从输出维度的角度来看，基于视觉传感器的感知方法可以分为2D感知和3D感知两种。（相关阅读：一文读懂2D感知算法）

从传感器的数量上看，视觉感知系统也分为单目系统，双目系统，以及多目系统。2D感知任务通常采用的是单目系统，这也是计算机视觉和深度学习结合最紧密的领域。但是自动驾驶感知最终需要的是3D输出，因此我们需要将2D的信息推广到3D。

在深度学习取得成功之前，通常的做法是根据目标的先验大小以及目标处于地平面上等假设来推断目标的深度（距离），或者采用运动信息进行深度估计（Motion Stereo）。有了深度学习的助力之后，从大数据集中学习场景线索，并进行单目深度估计成为了可行的方案。但是这种方案非常依赖于模式识别，而且很难处理数据集之外的场景（Corner Case）。比如施工路段的特殊工程车辆，由于数据库中很少出现或者根本没有此类样本，视觉传感器无法准确检测该目标，因而也就无法判断其距离。双目系统可以自然的获得视差，从而估计障碍物的距离。这种系统对模式识别的依赖度较小，只要能在目标上获得稳定的关键点，就可以完成匹配，计算视差并估计距离。但是，双目系统也有以下缺点。

首先，如果关键点无法获取，比如在自动驾驶中经常引发事故的白色大货车，如果其横在路中央，视觉传感器在有限的视野中很难捕捉关键点，距离的测算就会失败。

其次，双目视觉系统对摄像头之间的标定要求非常高，一般来说都需要有非常精确的在线标定功能。

最后，双目系统的计算量较大，需要算力较高的芯片来支持，一般都会采用FPGA。

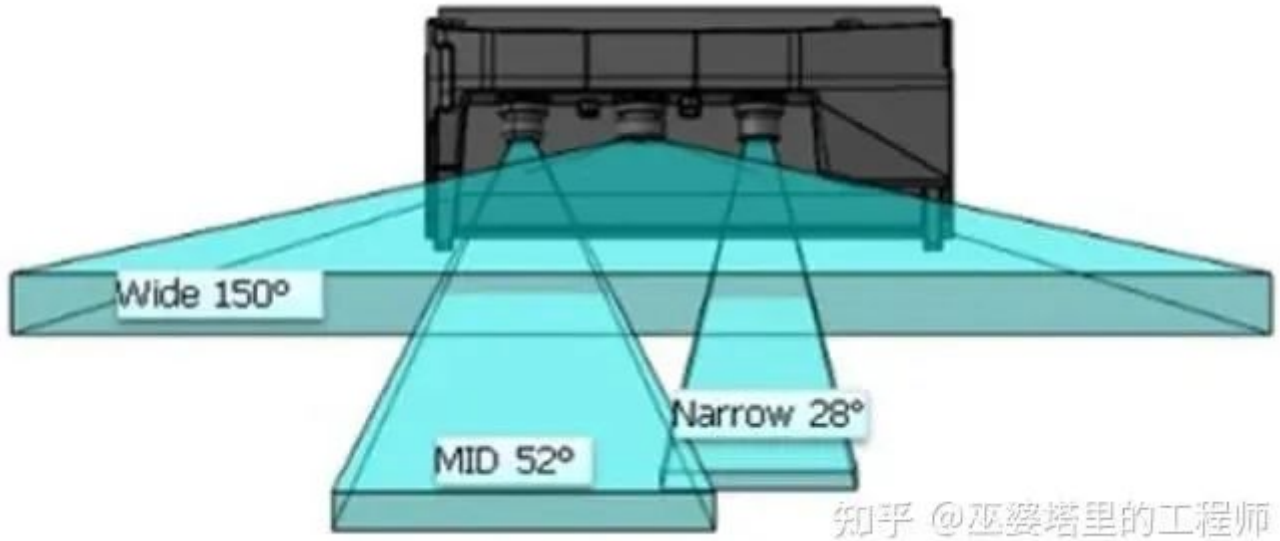
双目系统的成本介于单目和激光雷达之间，目前也有一些OEM开始采用双目视觉来支持不同级别的自动驾驶系统，比如斯巴鲁，奔驰，宝马等。理论上说，双目系统已经可以解决3D信息获取的问题，那么为什么还需要多目系统呢？

原因大致有两点：一是通过增加不同类别的传感器，比如红外摄像头，来提高对各种环境条件的适应性；二是通过增加不同朝向，不同焦距的摄像头来扩展系统的视野范围。下面我们就来分析几个典型的多目系统。

Mobileye的三目系统

对应定焦镜头来说，探测距离和探测视角是成反比的关系。视角越宽，探测的距离越短，精度越低；视角越窄，探测的距离越长，精度越高。车载摄像头很难做到频繁变焦，因此一般来说探测距离和视野都是固定的。

多目系统，可以通过不同焦距的摄像头来覆盖不同范围的场景。比如Mobileye和ZF联合推出的三目系统，三目包含一个150°的广角摄像头，一个52°的中距摄像头和一个28°的远距摄像头。其最远探测距离可以达到300米，同时也可以保证中近距的探测视野和精度，用于检测车辆周边的环境，及时发现车辆前方突然出现的物体。



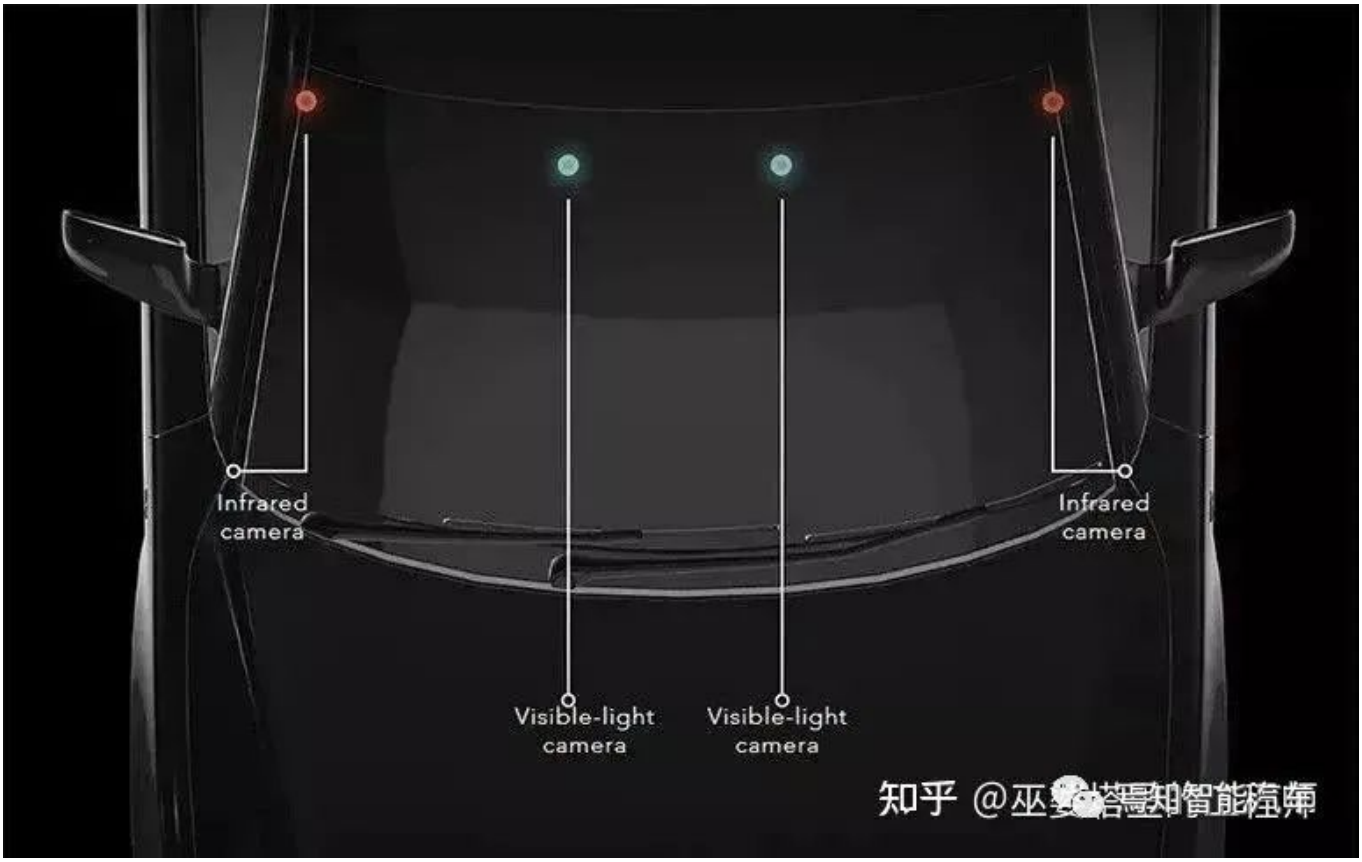
Mobileye和ZF的三目相机

这种三目系统主要的难点在于如何处理重叠区域中不一致的感知结果。不同摄像头对于同一场景给出了不同的理解，那么就需要后面的融合算法来决定信任哪一个。不同摄像头自身的误差范围也不同，很难设计一个合理的规则去定义各种不同情况下的决策，这给融合算法带来了更大的挑战。文章后面会介绍，多目系统其实还可以采用数据层的融合，利用深度学习和大数据集来学习融合规则。当然也不是说交给机器学习就完事大吉了，黑盒子的深度神经网络有时也会给出难以解释的输出。

Foresight的四目感知系统

多目系统的另外一个思路是增加不同波段的传感器，比如红外摄像头（其实激光雷达和毫米波雷达也是不同波段的传感器而已）。来自以色列的Foresight公司设计并演示了一个四目感知系统（QuadSight）。在可见光双目摄像头的基础上，QuadSight增加了一对长波红外（LWIR）摄像头，使探测范围从可见光波段扩展到红外波段。红外波段的加入，一方面增加了信息量，另一方面也增强了在夜间环境以及在雨雾天气下的适应能力，保证了系统全天候运行的能力。

QuadSight系统中摄像头的视野范围为45度，最远可以探测150米的距离，可以在100米的距离内探测到35*25厘米大小的物体。运行速度方面可以达到45帧/秒，足以应对高速行驶的场景。

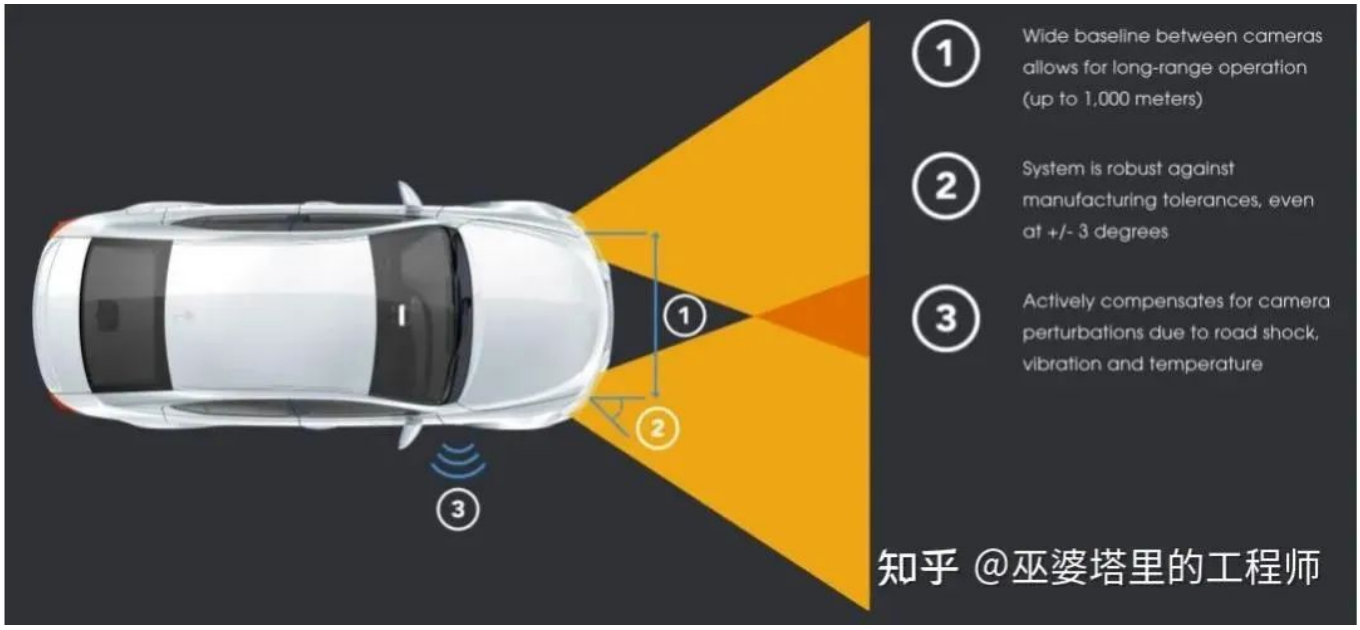


Foresight的QuadSight四目系统

QuardSight系统是由两对双目系统组成。从上图中可以看到，红外双目摄像头安装在挡风玻璃的左右两侧，其基线长度要比一般的双目系统大很多。这里稍微跑点题，讨论一下双目系统基线长度的问题。

传统的双目系统一般采用短基线模式，也就是说两个摄像头之间的距离比较短，这就限制了探测的最大距离。当一个目标距离很远时，其在左右图像上的视差已经小于一个像素，这时就无法估计其深度，既所谓的基线约束。这已是极限的情况，其实对于远距离目标，即使视差大于一个像素，深度估计的误差也是很大的。一般来说，深度估计的误差应该与距离的平方成正比。

为了提高双目系统的有效探测距离，一个直观的方案就是增加基线长度，这样可以增加视差的范围。NODAR的公司推出的Hammerhead技术，可以实现两个摄像头超大距离的宽基线配置，探测距离最远可达1000米，同时可以生成高密度的点云。这个系统可以利用整车的宽度，比如把摄像头安装在侧视镜、前大灯或车顶两侧。



Hammerhead技术中的宽基线配置

Tesla的全景感知系统

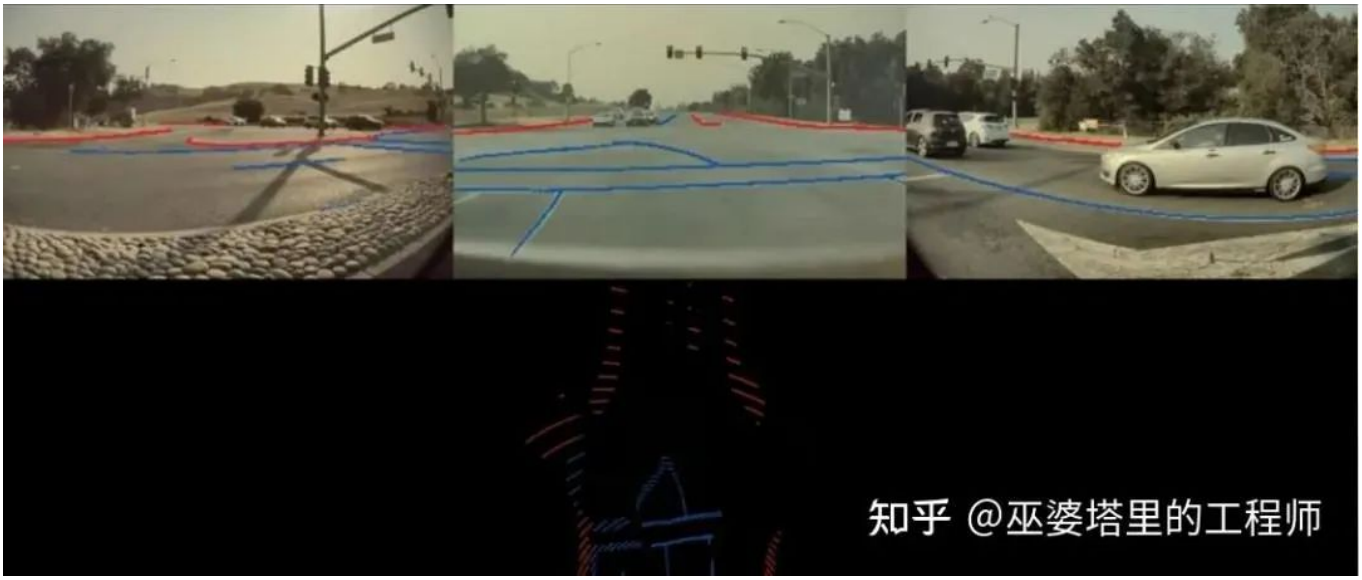
分析了三目和四目的例子后，下面进入本篇文章的重点，也就是基于多目的全景感知系统。这里我们采用的例子是Tesla在2021年的AI Day上展示了一个纯视觉的FSD（Full Self Driving）系统。虽然说只能算是L2级别（驾驶员必须做好随时接管车辆的准备），但如果只是横向对比L2级的自动驾驶系统，FSD的表现还是不错的。此外，这个纯视觉的方案集成了近年来深度学习领域的很多成功经验，在多摄像头融合方面很有特点，个人觉得至少在技术方面还是值得研究一下。

Tesla FSD系统的多摄像头配置

这里再稍微跑个题，说一下Tesla AI和Vision方向的负责人，Andrej Karpathy。这位小哥1986年出生，2015年在斯坦福大学获得博士学位，师从计算机视觉和机器学习界的大牛李飞飞教授，研究方向是自然语言处理和计算机视觉的交叉任务以及深度神经网络在其中的应用。马斯克2016年将这位青年才俊召入麾下，之后让其负责Tesla的AI部门，是FSD这个纯视觉系统在算法方面的总设计师。

Andrej在AI Day上的报告中首先提到，五年前Tesla的视觉系统是先获得单张图像上的检测结果，然后将其映射到向量空间（Vector Space）。这个“向量空间”是报告中的核心概念之一，我理解其实它就是环境中的各种目标在世界坐标系中的表示空间。比如对于物体检测任务，目标在3D空间中的位置，大小，朝向，速度等描述特性组成了一个向量，所有目标的描述向量组成的空间就是向量空间。视觉感知系统的任务就是将图像空间中的信息转化为向量空间中的信息。这可以通过两种方法来实现：一是先在图像空间中完成所有的感知任务，然后将结果映射到向量空间，最后融合多摄像头的结果；二是先将图像特征转换到向量空间，然后融合来自多个摄像头的特征，最后在向量空间中完成所有的感知任务。

Andrej举了两个例子，说明为什么第一种方法是不合适的。首先，由于透视投影，图像中看起来不错的感知结果在向量空间中精度很差，尤其是远距离的区域。如下图所示，车道线（蓝色）和道路边缘（红色）在投影到向量空间后位置非常不准，无法用支持自动驾驶的应用。



图像空间的感知结果(上)及其在向量空间中的投影(下)

其次，在多目系统中，由于视野的限制，单个摄像头可能无法看到完整的目标。比如在下图的例子中，一辆大货车出现在了一些摄像头的视野中，但是很多摄像头都只看到了目标的一部分，因此无法根据残缺的信息做出正确的检测，因此后续的融合效果也就无法保证。这其实是多传感器决策层融合的一个一般性问题。

单摄像头受限的视野

综合以上分析，图像空间感知+决策层融合并不是一个很好的方案。直接在向量空间中完成融合和感知可以有效地解决以上问题，这也是FSD感知系统的核心思路。为了实现这个思路，需要解决两个重要的问题：一个是如何将特征从图像空间变换到特征空间，另一个是如何得到向量空间中的标注数据。

特征的空间变换

对于特征的空间变换问题，专栏之前在3D感知的文章中也做了介绍，一般性的做法就是利用摄像头的标定信息将图像像素映射到世界坐标系。但这是个病态问题，需要有一定的约束，自动驾驶应用

中通常采用的是地平面约束，也就是目标位于地面，而且地面是水平的。这个约束太强了，在很多场景下无法满足。

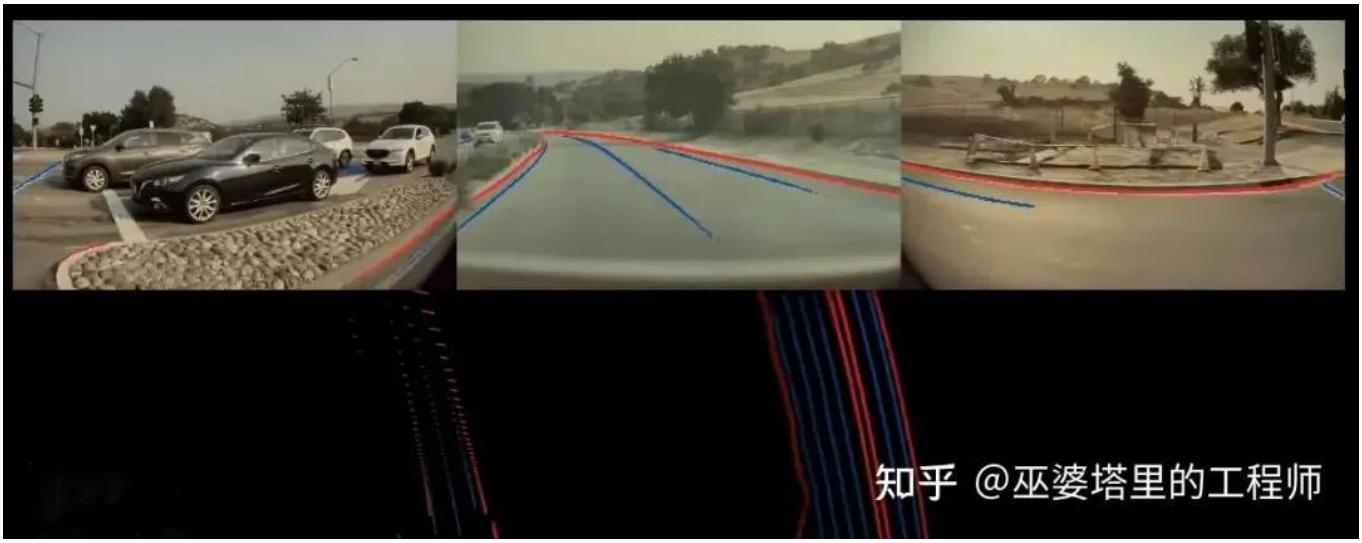
Tesla的解决方案中核心的有三点。

首先，通过Transformer和Self-Attention的方式建立图像空间到向量空间的对应关系，这里向量空间的位置编码起到了很重要的作用。具体实现细节这里就不展开说了，以后有时间再单开一篇文章详细的介绍。简单来理解的话，向量空间中每一个位置的特征都可以看作图像所有位置特征的加权组合，当然对应位置的权重肯定大一些。但是这个加权组合的过程通过Self-Attention和空间编码来自动的实现，不需要手工设计，完全根据需要完成的任务来进行端对端的学习。

其次，在量产应用中，每一辆车上摄像头的标定信息都不尽相同，导致输入数据与预训练的模型不一致。因此这些标定信息需要作为额外的输入提供给神经网络。简单的做法可以将每个摄像头的标定信息拼接起来，通过MLP编码后再输入给神经网络。但是，一个更好的做法是将来自不同摄像头的图像通过标定信息进行校正，使不同车辆上对应的摄像头都输出一致的图像。

最后，视频（多帧）输入被用来提取时序信息，以增加输出结果的稳定性，更好的处理遮挡场景，并且预测目标的运动。这部分还有一个额外的输入就是车辆自身的运动信息（可以通过IMU获得），以支持神经网络对齐不同时间点的特征图。时序信息的处理可以采用3D卷积，Transformer或者RNN。FSD的方案中采用的是RNN，以我个人的经验来看，这确实也是目前在准确度和计算量之间平衡度最好的方案。

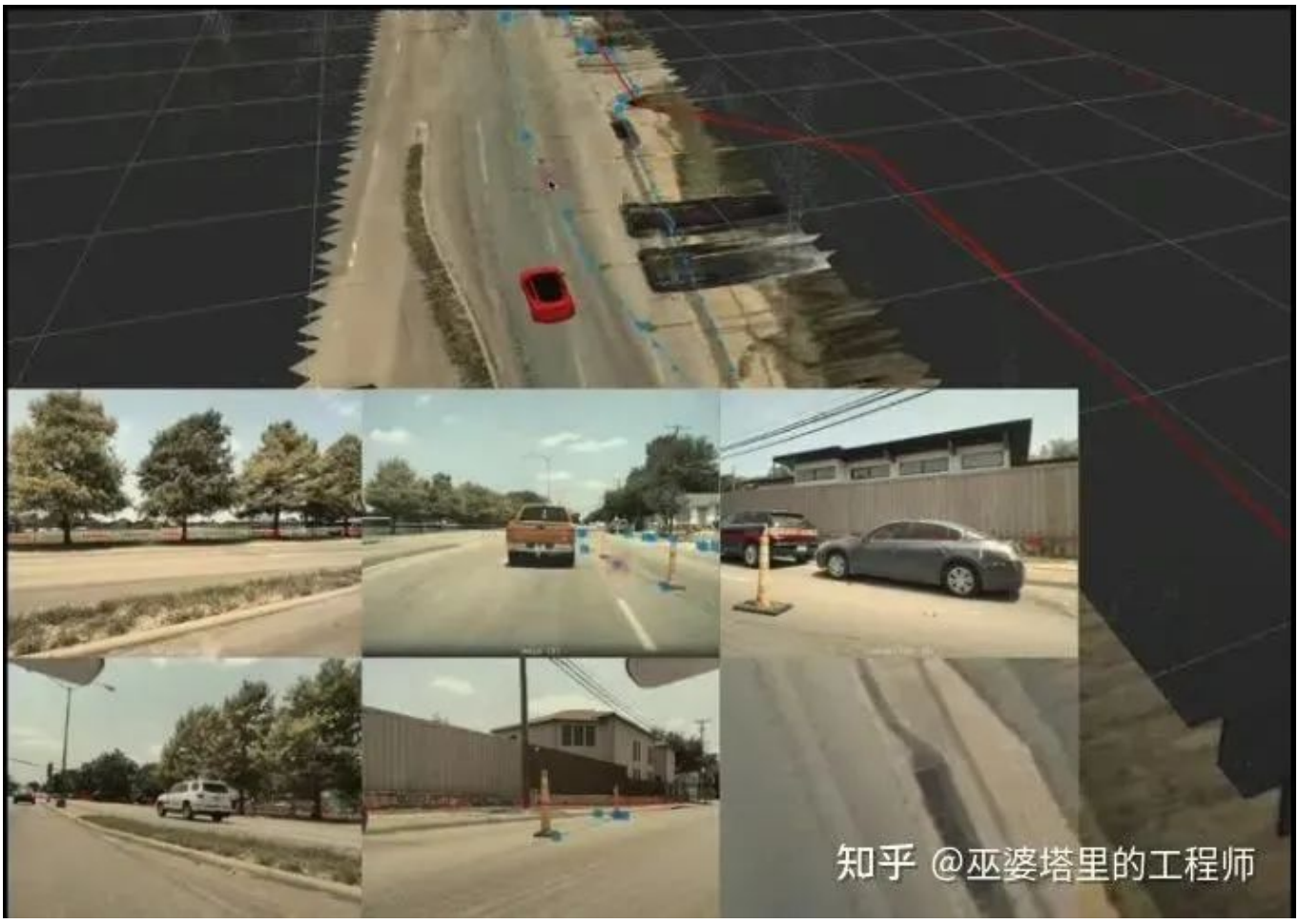
通过以上这些算法上的改进，FSD在向量空间中的输出质量有了很大的提升。在下面的对比图中，下方左侧是来自图像空间感知+决策层融合方案的输出，而下方右侧上述特征空间变换+向量空间感知融合的方案。



图像空间感知（左下） vs. 向量空间感知（右下）

向量空间中的标注

既然是深度学习算法，那么数据和标注自然就是关键环节。图像空间中的标注非常直观，但是系统最终需要的是在向量空间中的标注。Tesla的做法是利用来自多个摄像头的图像重建3D场景，并在3D场景下进行标注。标注者只需要在3D场景中进行一次标注，就可以实时的看到标注结果在各个图像中的映射，从而进行相应的调整。



3D空间中的标注

人工标注只是整个标注系统的一部分，为了更快更好的获得标注，还需要借助自动标注和模拟器。自动标注系统首先基于单摄像头的图像生成标注结果，然后通过各种空间和时间的线索将这些结果整合起来。形象来说就是各个摄像头凑在一起讨论出一个一致的标注结果。除了多个摄像头的配合，在路上行驶的多台Tesla车辆也可以对同一个场景的标注进行融合改进。当然这里还需要GPS和IMU传感器来获得车辆的位置和姿态，从而将不同车辆的输出结果进行空间对齐。自动标注可以解决标注的效率问题，但是对于一些罕见的场景，比如报告中所演示的在高速公路上奔跑的行人，还需要借助模拟器来生成虚拟数据。以上所有这些技术组合起来，才构成了Tesla完整的数据收集和标注系统。

-- END --