

综述：一文详解50多种多模态图像融合方法

原创

CV开发者都爱看的

极市平台

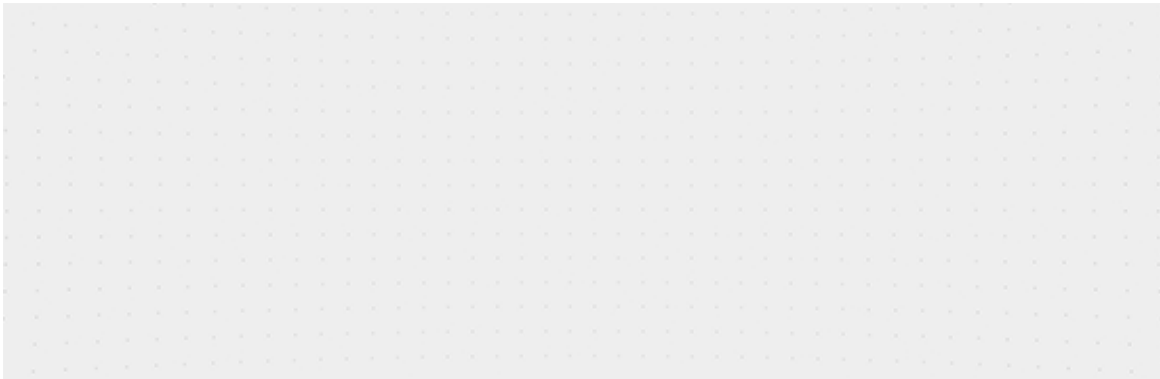
2022-08-25 22:00:41

发表于广东

手机阅读

𠄎

↑ 点击蓝字 关注极市平台



作者 | matrix明仔

编辑 | 极市平台

极市导读

本工作总结了50篇论文中Lidar和camera的多模态融合的一些概念方法。笔者结合原文以及自己的思考 and 解读，希望能给大家够带来关于多模态图像融合的未来一些新思考～ >>加入极市CV技术交流群，走在计算机视觉的最前沿

Multi-modal Sensor Fusion for Auto Driving Perception: A Survey

Keli Huang*
University of California
Los Angeles, US
kelihuang@cs.ucla.edu

Botian Shi
Shanghai AI Lab
Shanghai, China
shibotian@pjlab.org.cn

Xiang Li
Beijing Institute of Technology
Beijing, China
3120211007@bit.edu.cn

Xin Li
East China Normal University
Shanghai, China
51194506020@stu.ecnu.edu.cn

Siyuan Huang
Shanghai AI Lab
Shanghai, China
huangsiyuan@pjlab.org.cn

Yikang Li
Shanghai AI Lab
Shanghai, China
liyikang@pjlab.org.cn

文章链接：<https://arxiv.org/abs/2202.02703>

说在前面的话：

多模态感知融合是自动驾驶的基础任务，吸引了许多“磕盐”人员的关注。但是，由于原始数据噪声大、信息利用率低以及多模态传感器未对齐等这些原因，要想实现一个好的性能也并非易事。所以在这篇调研报告里面，总结了50篇论文中Lidar和camera的多模态融合的一些概念方法，希望通过我的这波翻译+解读，能给大家够带来关于多模态图像融合的未来一些新思考。

（然后这次的分享其实更多是介绍和总结，我尽量通过自己理解还有一些经验把内容进行整理和细化，这是一版能够作为新手入门，也能作为搬砖老手回顾的内容，如果大家觉得内容不错的话可以分享给身边的伙伴们！）

为啥需要多模态融合

在复杂的驾驶环境中，单一的传感器信息不足以有效的处理场景的变化。比如在极端恶劣天气中（大暴雨、沙尘暴）能见度较低的情况下，此时只依靠camera的所反馈的RGB图像完全没有办法对环境的变化做出反馈。而在普通的道路环境中，如红绿灯、色锥等，只依靠Lidar的信息也是无法进行有效识别的，也需要结合camera所带来的RGB信息，才能有效的处理。因此，在自动驾驶感知场景的任务中，不同模态信息的互补会更加的重要。

有什么特征的融合的方法

多模态融合的能用的场景有很多，比如2D/3D的目标检测、语义分割，还有Tracking任务。在这些任务中，重中之中就是模态之间的信息交互融合的工作。从传感器的的信息获取越来越高效精确，成本被压缩得越来越低，自动驾驶中感知任务中的多模态融合方法得到了快速发展的机遇。所以，紧接着来的问题就是，我们到底应该怎么做才能使得多模态融合的工作更加的丝滑和高效呢？

融合的类型

根据50篇论文的统计分析结果，大多数方法遵循将其分为早期（前）融合、特征融合和后融合三大类的传统融合规则。重点关注深度学习模型中融合特征的阶段，无论是数据级、特征级还是建议级。首先，这种分类法没有明确定义每个级别的特征表示。其次，我们一般的方法是对激光雷达和相机的数据信息是开两个分支，在模型的处理过程中两个模态的分支始终是保持对称的，从而使得得两个模态的信息可以在同一特征等级下进行交互。综上所述，传统的分类法可能是直观的，但对最近出现的越来越多的多模态融合的内容，按照传统的理解合分类方式，不足以应付！

两大类和四小类融合方式

最新的融合任务为自动驾驶感知任务提出了一些创新的多模式融合方法。总的来说包括了两大类：即强融合和弱融合，以及强融合中的四个小类，即早期（前）融合、深度（特征）融合、

后期（后）融合、不对称融合（这个表示两个分支的特征进行相互决策）

各种任务以及数据集的介绍

一般来说多模态感知融合在自动驾驶环境中的任务包括了经典的目标检测、语义分割、深度估计和深度预测这类的工作。其实常见的任务也主要也还是语义分割和目标检测。

目标检测

其实常见的无人驾驶的场景的目标检测有几个类型（汽车、行人、自行车、交通灯、交通指示牌、路锥、减速带）这些类型的物体。一般来说，目标检测使用由参数表示的矩形或长方体来紧密绑定预定义类别的实例，例如汽车或行人，这需要在定位和分类方面都表现出色。由于缺乏深度通道，2D对象检测通常简单地表示为 (x, y, h, w, c) ，而3D对象检测边界框通常会比2D的标注信息多了深度和方向两个维度的信息，表示为 $(x, y, z, h, w, l, \theta, c)$ 。

语义分割

除目标检测外，语义分割就是自动驾驶感知的另一个山头了。例如，我们会检测环境中的背景和前景目标，并加以区分，使用语义分割了解物体所在的区域以及区域的细节在自动驾驶任务中也是相当重要的。其次，一些车道线的检测方法还使用多类语义分割的mask来表示道路上的不同车道。

语义分割的本质是将输入数据的基本成分（如像素和三维点）聚类到包含特定语义信息的不同区域中去。具体来说，语义分割是指给定一组数据，例如图像像素 $D_I = \{d_1, d_2, \dots, d_n\}$ 或激光雷达3D点云 $D_L = \{d_1, d_2, \dots, d_n\}$ ，以及一组预定义的候选标签 $Y = \{y_1, y_2, y_3, \dots, y_k\}$ ，我们使用模型为每个像素或点 D_I 分配 k 个语义标签并将其放置在一个区域的任务。（其实这里说得有点复杂和晦涩，语义分割其实就当成是像像素级别的分类问题就行了，就是这么简单）

如果大家觉得还是比较抽象的话，可以看看下面三幅图像，这里面具体交代了不同场景下的任务，2D/3D的目标检测和语义分割任务。

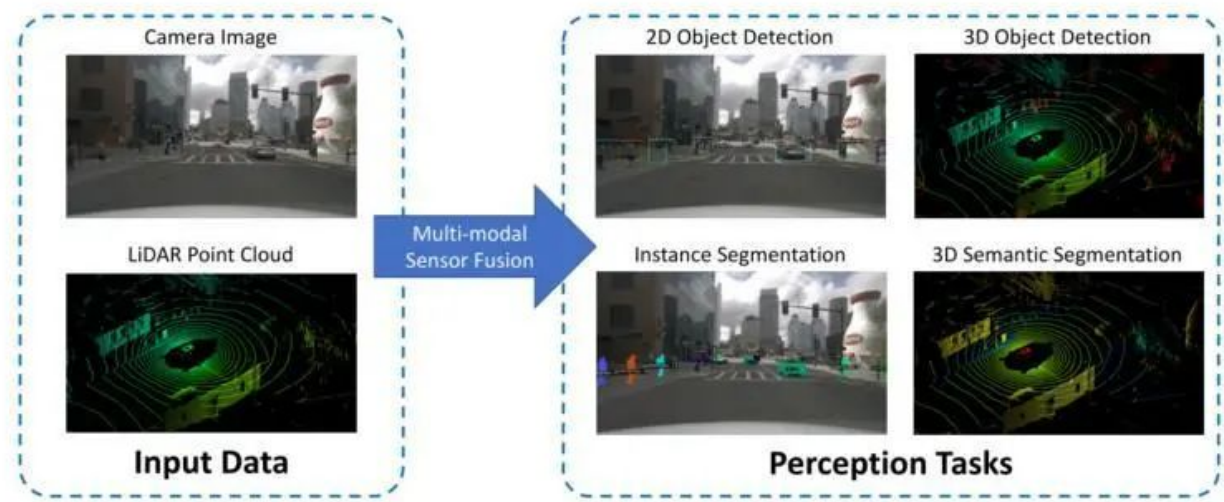


Figure 1. Perception Tasks of Autonomous Driving by Multi-modal Sensor Fusion Model.

基于多模态传感器融合的自动驾驶感知模型。

数据集

自动驾驶几个常见的老朋友，Kitti、Waymo、NuScenes，这些都是常见的带有3D信息的自动驾驶场景的数据集，基本上我们的一些多模态融合的任务也是围绕着这三个数据集进行刷榜的。具体的数据集的组成我这里就不交代了，大家可以去网站看看数据集的分布，其实要重点留意的是评价指标，这个其实非常重要，关系道整个优化的方向，所以大家要注意一点啦！其次了解数据集的时候（特别是新手，要注意数据的格式）因为Lidar的数据与以往的RGB图像的数据是不同的，所以要小心设计Lidar的分支，保护数据输入。

Table 1. Survey of Commonly Used Open Dataset and Benchmarks

Dataset	Year	Hours	LiDARs	Cameras	Annotated LiDAR Frames	3D Boxes	2D Boxes	Traffic Scenario	Diversity
KITTI [26]	2012	1.5	1 Velodyne HDL-64E	2 color, 2 grayscale cameras	15k	80k	80k	Urban, Suburban, Highway	-
Waymo [71]	2019	6.4	5 LiDARs	5 high-resolution pinhole cameras	230k	12M	9.9M	Urban, Suburban	Locations
nuScenes [6]	2019	5.5	1 Spinning 32-beams LiDAR	6 RGB cameras	40k	1.4M	-	Urban, Suburban	Locations, Weather
ApolloScape [33]	2018	2	2 VUX-1HA laser scanners	2 front cameras	144k	70k	-	Urban, Suburban, Highway	Weather, Locations
PandaSet [89]	2021	-	1 Mechanical spinning LiDAR and 1 Forward-facing LiDAR	5 wide-angle cameras and 1 forward-facing long-focus camera	6k	1M	-	Urban	Locations
EU Long-term [95]	2020	1	2 Velodyne HDL-32E LiDARs and 1 ibeo LUX 4L LiDARs	2 stereo cameras and 2 Pixelink PL-B742F industrial cameras	-	-	-	Urban, Suburban	Season
Brno Urban [47]	2020	10	2 Velodyne HDL-32e LiDARs	4 RGB cameras	-	-	-	Urban, Highway	Weather
A*3D [59]	2020	55	1 Velodyne HDL-64ES3 3D-LiDAR	2 color cameras	39k	230k	-	Urban	Weather
RELLIS [34]	2021	-	1 Ouster OS1 LiDAR and 1 Velodyne Ultra Puck	1 3D stereo camera and 1 RGB camera	13k	-	-	Suburban	-
Cirrus [82]	2021	-	2 Luminar Model H2 LiDARs	1 RGB camera	6k	100k	-	Urban	-
HUAWEI ONCE [51]	2021	144	1 40-beam LiDAR	8 high-resolution cameras	16k	417k	769k	Urban, Suburban	Weather, Locations

融合模式

这是本次宵夜讨论的高潮，就是关于怎么去融合这两种不同模态（类型）的数据呢？按现在的融合模态的发展模式来说是，一共是分为两大类四小类的模式，什么是两大类，什么是四小类呢？我们继续看下去！

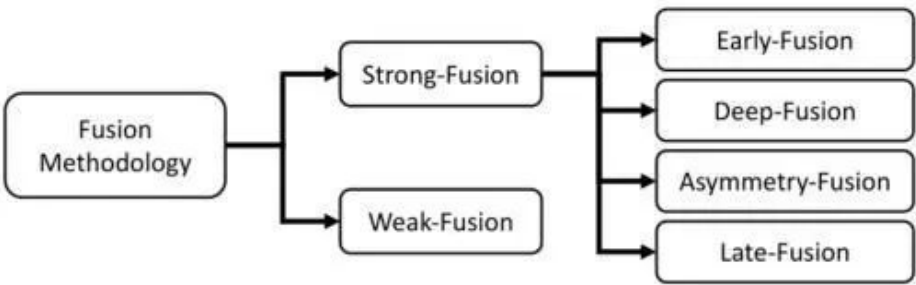


Figure 2. Fusion Methodology Overview

Early Fusion

有的伙伴会说，直接把数据的格式统一，合并起来输入不就行了吗？但是现阶段的Early Fusion并不是这么暴力。

Early Fusion一般是激光雷达数据和Image图像进行融合或者是激光雷达数据与Image的特征进行融合，两种方式。如下图所展示的情况LiDAR这个分支与Image信息的早期信息交互的过程。这种方式在reflectance, voxelized tensor, front-view/ range-view/ BEV,pseudo-point clouds都可以使用。尽管Image的特征在各个阶段都不同，但是都与LiDAR的信息高度相关。所以LiDAR信息+Image特征融合也是可以有效进行融合的。因为LiDAR的分支没有经过抽象化的特征提取阶段，所以这一阶段的数据仍具有可解释性，因此，LiDAR的数据表示依然可以进行直观的可视化。

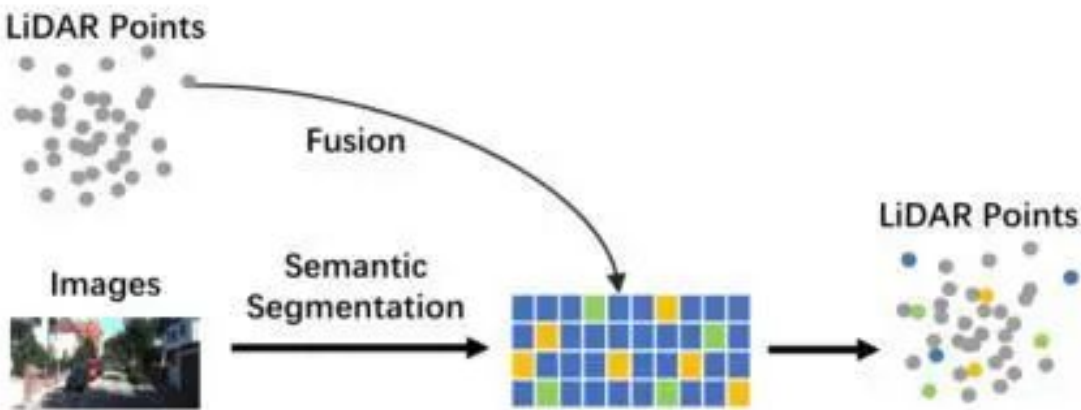


Figure 4. An Example of Early-Fusion

从图像的角度上看，严格意义上对数据级别图片的定义只能是包含RGB或Gray等数据。其实这种定义方式缺乏通用性和合理性，也比较局限。所以我们格局要大一点，数据级别可以不仅仅是图像，也可以是特征图。与传统的早期融合定义相比，文章将相机数据的定义不仅仅局限在image上，也将特征信息纳入其中。有意识的对特征信息进行选择融合，得到一个语义连接更加紧密的输入数据，之后将这个数据集进行放入网络进行特征提取。

无论是直接将数据类型转化一致，然后concat成一体，还是LiDAR信息与Image的特征信息进行融合，还是说两者先进行特征的语义连接后成为输入，这些都是Early Fusion的操作。其实这样的输入一体化操作的好处自然是结构简便、容易部署。通过语义的提前交互，也解决了传统早期融合，模态之间语义信息交互不充分的问题。所以一定程度上，选择Early Fusion也是一个不错的选择。

Deep-fusion

深度特征融合的方法其实也很常见。如下图

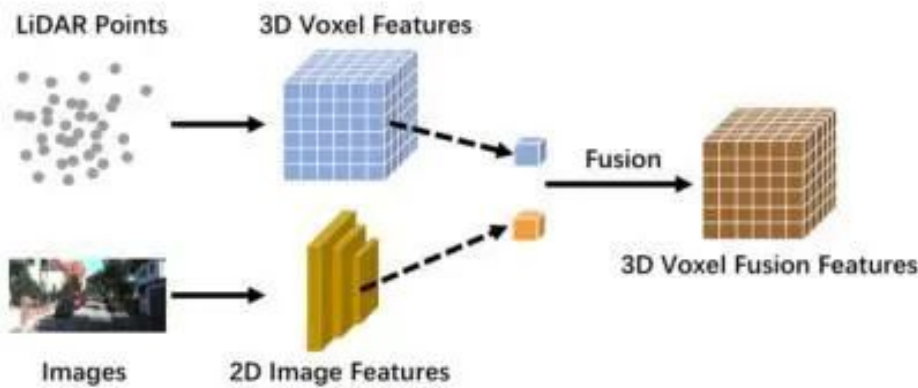


Figure 5. An Example of Deep-Fusion

深度特征融合

我们可以很直接清晰的看到LiDAR点图分支和Images分支在经过各自的特征提取器后，得到高维度的特征图，并通过一系列下游模块对两个分支模态进行融合。与其他的融合方式不同，深度融合有时候也会通过级联的方式对高级特征和原始特征进行融合，同时利用高级的特征信息和含有丰富物理信息的原始特征。

Late-fusion

后融合，也称为目标对象级别融合，表示在每个模态中融合结果的方法。一些后融合方法其实是同时利用了LiDAR点云分支和相机图像分支的输出，并通过两种模式的结果进行最终预测。

后期融合可以看作是一种利用多模态信息对最终方案进行优化的集成方法。

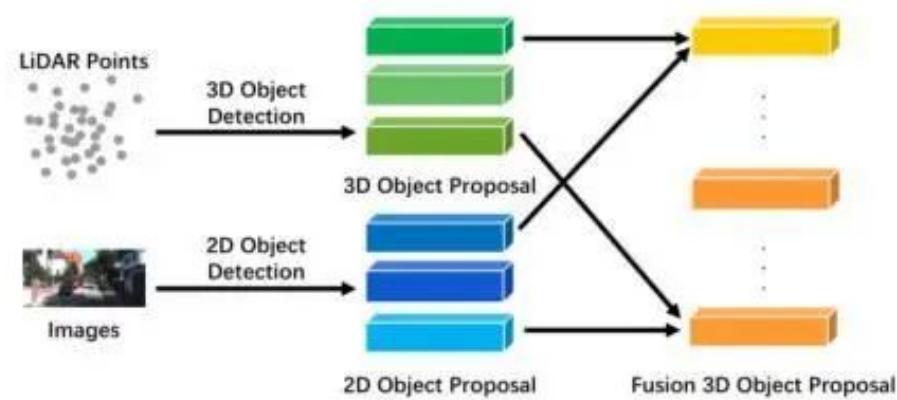


Figure 6. An Example of Late-Fusion

后融合

Asymmetry-fusion

除了早融合、深度融合和后融合之外，还有一些方法会对不同的分支赋予不同的特权，因此我们将融合来自一个分支的对象级信息，而来自其他分支的数据级或功能级信息的方法定义为不对称融合。与其他强融合方法看似平等地对待两个分支不同，不对称融合方法至少有一个分支占主导地位，其他分支只是提供辅助信息来完成最后的任务。下图就是一个经典的例子。与后期融合相比，虽然它们提取特征的过程是相似的，但不对称融合只有来自一个分支的一个提议，而后融合会融合所有的分支信息。

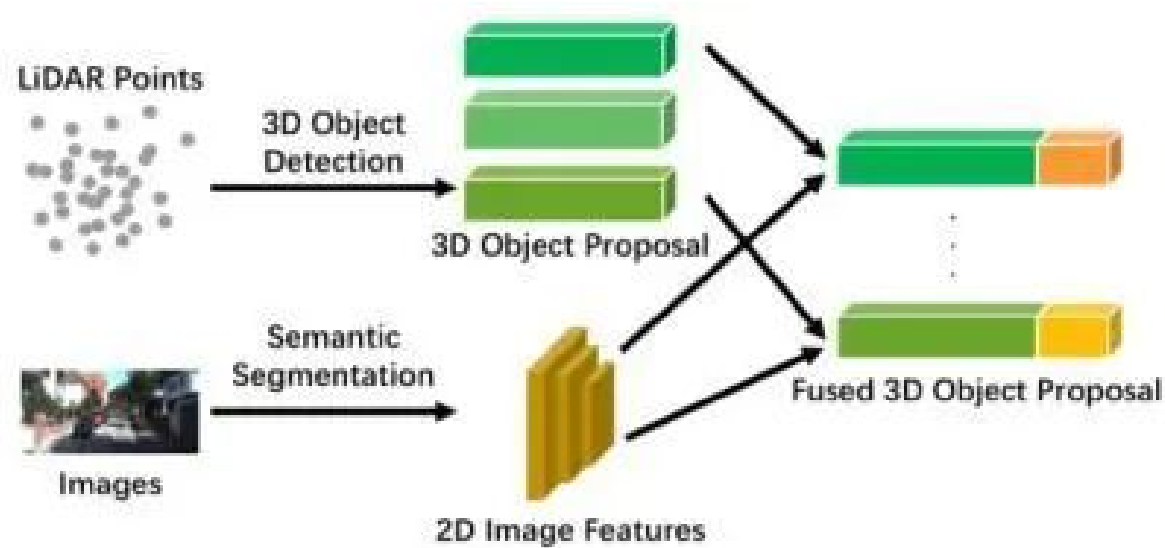


Figure 7. An Example of Asymmetry-Fusion

非对称融合

很显然这类型的融合方法也是合理的，因为卷积神经网络对摄像机数据具有良好的性能，它可以有效的过滤出在点云数据中没有实际语义信息的无用点，在融合的时候就可以一定程度上避免噪声点的干扰。不仅如此，还有一些作品尝试跳出常规，使用激光雷达骨干来指导2D多视角的数据进行融合，通过信息的交互指导实现更高的准确度。

Weak-Fusion

与强融合不同，弱融合方法不直接从多模态分支融合（数据/特征/对象），而是以其他方式操作数据。基于弱融合的方法通常使用基于规则的方法，利用一种模式中的数据作为监督信号，以指导另一种模式的交互。下图展示了弱融合模式的基本框架。弱融合不同于上述不对称融合融合图像特征的方法，它直接将选中的原始LiDAR信息输入到LiDAR主干中，过程中不会直接与Image的分支主干进行特征的交互，会通过一些弱连接的方式（比如loss函数）等方式进行最后的信息融合。与之前的强融合的方法比，分支的信息交互是最少的，但是同时也能够避免在交互过程中彼此的信息不对称带来的信息干扰，又或者是避免了因为单一分支的质量不过关，而影响整理整体的融合推理。

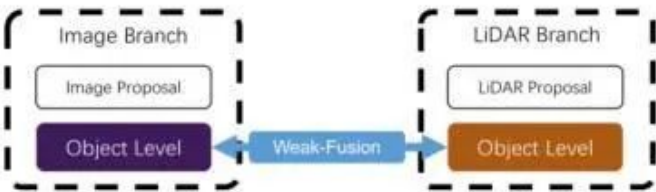


Figure 8. Weak-Fusion Overview

弱融合

Other-Fusion

有些工作不能简单地定义为上述任何一种融合，因为它们在整个模型框架中拥有不止一种融合方法。造融合怪是我们这些盐究圆天生的技能，你说A有xxx好处，B有xxx好处，那我A+B不就是直接赢麻了吗？但是事实上，很多情况都不能有效的把方法缝合进去。如深度融合和后融合方案的相互结合，或者将前融合和深度融合结合在一起。这些方法在模型设计上存在冗余问题，并不是融合模块的主流方法，即没有取得A+B的效果，反而极大的牺牲了推理时间已经增大了算法的复杂度。一些实验结果

Table 2. Survey of BEV Task Results in KITTI [26] for Test Dataset

Method	Year	Car			Pedestrian			Cyclist		
		Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
Early-Fusion										
PFF3D [87]	2021	89.61	85.08	80.42	48.74	40.94	38.54	72.67	55.71	49.58
Painted PointRCNN [76]	2020	92.45	88.11	83.36	58.70	49.93	46.29	83.91	71.54	62.97
PI-RCNN [90]	2020	91.44	85.81	81.00	-	-	-	-	-	-
Complexer-YOLO [68]	2019	77.24	68.96	64.95	21.42	18.26	17.06	32.00	25.43	22.88
MVX-Net(PF) [69]	2019	89.20	85.90	78.10	-	-	-	-	-	-
Deep-Fusion										
RoIFusion [9]	2021	92.88	89.03	83.94	46.21	38.08	35.97	83.13	67.71	61.70
EPNet [32]	2020	94.22	88.47	83.69	-	-	-	-	-	-
MAFF-Net [105]	2020	90.79	87.34	77.66	-	-	-	-	-	-
SemanticVoxels [22]	2020	-	-	-	58.91	49.93	47.31	-	-	-
MVAF-Net [78]	2020	91.95	87.73	85.00	-	-	-	-	-	-
3D-CVF [102]	2020	93.52	89.56	82.45	-	-	-	-	-	-
MMF [45]	2019	93.67	88.21	81.99	-	-	-	-	-	-
ContFuse [46]	2018	94.07	85.35	75.88	-	-	-	-	-	-
SparsePool [85]	2017	-	-	-	43.33	34.15	31.78	43.55	35.24	30.15
Late-Fusion										
CLOCs [55]	2020	92.91	89.48	86.42	-	-	-	-	-	-
Asymmetry-Fusion										
VMVS [40]	2019	-	-	-	60.34	50.34	46.45	-	-	-
MLOD [17]	2019	90.25	82.68	77.97	55.09	45.40	41.42	73.03	55.06	48.21
MV3D [12]	2017	86.62	78.93	69.80	-	-	-	-	-	-
Weak-Fusion										
Faraway-Frustum [104]	2020	91.90	88.08	85.35	52.15	43.85	41.68	79.65	64.54	57.84
F-ConvNet [83]	2019	91.51	85.84	76.11	57.04	48.96	44.33	84.16	68.88	60.05
IPOD [99]	2018	86.93	83.98	77.85	60.83	51.24	45.40	77.10	58.92	51.01
F-PointNet [60]	2017	91.17	84.67	74.77	57.13	49.57	45.48	77.26	61.37	53.78

2Dkitti上的结果

Table 3. Survey of 3D Task Results in KITTI [26] for Test Dataset

Method	Year	Car			Pedestrian			Cyclist		
		Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
Early-Fusion										
PFF3D [87]	2021	81.11	72.93	67.24	43.93	36.07	32.86	63.27	46.78	41.37
Painted PointRCNN [76]	2020	82.11	71.70	67.08	50.32	40.97	37.87	77.63	63.78	55.89
PI-RCNN [90]	2020	84.37	74.82	70.03	-	-	-	-	-	-
Complexer-YOLO [68]	2019	55.93	47.34	42.60	17.60	13.96	12.70	24.27	18.53	17.31
MVX-Net(PF) [69]	2019	83.20	72.70	65.20	-	-	-	-	-	-
Deep-Fusion										
RoIFusion [9]	2021	88.09	79.36	72.51	42.22	35.14	32.92	80.84	64.05	58.37
EPNet [32]	2020	89.81	79.28	74.59	-	-	-	-	-	-
MAFF-Net [105]	2020	85.52	75.04	67.61	-	-	-	-	-	-
SemanticVoxels [22]	2020	-	-	-	50.90	42.19	39.52	-	-	-
MVAF-Net [78]	2020	87.87	78.71	75.48	-	-	-	-	-	-
3D-CVF [102]	2020	89.20	80.05	73.11	-	-	-	-	-	-
MMF [45]	2019	88.40	77.43	70.22	-	-	-	-	-	-
SCANet [49]	2019	76.09	66.30	58.68	-	-	-	-	-	-
ContFuse [46]	2018	83.68	68.78	61.67	-	-	-	-	-	-
PointFusion [92]	2018	77.92	63.00	53.27	-	-	-	-	-	-
SparsePool [85]	2018	-	-	-	37.84	30.38	26.94	40.87	32.61	29.05
Late-Fusion										
CLOCs [55]	2020	89.16	82.28	77.23	-	-	-	-	-	-
Asymmetry-Fusion										
VMVS [40]	2019	-	-	-	53.44	43.27	39.51	-	-	-
MLOD [17]	2019	77.24	67.76	62.05	47.58	37.47	35.07	68.81	49.43	42.84
MV3D [12]	2017	74.97	63.63	54.00	-	-	-	-	-	-
Weak-Fusion										
Faraway-Frustum [104]	2020	87.45	79.05	76.14	46.33	38.58	35.71	77.36	62.00	55.40
F-ConvNet [83]	2019	87.36	76.39	66.69	52.16	43.38	38.80	81.98	65.07	56.54
IPOD [99]	2018	79.75	72.57	66.33	56.92	44.68	42.39	71.40	53.46	48.34
F-PointNet [60]	2018	82.19	69.79	60.59	50.53	42.15	38.08	72.27	56.12	49.01

3Dkitti上的结果

上面的实验结果，大家可以简单的看一下就行

多模态感知融合的未来

近年来，自动驾驶感知任务的多模态融合方法取得了快速进展，从更高级的特征表示到更复杂的深度学习模型。然而，仍有一些更开放的问题有待解决。在此，我们总结了今后需要做的一

些关键和必要的工作。

如何做更先进的融合方法

其实当前阻碍模态融合的最大拦路虎有两只。

A：融合模型不对齐

B：信息丢失

相机和激光雷达的内在和外在是截然不同的。两种方式的数据都需要在新的坐标系统下重新组织。

传统的早期和深度融合方法利用外部校准矩阵将所有激光雷达点直接投影到相应的像素上，或反之亦然。因为数据样本会存在噪声，在噪声的干扰下，这种对齐的方式，很显然是没有办法做到精准对齐的。无论怎样，想要单靠机械的手段消除机器带来的误差，不仅难度大，还要付出比较大的成本。所以我们可以看到现在的方法，除了这种严格的转化，一一对应之外，还可以利用一些周围信息作为补充以使得融合工作可以获得更好的性能。

此外，在输入和特征空间的转换过程中不可避免会还存在一定的信息丢失。因为在特征提取的降维过程中投影会不可避免地导致大量的信息丢失。

因此，通过将两个模态数据映射到另一种专门用于融合的高维表示，可以在未来的工作中有效地利用原始数据，减少信息损失。还有一些方法是采用直接的串联数据，通过赋权值的方式进行融合。但是当前的方案依旧是不太成熟，只通过像素之间的赋权值，相加这些简单的操作可能无法融合分布差异较大的数据，因此，很难弥合两种模式之间的语义差距。一些工作试图使用更精细的级联结构来融合数据并提高性能。在未来的研究中，双线性映射等机制可以融合不同特征的特征。

合理利用多个模态的信息

大多数框架可能只利用了有限的信息，没有精心设计进一步的辅助任务来进一步了解驾驶场景。

我们当前做的内容，会把语义分割、目标检测、车道线检测这些任务单独讨论，割裂这些任务。之后再不同的模型组合到一起提供服务，其实这显然是冗余的工作。所以我们为啥不做一个多任务框架，一次性覆盖不同的任务呢？在自动驾驶场景中，许多具有显式语义信息的下游任务可以大大提高目标检测任务的性能。例如，车道检测可以直观地为车道间车辆的检测提供额外的帮助，同时语义分割结果可以提高目标检测性能。

因此，未来的研究可以同时通过对车道、交通灯、标志等下游任务进行同时的检测，构建大一统的自动驾驶任务，辅助感知任务的执行。

与此同时，其实时序信息这类型的信息在自动驾驶感知任务里面也十分的重要。像BEVFormer就使用了RNN对时序信息进行整合，最后使得整体任务可以有效的生成BEV的视图。时间序列信息包含了序列化的监督信号，可以提供比单一帧的方法更稳定的结果，也更加适应自动驾驶的整体任务需求。

未来的研究可以集中在如何利用多模态数据进行自监督学习，（包括预训练、微调或对比学习）。通过实现这些最先进的机制，融合模型将导致对数据的更深入的理解，并取得更好的结果。大家看到MAE这么好的效果，其实如果我们的感知任务也引入这套方法进行实验，我相信一定会取得更可喜的成绩。

感知传感器的内在问题

区域的偏差或者分辨率上的不一致，与传感器设备有密不可分的关系。这些意想不到的问题严重阻碍了自动驾驶深度学习模型的大规模训练和实现，数据的质量以及数据的收集方案都是当前阻碍自动驾驶感知任务再发展的一大问题点。

在自动驾驶感知场景中，不同传感器提取的原始数据具有严重的领域相关特征。不同的相机系统有不同的光学特性，成像原理也不一致。更重要的是，数据本身可能是有领域差异的，如天气、季节或位置，即使它是由相同的传感器捕获的，他们所呈现出来的影像也有着很大的出入。由于这种变化，检测模型不能很好地适应新的场景。这种差异会导致泛化失败，导致大规模数据集的收集和原始训练数据的可重用性下降。因此，如何消除领域偏差，实现不同数据源的自适应集成也会是今后研究的关键。

来自不同模式的传感器通常具有不同的分辨率。例如，激光雷达的空间密度明显低于图像。无论采用哪种投影方法，都没有办法找到一一对应关系，所以常规的操作会剔除一些信息。无论是由于特征向量的分辨率不同还是原始信息的不平衡，都可能会导致弱化了一边模态分支的信息量，或者说是存在感。变成以某一特定模态的数据为主，因此，未来的工作可以探索一种与不同空间分辨率的传感器兼容的数据方式。

总结

文章很详细的为我们整理了当前多模态融合的一些工作以及一些未来发展的方向。其实我觉得我们需要一套合理化决策，而且成本较低的多模态融合框架，就需要我们对我们的数据更加的了解。此外我们也需要更多的数据。更多的分析成本的投入。像自监督、对比学习、大规模预训练这类型的赋能工作，直接一套或许也能取得很棒的成绩，但是这种核弹式的打击并不是优化工作的核心内容。无论是前、深、后、不对称哪种强融合方式，都有自身的弊端。都需要我们对其进行进一步的优化改进，我们需要根据融合的数据特性进行调整。当前的融合可能

是单一的两个模态的交互融合，但是实际上未来的工作，远远不止两个模态，两个传感器进行交互这么简单，那么我们现在的工作又可以怎么做呢？这个故事还需要我们继续进行探讨！

公众号后台回复“ECCV2022”获取论文分类合集下载~



极市平台

为计算机视觉开发者提供全流程算法开发训练平台，以及大咖技术分享、社区交流、竞赛...
848篇原创内容

公众号

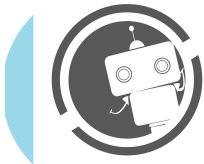
△点击卡片关注极市平台，获取最新CV干货

极市干货

- 算法竞赛：1000W奖池，AI算法等你来战！粤港澳大湾区（黄埔）国际算法算例大赛正式开赛！
- 技术综述：防止模型过拟合的方法汇总 | 浅析深度学习在图像处理中的应用趋势及常见技巧
- 极视角动态：极视角与惠普签署战略合作协议，共拓智慧工业市场 | 极智产品融合 | 极视角罗韵解读「行业+AI」融合创新的实操手册



极市平台签约作者



matrix明仔

知乎：战斗系牧师

一个希望百发百中CCF-A的在读美食爱好者

研究领域：目标检测、语义分割、多模态感知融合

(RGB、Thermal、Depth、Ldair等不同模态下感知信息间的决策与融合)

作品精选

一个RGB-T工作的小小感悟与总结

在计算机领域的科研中，最初的创新点从何而来？

深度学习刷SOTA有哪些trick？



投稿方式：

添加小编微信Fengcall（微信号：fengcall19），备注：姓名-投稿

△长按添加极市平台小编

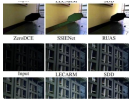
// 点击阅读原文进入CV社区
收获更多技术干货

阅读原文

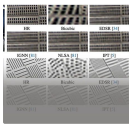
喜欢此内容的人还喜欢



ICCV23 | 将隐式神经表征用于低光增强，北大张健团队提出NeRCo
极市平台



ICCV 2023 | 南开程明明团队提出适用于SR任务的新颖注意力机制（已开源）
极市平台



ICCV 2023 | Pixel-based MIM: 简单高效的多级特征融合自监督方法
极市平台

