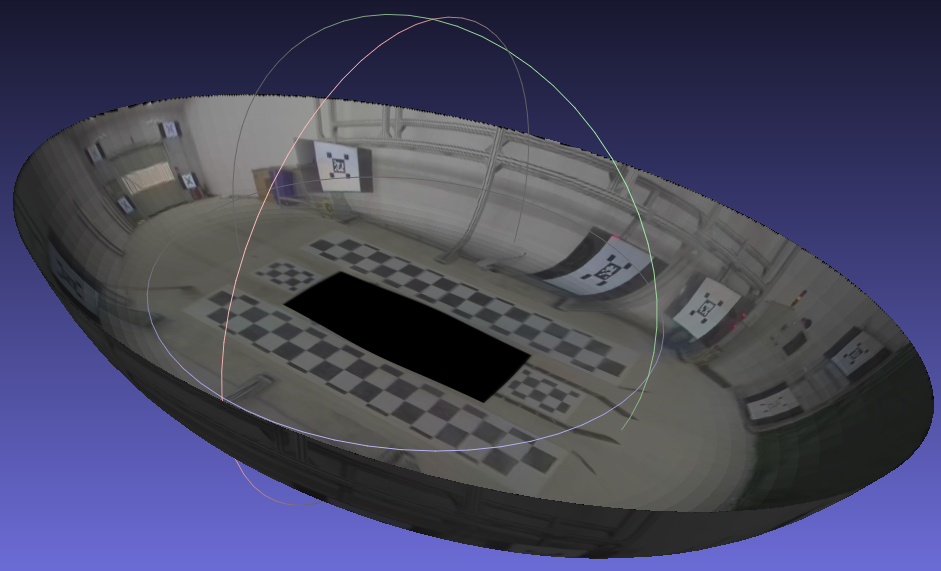


但3D环视中需求为客户可通过触摸屏进行环视视角的切换，实现类似于3D渲染中的漫游操作。

如何实现该功能？算法端不能再像2D中一样只建立平面的映射关系，而需要建立3D的视角和映射关系。这里我们选择生成了一个椭圆形状的3D面片（这是一种经验性的结构构造，符合大部分场景下的实际空间分布，但仍然是一种伪3D，因为其高度分布是我们假设的），如下图所示。



同时在标定过程中会读取上述生成的3D文件，并利用标定得到的相机外参、已知的相机内参，计算上述3D文面片对应的纹理与输入图像之间的映射关系并保存（用于拼接模块），同时保存3D文件面片与纹理图像的映射关系（纹理坐标）。在已知面片点的世界坐标、纹理、纹理坐标下，可利用OpenGL进行渲染，得到的结果如下图。同时可利用OPENGL实现不同视角的渲染，达到漫游的效果。



注意2D/3D模型生成模块只涉及其中的模型生成部分，最终的渲染需要工程端利用该模块与拼接模块的输出结果综合产生。

该模块的demo是test\_create\_ply。

完成标定与拼接之后，在文件夹下会生成一系列的txt与ply文件。其中sv\_bowl\_rendering.ply就是3D环视的渲染效果文件，使用meshlab可打开查看其效果，如上图所示。工程侧会基于opengl实现类似于meshlab中对模型的漫游/缩放操作。

## 拼接

1）在标定过程中会保存2D/3D下结果纹理与原始输入图像的映射关系

2）拼接过程就是读取上述映射关系，使用四路输入图像生成结果纹理图像的过程。

该模块的demo是test\_pipeline。

## 单视图

环视界面中一般会包括单视图，是指对某一路相机原始图像进行一些操作后再进行展示。常见的操作包括：

1）畸变校正，去除原始图像中的畸变

2）裁剪，减小结果图中的畸变观感

目前算法代码支持功能1），demo见test\_undistortview\_plane；功能2）由工程同学自行完成。

## 行车辅助线

算法模块会利用车辆结构信息、相机内外参，计算得到车辆周围某参考点（譬如左前轮）在某个方向盘转角下的运动轨迹的离散点坐标，提供给工程进行绘制，从而给用户用于行车/倒车的辅助。

demo见test\_sv\_trajactory。

## 透明车底

常见环视拼接结果中车底区域不可见，但我们可利用车辆运动信息和历史帧图像数据来“填充”车底区域，实现透明车底的效果。

demo见test\_transparent\_bottom。

## 颜色一致

背景：环视中四个摄像头安装于车辆的不同位置，由于摄像头自身差异、成像环境的差异等原因，导致临近的摄像头成像存在色彩上的差异，最终使得拼接结果图像出现颜色/亮度不一致的情况。

目前模块功能由[@李文学(liwenxue.lwx)](https://yuque.antfin-inc.com/liwenxue.lwx)开发完成，尚未合入到主分支。

## 在线标定

背景：车辆使用中会出现一些非人为因素导致环视摄像头位姿发生一定变化，导致环视拼接错位；传统做法是需要将车辆送至4S店做售后标定，费时费力。希望能够实现利用车辆运动时的自然场景来重新计算相机位姿，免去用户需要去4S店的麻烦。

待开发。

# 3.项目情况（更新至2022年4月）

1）AS28项目；2D环视，无透明车体，目前试生产中

2）EP33L（上汽智己）；2D环视+透明车体，目前版本迭代中，尚未锁版

3）AS33项目；3D环视+透明车体；算法侧完成适配，待工程完成初版后实车测试

4）捷达：明确PRD需求，新增cross traffic视角、标定算法功能开发中