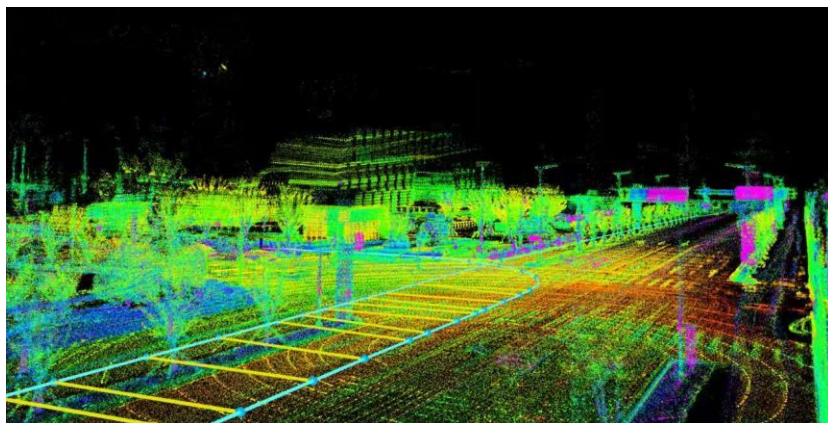




微信扫码加入星球

# 自动驾驶中实战之Lidar与Radar融合

## Camera + LiDAR + Radar + IMU



主 讲 人：爱喝苦咖啡的小阿飞

公 众 号：3D 视 觉 工 坊

# 内容

一、整体结构

二、要点讲解

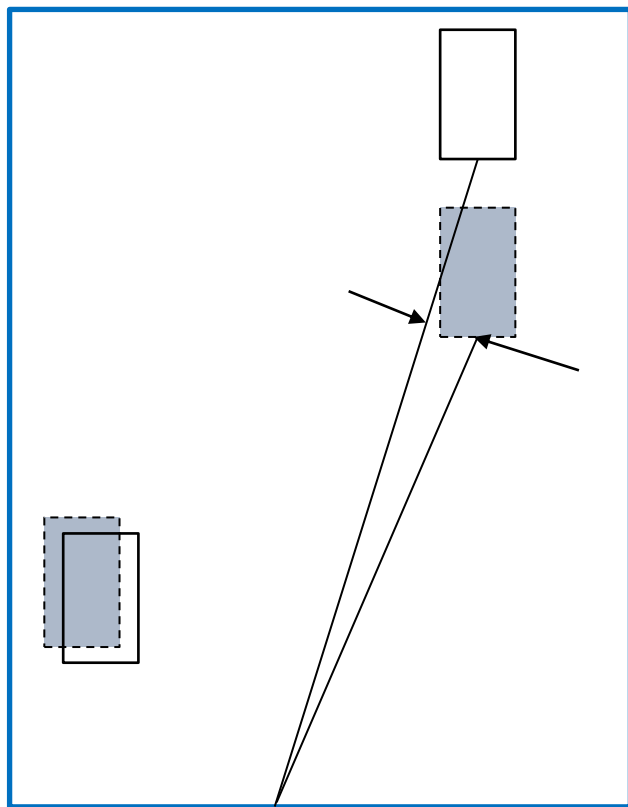
三、如何优化

四、作业讲解

五、仍未结束



见视频

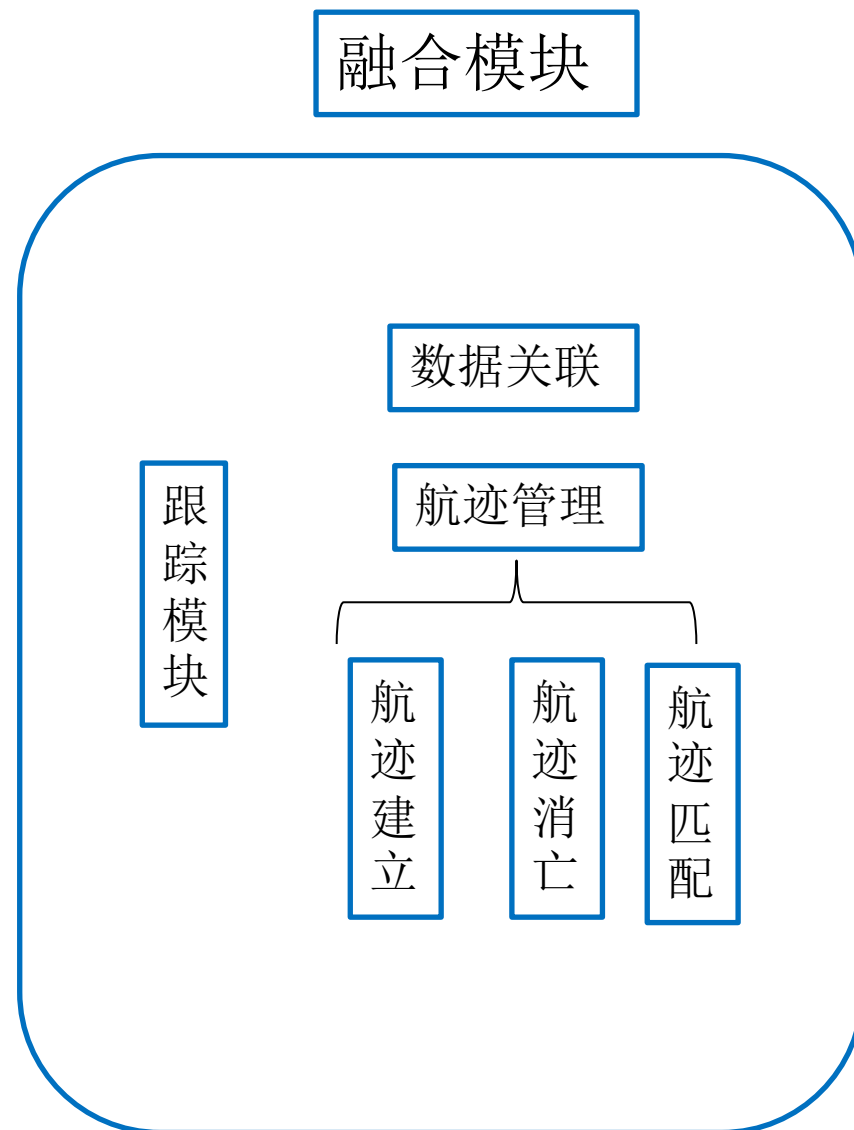
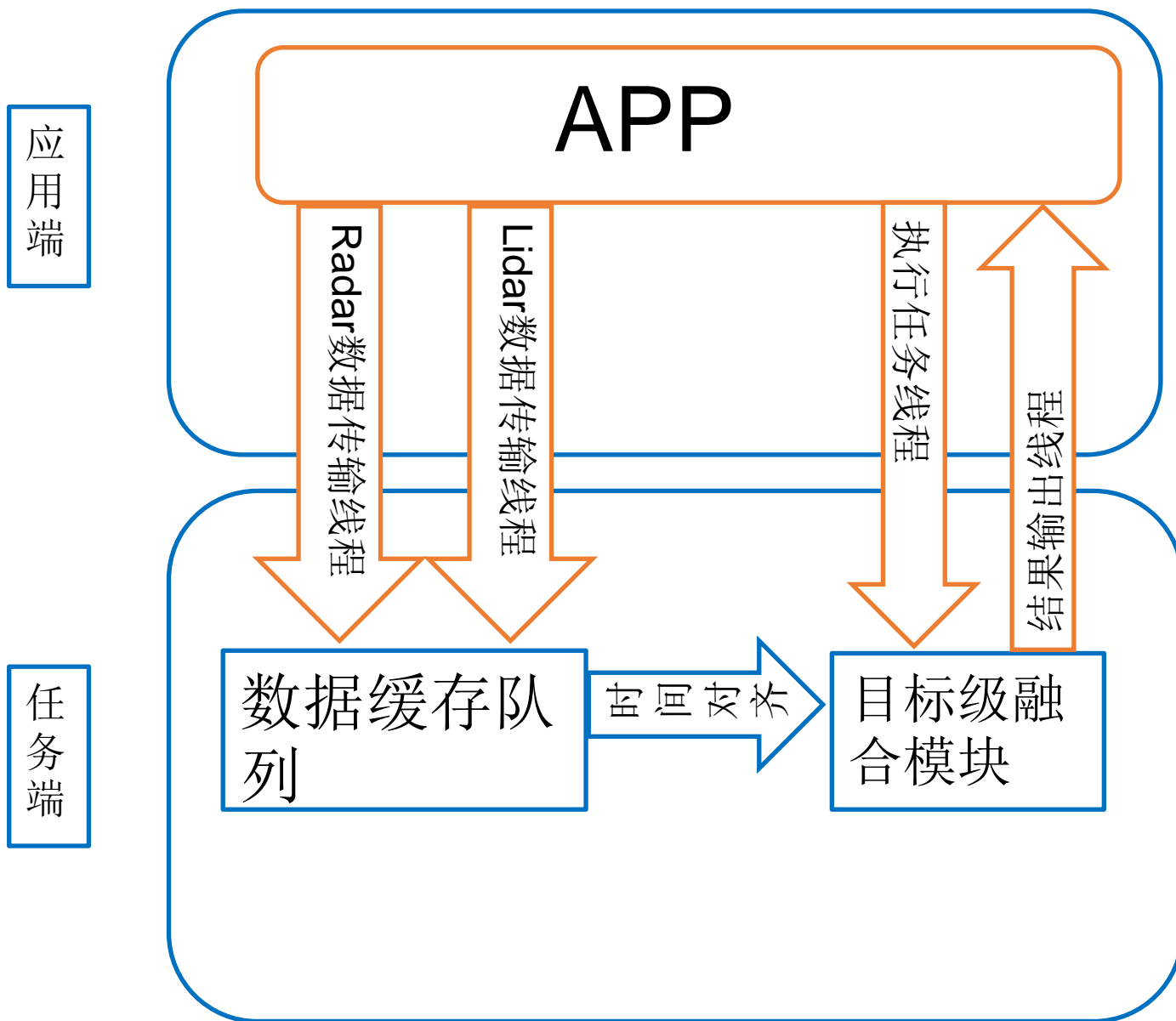


0.0	0.8	0.0
0.1	0.3	0.9
0.8	0.0	0.2

radar

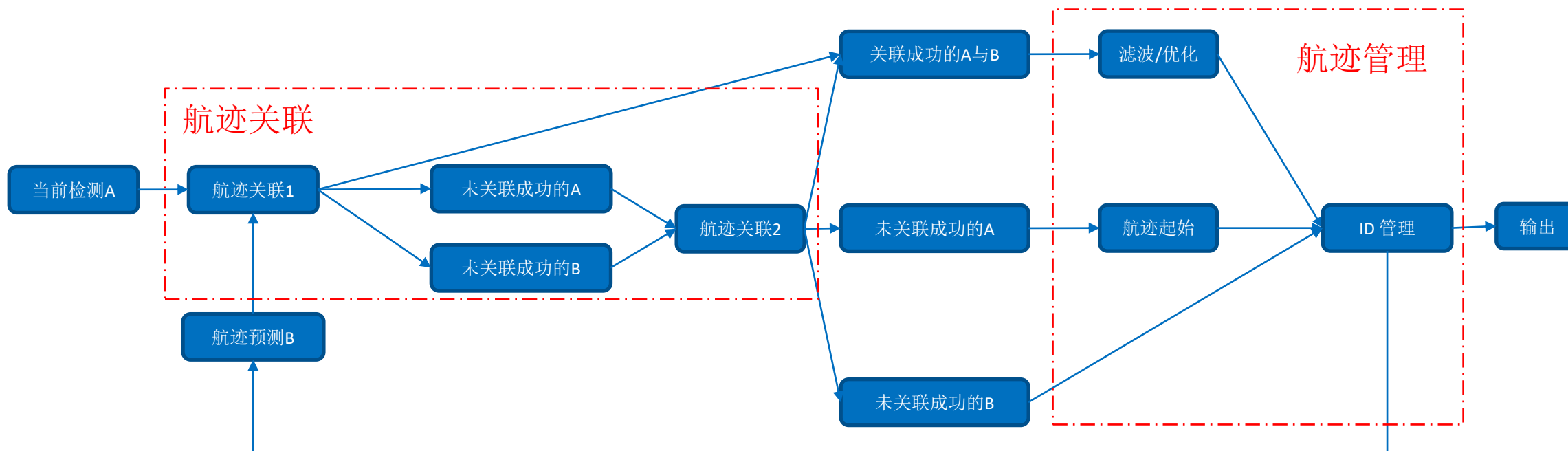
lidar

代价矩阵





跟踪架构主要可分为两部分：一部分为航迹关联，一部分为航迹管理。

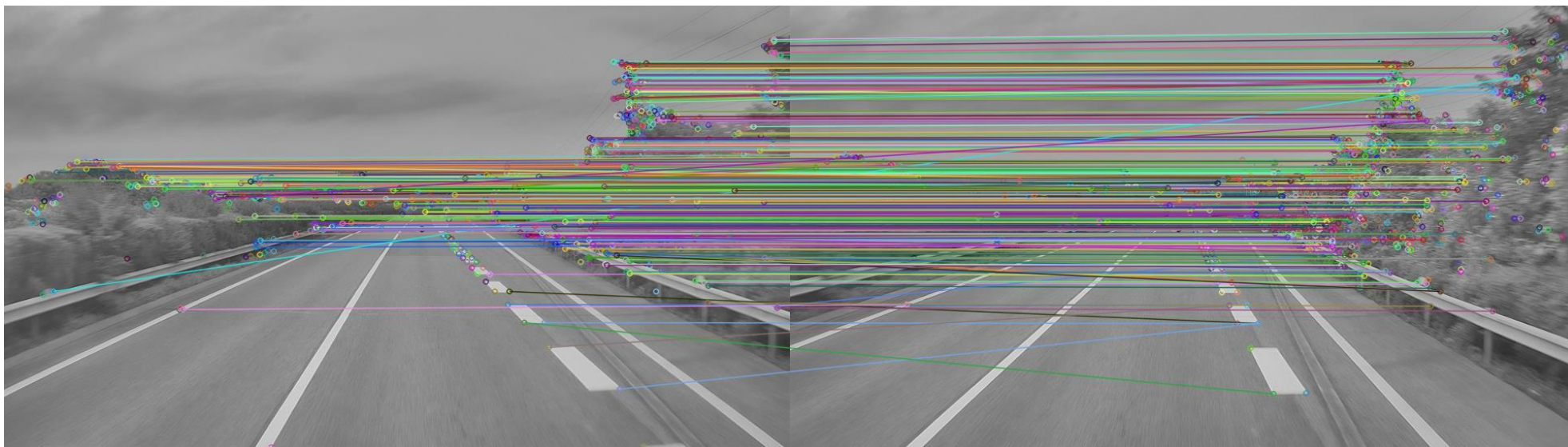


多目标跟踪架构



## 基于外观的图像域跟踪





1. 请在camera\_camera\_sync文件下代码中添加合适的代码生成上述图像。

```
findGoodMatch(matches, good_matches);  
  
Mat imgMatch;  
drawMatches(img1, keypoints1, img2, keypoints2, good_matches, imgMatch);  
  
// imshow("imgMatch", imgMatch);  
// waitKey(2);  
cv::imwrite("/home/match.png", imgMatch);  
  
for(int i = 0; i < good_matches.size(); i++)  
{
```

```
void findGoodMatch(std::vector<DMatch> matches, std::vector<DMatch> &good_matches)  
{  
    int sz = matches.size();  
    double max_dist = 0, min_dist = 50;  
  
    for(int i=0; i < sz; i++)  
    {  
        double dist = matches[i].distance;  
        if (dist < min_dist) min_dist = dist;  
        if (dist > max_dist) max_dist = dist;  
    }  
  
    for (int i = 0; i < sz; i++)  
    {  
        if(matches[i].distance < 0.5 * max_dist)  
        {  
            good_matches.push_back(matches[i]);  
        }  
    }  
}
```

2. 调整合适的阈值使得左右特征点匹配相对较好。
3. 如何优化相机间的yaw角？





补充双目相机间优化Yaw角代码；

补充双目相机间测距目标匹配方式；

补充Lidar与Radar基于优化的融合方法；

补充激光雷达到RTK（IMU）外参标定方法；



购买该课程请扫描二维码



微信扫码加入星球



**感谢聆听**

Thanks for Listening