计算机研究与发展
Journal of Computer Research and Development
https://crad.ict.ac.cn
62(1): 2−21, 2025

亮点论文

车计算:自动驾驶时代的新型计算范式

鲁思迪[1] 何元恺[2] 施巍松[2]
[1](威廉玛丽计算机科学系 美国弗吉尼亚州威廉斯堡 23185)
[2](特拉华大学计算机与信息科学系 美国特拉华州纽瓦克 19716)
(sidi@wm.edu; willhe@udel.edu)

Vehicle Computing: An Emerging Computing Paradigm for the Autonomous Driving Era

Lu Sidi[1], He Yuankai[2], and Shi Weisong[2]
[1](*Department of Computer Science*, *William & Mary*, *Williamsburg*, *VA*, *USA* 23185)
[2](*Department of Computer & Information Sciences*, *University of Delaware*, *Newark*, *DE*, *USA* 19716)

Abstract With rapid advancements in edge computing, sensing, AI, and communication technologies, vehicles are undergoing an unprecedented transformation. We introduce a new computing paradigm for the autonomous driving era—vehicle computing. In this paradigm, data and control layers are separated, creating an open computing platform that supports multi-party collaboration and data sharing, breaking away from the limitations of traditional, closed vehicle systems. This paradigm enables vehicles to transcend conventional transportation roles, evolving into versatile mobile computing platforms that support a wide range of advanced applications and third-party services. We define the core concepts of vehicle computing, analyze the revolutionary evolution of software and computing architectures within vehicles, and present promising application examples, as well as a novel business model enabled by this paradigm. The five core functionalities of vehicle computing, i.e., computation, communication, energy management, sensing, and data storage, and their related cutting-edge technologies are thoroughly explored. We conclude by discussing key technical challenges and promising opportunities within vehicle computing, aiming to inspire further academic and industry research in this innovative field.

Key words vehicle computing; autonomous driving; computation; communication; energy management; mobile sensing; data storage; software-defined vehicles

摘 要 随着边缘计算、传感、人工智能和通信技术的迅猛发展,车辆正经历前所未有的深刻变革.提出了一种适用于自动驾驶时代的全新计算范式——车计算.在这一新型范式下,车辆的数据层与控制层被分离,车辆系统转向开放的计算平台,构建了支持多方协作的数据共享结构,从而打破了传统车辆系统的封闭局限,使车辆从单一的交通工具演变为支持丰富的高级应用和第三方服务的多功能移动计算平台.全面阐述了车计算范式的核心理念,深入分析了车辆软件与计算架构的革命性演进,提供了具有应用前景的车计算实例,并介绍了在该范式支持下具有颠覆潜力的新型商业模式.此外,深入探讨了车计算的五大核心功能:计算、通信、能源管理、传感和数据存储,以及相关的前沿技术.最后,总结了车计算领域的关键技术挑战和广阔的机遇,期望能够激发学术界和工业界对这一创新领域的深入研究与高度关注.

关键词 车计算;自动驾驶;计算;通信;能源管理;移动感知;数据存储;软件自定义车辆

中图法分类号 TP391

收稿日期:2024-06-17;**修回日期**:2024-11-14

通信作者:施巍松(weisong@udel.edu)

DOI: 10.7544/issn1000-1239.202440538 CSTR: 32373.14.issn1000-1239.202440538

边缘计算、机器学习和通信技术的发展正推动汽车行业发生革命性改革. 传统汽车公司与互联网科技公司正展开密切合作. 例如, 福特公司与谷歌公司合作, 推动前沿车载技术的创新并提升驾驶体验; 通用汽车公司与微软公司携手, 促进自动驾驶时代的到来; 亚马逊公司不仅与丰田公司合作提升其车载服务, 还与宝马集团联手, 推动数据驱动的车载服务的创新. 同时, 本田公司与谷歌公司也在车载服务方面展开合作; 大众公司与微软公司的联盟旨在加速自动驾驶技术的发展; 阿里巴巴公司与上汽集团携手进军电动汽车市场; 百度公司与吉利公司合资成立了智能电动汽车公司. 与此同时, 特斯拉公司也在全球积极扩建数据中心和人工智能评估基础设施. 小米公司也大力进军汽车领域, 提出了“整合人、车、家的全生态系统”这一口号.

由此可见, 车计算的时代已经到来: 车辆从传统的封闭、机械为主的运输工具, 逐渐演变为以人工智能为核心、互联互通、具备动态功能的开放的移动计算平台[1-2]. 这些车辆正成为智能城市和人们生活中的关键组成部分, 承担起计算、通信、能量管理(包括能量的消耗、存储和发送)、感知以及数据存储等多重功能, 如图 1 所示.



Fig. 1 Five core functions of vehicle computing [3]

图 1 车计算的 5 个核心功能[3]

1 背 景

1.1 汽车软件和计算系统架构的演变

在过去几十年里, 苹果等创新型信息技术公司通过提供尖端的应用和体验, 使智能手机从一次性交易的商品转变为承载丰富应用生态的智能设备. 类似地, 智能软件和先进算法的集成正成为汽车创新的关键, 以应对不断变化的市场需求和技术发展.

1)车辆软件的演变. 在车辆的计算系统中, 算法的开发与部署对于感知、定位、预测和控制等任务至关重要. 现代车辆由超过 1.5 亿行代码驱动, 这些代码分布在 100 多个电子控制单元(ECUs)上, 以实现各种复杂的智能服务[3]. 这些车载服务涵盖了广泛的应用, 包括机油寿命预测系统[4]、刹车片磨损预测[5]、优化的轨迹规划[6]、动态行程时间估算算法[7]、自适应巡航控制[8]、电池健康预测分析[9]、传感器数据异常检测[10]、高级车道检测技术[11]、黑冰等危险情况检测系统[12]、坑洞识别方法[13]、实时物体检测[14]、以多模态大语言模型为基础的高级驾驶[15], 以及先进的车内空气质量管理系统[16]等, 如图 2 所示. 汽车软件的复杂性与数量正以指数级快速增长[3].

2)车辆计算系统架构的演变. 此外, 车辆的硬件架构也正从分布式架构向集中式架构转变. 车辆通常配备多个由不同供应商开发的 ECU, 这些 ECU 通

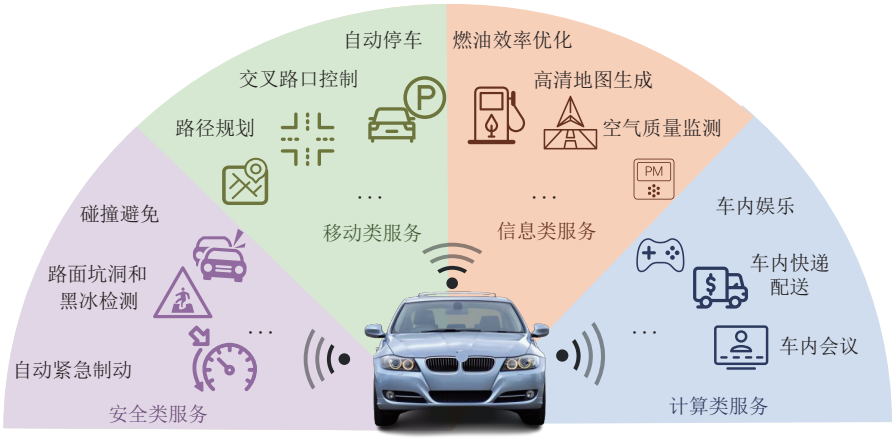
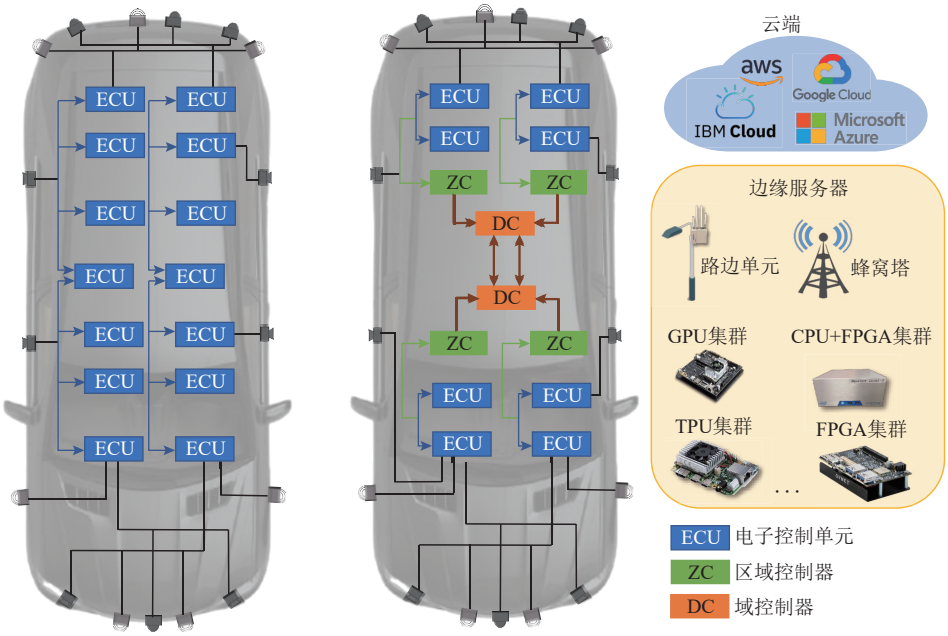


Fig. 2 Major types of vehicle applications[17]

图 2 主要的车辆应用类型[17]

过控制器局域网(CAN 总线)相互连接,使数据按顺序传输到所有连接的设备,从而使汽车功能分布在多个 ECU 中,如图 3(a) 所示. 然而,由于 ECU 的计算资源有限,这种传统的分布式架构在支持软件多样性和计算密集型车载应用方面面临重大限制.

行业领导者如特斯拉和 APTIV[18] 率先采用了集中式的车载高性能计算架构,如图 3 (b) 所示: ECU 被划分为功能相关的区域,由区域控制器(ZC)统一管理,每个区域再由域控制器(DC)统筹控制. 这些控制器均可运行嵌入式 Linux 操作系统. 此外,通过外围组件互连(PCIe)和计算快车道(CXL)等标准微处理器接口,可将加速器(如 GPU 和 FPGA)集成至 ZC 和 DC 中,从而增强了计算能力. 该架构的固有兼容性还促进了车辆与边缘服务器和云端的协作.



(a) 传统架构 (b) 软件定义架构

Fig. 3 Evolutionary trends in automotive computing system architecture[17]

图 3 汽车计算系统架构的演变趋势[17]

1.2 自动驾驶时代下的软件定义汽车

1)软件定义汽车(SDV)的本质. 在自动驾驶时代,软件定义汽车[19] 已成为不可避免的趋势. SDV 是一种通过软件来控制、管理和更新其核心功能的汽车. 不同于传统的硬件驱动架构,SDV 强调车辆作为高度灵活且可升级的计算平台,在整个生命周期中(从开发验证到销售与服务),通过软件实现功能定义、性能优化和服务扩展,以支持包括自动驾驶在内的多样化应用[20]. 通过定期的软件更新,SDV 在性能、安全性和舒适性等方面不断提升,从而实现更高的互联性、自动化和个性化[21]. 自动驾驶汽车是 SDV 的典型代表,它利用先进的软件和算法实现车辆的自主决策与操作. 如今,高级汽车已经包含超过 1.5 亿行的代码,预计到 2030 年代码量将达到 3 亿行[17].

2)SDV 的重要性. SDV 不仅通过空中软件更新(OTA)使车辆能够远程接收软件补丁和漏洞修复,还具备其他显著优势,包括: ① 简化集成和自动部署. 利用持续集成/持续部署(CI/CD)框架,促进尖端服务的独立部署和高效协作. ② 跨应用的可扩展性和重用性. 促进代码的高效重用,使得软件能够在不同车型和代际中广泛部署,提升了系统的可扩展性. ③ 增强的故障隔离. 通过将软件与其操作环境分离,确保各模块能够独立且协调地运行,从而实现更有效的故障隔离,提升车辆安全性.

3)SDV、SDN 和 SDS 的本质区别. SDV 与软件定义网络(SDN)和软件定义卫星(SDS)在本质上有所不同. SDN 专注于通过分离控制和平面优化网络资源管理; SDS 则增强卫星通信的灵活性和可配置性. 而 SDV 旨在通过灵活的软件和计算平台,将汽车转变为可动态配置和管理的智能移动平台. SDV 不仅支持自动驾驶技术,还可以通过多样化的软件功能提升车辆的安全性、用户体验以及其他智能化应用.

2 车计算

2.1 车计算范式

图 4 展示了传统范式与车计算范式的差异. 传统车辆系统采用闭环控制结构, 如图 4(a) 所示, 其数据和控制仅限于车辆内部使用. 车内传感器收集的数据(如轮胎压力)直接作用于车辆硬件, 形成封闭的反馈系统. 甚至车内的 CAN 总线也是封闭的, 仅供自身使用, 几乎没有数据共享的概念. 车内传感器和控制系统紧密结合, 主要提供基本的驾驶功能和安全保障.

如图 4(b) 所示, 在车计算范式中, 车辆系统转向开放的计算平台, 数据层和控制层被清晰分离. 车辆数据可以通过接口向第三方开放, 支持高级应用(如高级驾驶辅助系统)和增值服务(如车内配送). 其中, 有些服务能够辅助控制车辆, 而另一些则仅利用数据提供功能(如视频服务). 这种数据与控制开放的模式使车厂能够通过接口支持第三方开发者利用车辆数据创建新应用, 从而拓展车计算服务的多样性. 这一新的车计算范式尚处于推广阶段, 虽然部分车厂已经开始采用, 但实现全面的 App 开放仍需时间. 本文仅提出了该范式的概念构想.

2.2 车计算技术及其四层计算架构

车计算技术指的是一套将车辆转变为具备计算、通信、能量管理(包括能量的消耗、存储和发送)、感知和数据存储等功能于一体的综合平台的技术. 它将车辆定位为强大的移动计算单元, 能够处理并分析来自车载传感器和周围连接设备的数据流[22]. 即使在车辆静止或充电时, 仍然可以执行复杂的计算任务, 并将相关结果反馈给终端用户. 例如, 假设一名配备随身摄像头的执法人员正在执行任务, 摄像头会实时采集视频数据并将其发送到周围停泊的执法车辆, 用于延迟敏感的应用如车牌识别. 在这个场景中, 执法车辆作为高效的计算平台, 利用从联网设备(即随身摄像头)接收的数据, 有效地分配计算资源以完成任务[23].

基于车计算技术的定义, 我们进一步总结未来的四层车计算架构, 涵盖了云端(支持大规模数据处理和复杂算法运算)、边缘服务器(拥有中等计算能力)、车辆(具有有限计算能力), 以及边缘设备(计算能力相对较弱但分布广泛), 如图 5 所示.

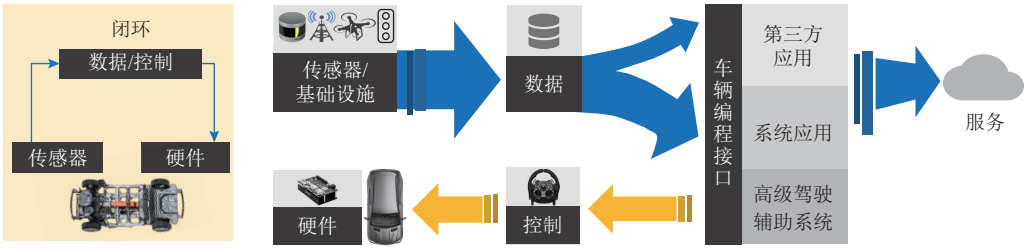
四层车计算架构强调车辆与云端、边缘服务器和物联网设备之间的协作完成计算. 通信作为其使能技术, 包括车辆到基础设施(V2I)、车辆到车辆(V2V)以及车辆到万物(V2X). 这种协作不仅使车辆能与交通系统组件(如蜂窝塔、交通路口、无人机、电动滑板车乃至行人)进行协同合作, 还能让车辆与周围的非交通设备(如智能家居传感器、工业物联网设备、健康传感器和边缘服务器)互动, 共同完成计算任务.

然而, 随着车计算引入更多计算需求, 现有的 V2V 和 V2X 功能可能需要相应调整. 例如, 当前的 5G 网络因复杂的基础设施、高昂的部署成本及信号穿透能力的限制, 仍难以在城市环境中实现稳定通信, 无法完全满足车计算的需求. 因此, 需要向专为车辆设计的通信网络方向发展, 如蜂窝车联网(C-V2X).

2.3 车计算的示例

接下来, 我们概述并分析了 5 个车计算的案例, 以进一步阐明车计算的概念.

1)基础设施健康管理. 在过去, 研究人员在桥梁上广泛布置了多种类型的传感器, 用于监测桥梁的健康状况[25]. 然而, 桥梁上的传感器长期承受众多车辆行驶带来的桥梁的震动以及日晒风吹等自然环境影响, 导致传感器老化严重. 在桥梁上安装和长期维护传感器具有相当大的挑战, 使得传统方法在监测桥梁等基础设施方面面临诸多困难. 在自动驾驶时代, 由于车辆配备了多种传感器(如压力传感器、振



(a) 传统范式 (b) 车计算范式

Fig. 4 Traditional paradigm and vehicle computing paradigm

图 4 传统范式和车计算范式

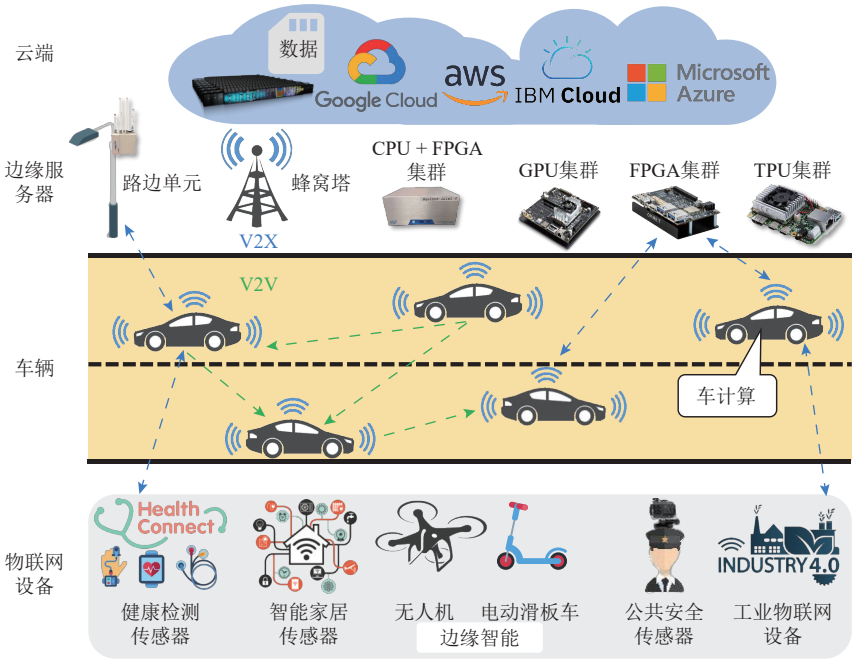


Fig. 5 The 4-tier vehicular computing architecture[24]

图 5 四层车计算架构[24]

动传感器、GPS、摄像头、雷达和激光雷达)和较为强大的计算单元，车辆本身可以视为移动传感器和移动计算平台，并通过车计算来有效监控桥梁等基础设施的健康状况.

如图 6 所示，当车辆行驶在桥梁上时，可持续收集桥梁的结构响应(例如振动)，这些数据可作为桥梁健康状态的指标. 除了传感器数据，车辆还能够生成实时驾驶数据(如速度、停止和加速情况)并捕获天气、车辆型号和道路信息等. 通过车计算，车辆能够直接分析这些数据以评估桥梁健康状况，并将初步决策发送至边缘服务器(如路边单元). 边缘服务器从多辆车接收相关信息并进行确认. 若有必要，边缘服务器会向周围及前方车辆发送警告和交通控制信号，并将相关结果推送至云端进行存储，供潜在的进一步分析和长期监测使用.

2)车内会议. 在自动驾驶时代，随着驾驶员注意

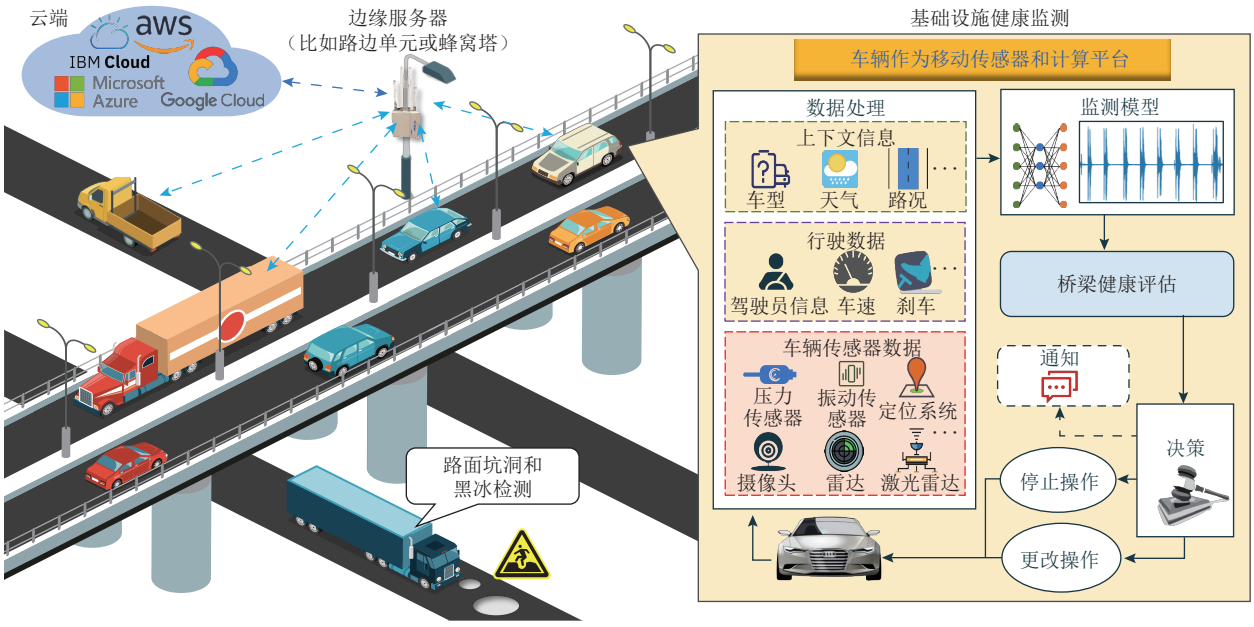


Fig. 6 An example of infrastructure health monitoring facilitated by vehicle computing[19]

图 6 利用车计算进行基础设施健康监测的示例[19]

力需求的降低甚至无需驾驶员，创新的车内服务成为可能，例如高效流畅的车内在线会议. 在车辆行驶过程中，除了自动驾驶模型，车辆还需运行多个计算模型来处理车内会议产生的大量数据，这要求强大的计算能力以确保实时性. 此外，这对网络通信技术和实现方式也提出了新的需求. 在车内会议的场景中，当用户返回家中时，车辆、云端和个人通信设备之间的安全通信变得尤为重要. 这样，无论用户位于家中、车内或办公室，均可无缝参与同一会议，无需反复注销和登录，从而显著提升工作效率并节省时间.

3）车内快递配送. 汽车将不仅仅是交通工具，还将成为智能家居的重要组成部分. 例如，即便用户不在家，汽车也可支持便捷且安全的车内快递服务. 目前，亚马逊公司正与汽车制造商合作，推出早期的车内配送服务：当配送司机抵达停放在公共场所的车辆旁边时，会发送请求以远程解锁车辆进行配送. 将包裹放置在后备箱或车内后，司机再次发送命令锁定车辆，用户随后会收到最终的确认通知. 车内快递服务不仅需要高效的物流管理，还依赖复杂的计算技术. 例如，车辆需实时规划最优路径，确保包裹能够及时、安全送达；在配送过程中，车辆还需进行安全验证和记录，以确保每个环节符合用户要求. 这些计算任务涵盖了物理路径规划、数据安全、用户身份验证以及交互式通知等多方面处理，从而保障配送的高效性与可靠性.

4）车内增强现实. 增强现实（AR）技术有望将汽车的挡风玻璃转变为互动显示屏，通过覆盖动态全彩图形在现实世界视图上来增强驾驶体验，提供车辆周围环境、路线和潜在危险的实时广角信息. 目前，3D 地图软件供应商 Civil Maps 向乘客展示了配备 AR 显示器的汽车如何在复杂的驾驶环境中导航. 此外，阿里巴巴公司投资了 1 800 万美元给 WayRay，这是一家推出 Navion 的抬头显示器公司. Navion 是首个全息 AR 车辆导航系统，能够无需佩戴 AR 头盔或眼镜显示行驶详情. 这些先进的 AR 界面预计将通过语音命令和手势提供直观的互动. 实现这些 AR 功能需要大量的计算资源来实时处理和显示复杂的信息，确保驾驶员和乘客在驾驶过程中获得准确和即时的环境信息.

5）车内娱乐. 车辆还将通过丰富的视频和音频娱乐选项来提升乘车体验. 预计到 2032 年，全球汽车娱乐市场将达到 581.8 亿美元[26]. 比如，数百万辆福特和林肯汽车将搭载谷歌公司的 Android 操作系统，为乘客提供嵌入式的谷歌应用和服务. 乘客将能够通过接口选择各种扩展现实（XR）游戏. 这些 XR 游戏能够利用车辆的实时数据和传感器信息（如加速度、车速、转向角度等），带来高度互动的体验. 乘客在虚拟环境中可以体验到车辆的真实物理反馈，仿佛置身于不同的驾驶场景. 在这一过程中，车辆需要实时处理和分析大量数据，以确保游戏画面和物理反馈的精准性与流畅性. 例如，系统需快速响应乘客的操作，同时适应车辆在实际道路环境中的动态变化，确保游戏体验的真实性与安全性. 此外，自动驾驶车辆的不同乘客还可以通过 V2V 通信在路上共同参与车内 XR 游戏，从而进一步丰富车内娱乐的多样性.

2.4 车计算的新型商业模式

尽管车计算能带来诸如支持实时数据处理等诸多好处，但也增加了计算和通信所需的硬件、软件及其持续服务和维护的成本. 为应对这些财务影响，采用创新且具有变革性的商业模式至关重要. 该模式应从传统的汽车制造商、经销商与车辆所有者之间的一次性交易，转变为一种能够创造新收入来源，并抵消车辆建造和维护高昂成本的稳定、长期运营模式.

1）车计算的新生态系统. 车计算技术的商业模型如图 7 所示，它描绘了由 5 个核心实体组成的多层次生态系统.

① 车载服务的客户. 包括保险公司、政府机构、科技公司和智能家居解决方案提供商等. 它们从车计算技术提供的服务中受益，例如道路状况监测（如检测坑洞、黑冰和消防栓泄漏）以及高清地图生成，并为这些服务提供资金支持.

② 服务提供商. 利用车计算提供了一系列新型移动服务. 例如，亚马逊 AWS 提供了一整套强大的车载服务和解决方案，推动汽车行业的数字化转型.

③ 基础设施供应商. 支持车计算开发、验证和扩展或满足通信需求的硬件组件. 例如，Cavnue 致力于开发基础设施，以提高道路安全和减少交通拥堵.

④ 汽车制造商. 负责收集和分发车辆数据，并利用 SOAFEE[17,27]、AUTOSAR[28] 和 OpenVDAP[29] 等平台或架构，促进数据的开放性和分析利用.

⑤ 车辆所有者. 车主通过利用其车辆的计算能力和生成的数据，可向其余 4 个核心实体提供服务，从而获得额外收入.

2）示例应用：电动汽车与慢速充电. 新型商业模式的一个典型应用场景是电动汽车的充电. 通常情况下，电动汽车的充电时长为 8～12 h，适合在夜间或工作日进行充电[24]. 这一场景为车计算的应用提供了良好的机会. 当电动汽车在慢速充电时，可以处理数据密集型任务，例如检测影响车道或道路边界的施

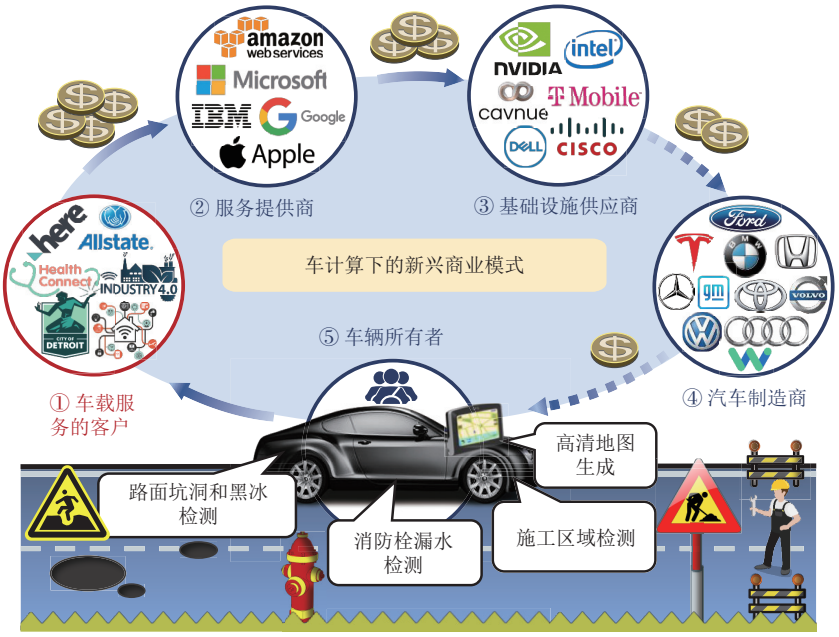


Fig. 7 A potential business model to support vehicle computing[19]

图 7 支持车计算的潜在商业模式[19]

工变化，并相应更新高清地图. 电动汽车还可以识别并报告基础设施问题(如受损消防栓)给相关部门. 地方政府等服务客户将为这些服务付费，从而确保所有参与方(包括服务提供商、基础设施供应商、汽车制造商和车辆所有者)能够通过财务收益或服务改进获益，如图 7 所示.

图 7 所示的商业模型涵盖了一个共生生态系统，每个参与者不仅为技术和服务的进步做出贡献，还分享经济利益，为车计算技术的可持续发展铺平道路.

3)传统模式和新兴商业模式的区别. 传统的商业模式仅涉及汽车制造商、汽车销售商和车辆所有者(消费者)这三方，通常是一次性的高额度交易. 然而，在新兴的商业模式中，参与方更加多样化，包括了车载服务的客户(如保险公司、政府机构、科技公司)、服务提供商(如亚马逊和微软公司)、基础设施提供商(如通信厂商、硬件厂商)、汽车制造商和车辆所有者. 在新兴商业模式下，车辆上的第三方应用能够为汽车制造商、车辆所有者、服务提供商和基础设施提供商带来新的收入来源，并抵消建造和维护车辆的高昂成本.

传统上，汽车制造商主要通过一次性的车辆销售赚取利润. 而在新模式下，这些制造商将以接近成本价格的售价出售汽车，并依靠后续的服务和应用来实现盈利. 对于消费者而言，过去他们需要支付昂贵的购车费用，而如今他们可以根据个人需求选择不同的服务和应用配置，通过帮助第三方运营服务和应用来获取额外收益. 这种转变不仅增加了消费者的选择权，还推动了汽车生态向更为开放和多元化的方向发展. 表 1 总结了传统商业模式和车计算的新兴商业模式的主要不同点.

3 车计算范式下的移动计算

3.1 计算硬件

车计算的到来得益于硬件的快速发展和多样化. 其中，图形处理单元(GPU)、数字信号处理器(DSP)、

Table 1 Traditional Business Model and Emerging Business Model in Vehicle Computing

表 1 传统商业模式与车计算的新兴商业模式对比

特征	传统商业模式	车计算的新兴商业模式
主要参与方	汽车制造商、汽车销售商、顾客	车载服务的客户、服务提供商、基础设施提供商、汽车制造商和车辆所有者
销售模式	一次性购车交易	销售汽车只是一个初始交易，它将进一步引发一系列的后续服务和应用盈利
汽车销售价格	相对较高，包括利润和生产成本	相对较低，接近甚至低于生产成本
消费者体验	购车费用高昂，较少定制选择	可根据个人需求定制服务和应用配置，并且获取额外收益
车厂利润来源	主要来自车辆销售利润	来自后续服务、应用配置和第三方合作的收益

现场可编程门阵列(FPGA)和专用集成电路(ASIC)等关键硬件组件发挥了重要作用[30].

举例来说, NVIDIA DRIVE AGX[31]是一款强大的系统芯片, 达到每秒320万亿次运算(TOPS). Xilinx推出的Zynq UltraScale+ MPSoC ZCU104是一款基于FPGA的汽车级产品, 具备可扩展的架构[32]. 它能够实现每瓦处理14张图像, 远高于Tesla K40 GPU的每瓦处理4张图像的效率[33]. 德州仪器的TDA3x系统芯片(SoC)提供了基于DSP的车计算应用解决方案. 相较于ARM Cortex-A15 CPU, 在计算机视觉任务性能方面提升了8倍, 同时保持较低的功耗[34]. 基于ASIC的Mobileye的EyeQ5 [35]在仅10瓦的功率内实现了24 TOPS的计算能力. 此外, 谷歌公司的张量处理单元(TPU)可提供高达420 TFLOPS的计算性能.

3.2 实时操作系统

实时操作系统(RTOS)已成为车辆系统架构中的关键组件[36]. RTOS在ECUs中有广泛应用, 管理如油门和刹车控制等功能.

QNX和VxWorks是汽车领域中最突出的2种商用RTOS. QNX采用极简内核架构, 处理CPU调度、进程间通信、中断重定向和定时. 另一方面, VxWorks通常用于需要安全和保护认证的场景[37]. VxWorks具有多架构支持, 如Intel、POWER和ARM, 并采用为具有严格时间限制的应用量身定制的实时内核.

其次, RTLinux采用微内核方法, 旨在支持硬实时要求. 其调度程序允许在整个Linux系统中进行抢占, 促进了与常规Linux进程并行执行任务[38]. 同时, NVIDIA DRIVE OS提供了一个基础软件栈, 包括嵌入式RTOS和NVIDIA Tensor RT等, 用于增强车计算应用所需的机器学习算法性能[39].

3.3 中间件系统

1)机器人操作系统(ROS). ROS[40-41]在大多数车计算的架构中发挥着关键作用. 它引入了主题、服务、参数服务器和动作库4种主要通信模式, 实现了车辆系统内不同组件之间的高效交互.

ROS目前有2个主要版本: ROS1[42]和ROS2[43]. 尽管二者均支持网络连接, 但主要用于内网环境, 安全功能相对薄弱. ROS2在此基础上增加了基本的安全认证和访问控制功能, 允许通过配置文件实现访问控制[44]. 然而, ROS2对这些配置文件的检查不够严格, 存在篡改或越权的风险[45]. 此外, 在车路协同等应用场景中, 实现协同数据的访问控制仍面临挑战.

2)开源项目. Autoware[46]基于ROS, 作为著名的开源中间件, 为车辆系统提供了基础支持. 百度公司的Apollo Cyber RT是一个开源的实时操作系统和计算框架, 专为满足车辆系统的高性能和低延迟需求而设计[47]. 此外, 全球合作项目AutoSAR还提供了一套标准化的软件架构, 提供了通用的开发标准[28].

4 车计算范式下的通信和网络技术

4.1 主流车载通信机制

1)从LTE到6G的演变. 移动网络从LTE到6G不断演进发展. 4G增强了通信能力, 支持固定1 Gbit/s和移动100 Mbit/s的速度; 5G进一步提高了下载速度(平均494.7 Mbps), 并显著改善了通信性能. 然而, 其复杂的基础设施、高成本和信号穿透限制仍是城市环境中的障碍. 6G则旨在克服这些问题, 提供超高频、近乎零延迟以及更高的网络可靠性和容量, 目前仍在研发中, 尚需技术突破与标准化.

2)专用短程通信(DSRC). DSRC是一种专为联网车辆设计的通信协议. DSRC基于IEEE 802.11p标准, 工作频率为5.9 GHz. SAE J2735标准中定义了15种消息类型, 涵盖了车辆位置、地图信息、紧急警报等信息[48]. 受限于可用带宽, DSRC传输消息的大小和频率都较低.

3)C-V2X. C-V2X(包括V2X和蜂窝网络)提供了Uu和PC5接口. 前者用于支持时延不敏感的信息共享, 后者则用于实现车辆与RSU及其他车辆之间的快速互联互通. 相比DSRC, C-V2X具备更远的通信距离、更高的可靠性、更大的带宽以及更优的拥塞控制能力. 2020年, 美国联邦通信委员会(FCC)决定取消DSRC频段分配, 转而将其划拨给C-V2X, 标志着美国正式放弃DSRC, 转向C-V2X. 此外, 福特公司和高通公司在安娜堡和圣地亚哥的现场测试显示, C-V2X在各项性能指标上均优于DSRC[49].

4.2 通信架构

1)集中式架构. 集中式架构, 通常被称为客户端-服务器模型[50]. 在这种架构中, 车辆作为客户端发起服务请求, 中央服务器处理这些请求并返回必要的响应. 该模型依赖于服务器提供的标准化接口, 使客户端车辆无需了解服务提供系统的复杂性.

2)分布式架构. 分布式或点对点(P2P)架构促进车辆之间的直接交互, 适合共享出行服务和P2P车辆租赁. 例如, HireGo[51]在私人车辆租赁中应用区块链技术, 以确保安全性和信任. 由本田和通用汽车主导的Mobility Open Blockchain Initiative(MOBI)[52]则将区块链技术整合到车辆充电网络中, 以支持电动

汽车之间的直接电力交换.

3)发布/订阅架构. 发布/订阅架构提供了网络内不同实体之间异步通信的动态框架. 在该架构中, 发布者发送消息, 订阅者接收相关信息, 确保所有与特定主题相关的消息对订阅者均可访问, 增强了事件驱动应用的可扩展性和响应性. 此架构不仅适用于响应式交通管理, 还支持事故警报、施工通知、停车位可用性、燃料价格和定向广告.

4)广播架构. 广播架构适用于实时交通信息的传播. 此架构允许向特定区域内的车辆广播重要的数据, 形成信息接力链. 广播架构强调信息的及时、高效传播, 但在平衡消息紧急性与网络容量方面仍面临挑战.

4.3 软件定义通信和安全

1)软件定义通信. 当前, C-V2X 用于连接车辆、边缘和云服务, 车内则通过蓝牙、WiFi、CAN 总线等网络实现车机系统与底层控制的互联, 并为用户提供接入功能. 软件定义通信(SDN)从底层赋能车辆的通信功能, 赋能的具体方面如下:

①从车路云协同的角度, 传统 SDN 方式可提升车路云协同效率[53]. 例如, SDN 和 MEC 架构下的 V2X 卸载和资源分配方案有效降低了系统开销[54]. 软件定义通信的 C-V2X 还可整合 V2X 和 5G 功能, 特别是在 V2X 信号薄弱的区域, 通过 5G 实现紧急消息的快速广播, 并为车机系统提供更大的带宽. ②从车内网络与应用的角度, 通过软件定义可更方便地打通不同网络, 为乘客提供无缝连接, 甚至聚合乘客的网络以增强车辆功能. 例如, 一种利用乘客移动设备提升车机计算的平台也可通过多种网络提高任务卸载效率[55]. ③从安全角度, SDN 能够在需要时开放相关接口, 并实施安全认证、访问控制和入侵检测等技术, 实现按需开放与安全防护.

2)软件定义通信安全. 国家正着手制定 C-V2X 的安全标准, 以规范和保障数据安全[56]. 目前已有大量关于 C-V2X 安全通信的研究, 提出了多种解决方案, 如基于区块链的车联网数据共享方法[57]. 在自动驾驶时代, 车内数据的安全性也成为关键问题, 尤其是通过 C-V2X 传输后的数据追溯和溯源亟待解决.

5 车计算范式下的传感和数据采集与存储技术

5.1 车载传感器

1)摄像头. 作为核心传感器, 摄像头提供二维视觉信息, 但在恶劣天气或夜间的可靠性较低, 处理遮挡和光线不足也有难度. 摄像头数据量大, 每秒 20～40 MB, 需要强大的处理与存储支持.

2)雷达. 雷达通过无线电波识别物体的位置、距离和速度, 通常在 24 GHz 或 77 GHz 下工作, 后者精度更高、干扰更少. 雷达在各种天气和低能见度环境下性能稳定、数据量较小, 每秒 10～100 KB.

3)激光雷达. 激光雷达利用激光扫描构建三维环境图像, 性能优异, 适用于恶劣环境, 但高成本限制了其广泛应用. 激光雷达每秒生成 10～70 MB 数据, 需实时处理支持.

4)超声波传感器. 超声波传感器通过 20 kHz 以上声波进行近距离检测, 每秒生成 10～100 KB 的数据, 稳定适用于恶劣天气和低光环境, 且价格较低, 适合停车辅助和近距离障碍物检测.

5)GPS/IMU. GPS 与惯性测量单元(IMU)共同构成车辆定位的基础. 传统 GPS 精度约为 1 m, 但在隧道等封闭环境下失效, 且刷新率较低. IMU 无需卫星信号, 可在遮挡环境中保持运行, 但精度低且误差随时间累积. 通常通过卡尔曼滤波器融合 GPS 与 IMU 数据, 实现精确定位[58].

5.2 数据来源

1)数据特点. 车辆数据分为实时数据和历史数据. 实时数据通过发布/订阅系统流动, 允许多个应用共享数据流, 提高响应速度和灵活性; 历史数据则存储以供参考, 分为结构化数据(如 GPS 坐标)和非结构化数据(如视频片段).

2)数据集和基准测试. 一些数据集以其覆盖面和基准测试在研究中广受认可: KITTI[59] 数据集支持立体视觉和 3D 检测; Cityscapes[60] 数据集提供城市街景的精细语义标注; BDD100K[61] 数据集包含多样化的驾驶条件; DDD17[62] 数据集则是首个动态像素传感器数据集, 涵盖各种驾驶和控制信息.

3)数据标注. 在监督学习中, 数据标注的准确性直接影响模型效果. 常用标注方法包括: ① 边界框: 通过矩形的左上角和右下角标示物体位置; ② 多边形分割: 捕捉不规则物体轮廓; ③ 语义分割: 为每个像素分配类别; ④ 3D 长方体: 提供物体体积与相对位置; ⑤ 关键点和地标检测: 标示物体几何关键点. 此外, 一些注释工具因其直观界面和多功能性而广受欢迎, 如 MakeSense.AI, LabelImg, VGG Image Annotator, LabelMe, Scalabel 和 RectLabel 等[63-68].

5.3 车辆中的数据存储

1)数据压缩. 现代车辆数据的快速增长推动了图像压缩技术的应用, 以缓解存储需求[69-71]. 图像压缩分为无损和有损 2 种: 无损压缩保证图像完全重

建但效率较低；有损压缩则牺牲部分数据以实现更高的压缩比，适合图像整体可解释性不受影响的场景[72]. 尽管数据压缩取得了不错的进展，这些压缩策略在车辆数据集中的应用仍需进一步探索.

2）车辆数据记录系统. 传统车辆数据记录系统通过 CAN 总线记录温度、制动、油门、发动机性能和速度等信号，但在管理现代车辆传感器生成大量数据方面日益不足. 为优化数据利用，地理哈希表（GHT）系统通过将键值对映射到地理坐标并分配给最近的传感器节点，实现数据持久性和网络负载均衡[73]. 分层云存储架构则增强了数据管理和系统演进，探讨了部署、虚拟化和数据安全[74]. 网络附加安全磁盘架构提供了线性扩展的存储带宽，适用于并行数据挖掘[75]，而 Diamond 存储架构通过早期丢弃策略加快未索引数据搜索，优化了过滤器评估[76].

3）车计算存储系统. 车计算的进步带来了大量数据管理需求. 为应对这一挑战，HydraSpace 计算存储系统采用多层存储架构和高效压缩算法，有效处理车辆传感器生成的数据[77]. OpenVDAP 架构的核心为驾驶数据集成器，通过分层存储系统平衡传感器与应用的数据存储和处理能力[29].

5.4 车辆数据访问控制

车计算中数据安全至关重要. 访问控制技术通过对数据访问实体进行授权和鉴权来保护数据安全. 基于引用监视器架构的 AC4AV 系统针对车辆的多种数据和应用场景，提供了有效的访问控制[78].

1）车辆数据访问模式. 车辆数据分为实时数据和历史数据，前者包括 GPS、摄像头、激光雷达和方向盘状态等传感数据，后者为存储的实时数据和部分应用数据（如高清地图）. 实时数据多采用发布订阅模式，通过话题访问，而历史数据则通过数据库、文件或 Web 接口提供统一访问.

2）访问控制框架. 车计算数据多样，应用场景复杂，增加了访问控制的难度. AC4AV 提出了一种分层架构，包括操作控制服务层、访问控制引擎层和策略日志存储层[78]. 该框架在数据访问点嵌入操作控制服务以捕获请求，并在访问控制引擎层对数据类型进行统一命名，在策略日志层支持数据持久化和快速检索. AC4AV 还支持多种访问控制模型（如基于证书/角色的控制），开放接口供用户自定义策略，为车路云协同环境下的分布式访问控制提供支持，并通过鉴权模块实现全面的安全控制.

5.5 车辆编程接口

车计算应用开发缺乏统一标准. 特拉华大学 CAR 实验室提出标准化接口 VPI[79]，涵盖硬件、数据、计算、服务和管理 5 层，以便开发者专注接口开发并应用其专业知识. 如图 8 所示，VPI 可划分为 5 层，包括硬件层、数据层、计算层、服务层和管理层. 硬件层 VPI 屏蔽底层差异，为高级软件开发提供统一接口，使开发者能控制摄像头、雷达等设备. 数据层 VPI 负责数据收集与管理，提供实时和历史数据访问. 计算层 VPI 在数据层之上，支持路径规划、车道检测等自动驾驶功能. 服务层 VPI 为开发者提供车载服务定制化和第三方协同功能. 管理层 VPI 整合设备连接、访问控制等功能，支持车载应用的个性化和安全管理.

6 车计算范式下能量的消耗、存储和发送

6.1 能源消耗

集成先进传感器和高性能计算设备大幅增加了

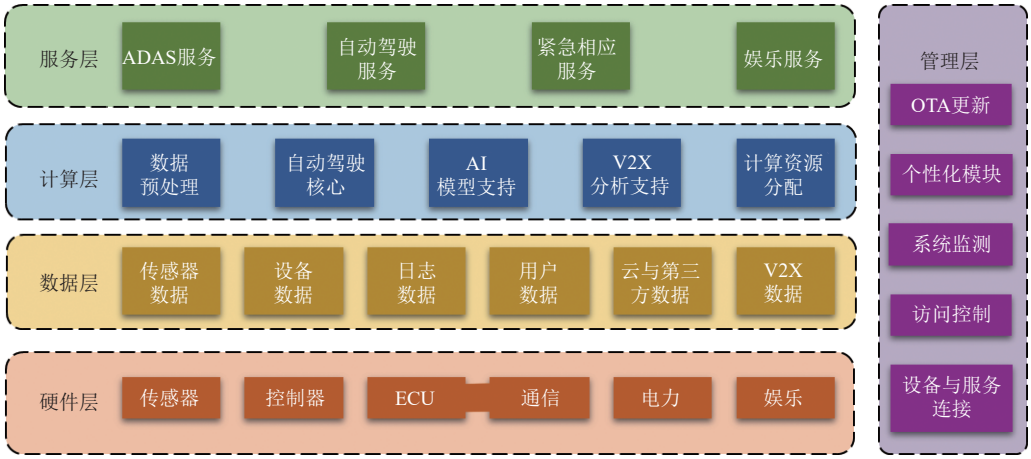


Fig. 8 Architecture design of VPI[79]

图 8 VPI 架构设计[79]

车辆能耗. 例如，NVIDIA Drive PX Pegasus 在 500 W 功率下提供 320 INT8 TOPS 计算，再加上传感器、天线和电池等组件，整体能耗可超 1 000 W，安装冗余系统后能耗接近 2 000W[31]. 此外，美国电动汽车的年能源需求估计达 180 TWh[80]，相当于主流科技公司(如谷歌)数据中心年电力使用总量的 15 倍[81]. 解决高功耗问题不仅要考虑电池管理系统、散热解决方案，还需符合车辆架构的空间限制和自动化要求.

6.2 能源存储

电动汽车不仅消耗能源，还具备作为移动能源存储单元的潜力. 一辆设计良好的电动汽车可生成 60 Hz 交流电，功率超过 10 kW，足以满足约 10 个普通家庭的电力需求. 连接电网后，电动汽车电池能够提供类似于固定储能系统的电网服务. 随着更多电动汽车接入电网，管理其充电对电网的影响变得至关重要. 此外，住宅和商业充电器的普及使这些车辆在电网紧急或高负荷期间成为可靠的电力来源.

6.3 节能设计与策略

为了提高在车计算系统的能效，研究分为 2 个主要的方向[82]：① 针对资源受限的车计算单元，设计专用的机器学习加速器架构. ② 探索基于硬件的全栈优化方法，从硬件层到应用层采取整体方法，优化在车辆计算单元上执行机器学习的性能.

1)节能加速器架构. ① GPU 加速. NVIDIA 推出了一系列适用于车辆应用的 GPU 节能产品. NVIDIA® DRIVE™ PX2 耗电 250W，而 NVIDIA® Jetson™ 平台功耗范围为 5～30W. ② DianNao 加速器. DianNao 家族是为机器学习算法设计的节能硬件加速器，核心为神经功能单元(NFU)，辅以 SRAM 缓冲器，有效减少能耗达 21 倍. 其扩展版 ShiDianNao 专注于边缘视觉算法，能效提高了 60 倍. ③ 谷歌公司的张量处理单元(TPU). TPU 通过将网络划分为基本计算单元并利用脉动矩阵乘法器来优化神经网络处理，显著提高了能效，比传统的 CPU 和 GPU 高出 30~80 倍. ④神经形态处理器架构. IBM TrueNorth 和 Intel Loihi 引入了神经形态处理器，利用基于硅的技术模拟大脑功能，通过可编程的神经元和突触显著减少能耗到毫瓦级别，同时能够实时处理神经网络应用，为车辆及其他领域的节能计算提供了有前途的路径.

2)基于硬件的全栈优化. ① FPGA 优化. 高效语音识别引擎 (ESE)[83] 利用 FPGA 在移动平台上加速长短期记忆(LSTM)模型. 通过负载平衡感知的剪枝方法，其能效比 CPU 和 GPU 分别高出 40 倍和 11.5 倍. ②节能推理引擎. 高效推理引擎 EIE 在压缩深度神经网络处理中避免 DRAM 访问、跳过零激活，能效比 CPU、GPU 和 DaDianNao 分别高出 24 000 倍、3 400 倍和 19 倍，展现了在车辆硬件上进行深度学习计算的节能潜力. ③深度神经网络的协同设计框架. ADMM-NN 采用交替方向乘子法(ADMM)优化深度神经网络，通过权重剪枝和量化提升模型压缩和能效，突显算法-硬件协同设计在车辆计算中的重要性.

6.4 车辆到电网

1)车辆到电网技术现状. 车辆到电网(V2G)技术推动了车辆作为电源的创新应用，尤其适用于灾后恢复或电力短缺时期. 支持 V2G 的电动汽车可以独立运行或与现有储能系统结合，为电网提供电力，缓解供应中断并维持电网稳定. 通过智能充放电策略，车辆在电价低时充电、高时放电，平衡电网负载，有效应对突发停电. 预计到 2027 年，V2G 市场将达到 174.3 亿美元，年均增长率约为 48%.

整合 V2G 能力可以为电动汽车车主带来多重利益，包括新的收入来源. 车主可以将车辆电池中的能量货币化，抵消电动汽车相对于汽油车较高的初始购买价格. 例如，一辆参与电力储备服务的丰田 RAV4 EV 每年可赚取高达 2 554 美元[84]. 此外，作为移动能源存储，V2G 车辆在高可再生能源生成期间储存多余能量，并在低生成期间释放，确保稳定可靠的能源供应. 并且，V2G 车辆作为分散的能源存储和备用系统，有助于减少电网扰动相关成本，这些成本占美国电力费用的 5%～10%，每年约 120 亿美元[85].

2)V2G 的电力流动. 电网运营商利用通信系统来组织和调节电动汽车电池与电网之间的电力交换.

① 单向 V2G.单向 V2G 技术仅允许电力从电网流向电动汽车，通过简单的充电速率控制器即可实现高效且具成本效益的单向充电. 通过鼓励车主在非高峰时段充电，电力公司不仅能更有效地管理高峰负载，还为车主提供节省成本的机会. 此外，优化技术的应用使单向 V2G 能够进一步提升收益并减少排放.

② 双向 V2G.双向 V2G 技术允许电动汽车与电网进行双向电力交换，通过在低需求期存储能量、高需求期供应电力，帮助削减峰值负载，实现负载平衡. 同时，通过调节充电器的 DC 连接电容和精确控制切换，提供无功率支持，协助电网电压调节和减少损耗，并利用电动汽车的移动储能特性缓解风能和太阳能的波动性. 尽管双向 V2G 潜力巨大，但频繁的充放电会加速电池退化，所需的充电基础设施也复杂且成

本较高. 此外, 车主通常偏好保持较高电量以备不时之需, 这也限制了双向 V2G 的广泛应用, 凸显出平衡能源管理和用户便利性的必要性.

7 车计算的测试

7.1 车计算系统的度量标准

1)准确性. 在物体和车道检测中, 交并比(IOU)和平均精度(mAP)等标准至关重要. 这些标准衡量检测到的物体边界或车道标记与其真实位置之间的精确度. 在车辆操作方面, 准确性涉及预期的刹车或转向动作与车辆实际执行的动作之间的一致性.

2)延迟. 配备先进计算系统的车辆必须实时执行控制功能以确保安全. 例如, 一辆在城市环境中以 40 km/h 速度自主导航的车辆应具备在每米内有效调整控制的能力. 为实现这一目标, 整个计算流程的响应时间必须小于 90 ms[86]. 这要求每个计算模块在严格的时间限制内运行, 以确保所需的响应速度和最佳的安全措施.

3)能耗. 车计算系统的能耗很关键, 特别是由车载电池供电时. 专用于自动驾驶功能的计算系统可能显著影响车辆的续航里程, 报告显示续航里程可能减少多达 30%[80]. 此外, 高能耗还会导致大量热量的产生, 进一步影响系统性能和车辆的整体效率.

4) 成本. 在自动驾驶时代下, 高级汽车的成本可高达 30 万美元, 其中传感器、计算和通信设备的成本约占三分之二[87]. 此外, 培训人员、车辆维护、保险、停车和维修等运营成本通常高于传统车辆, 突显了这些先进系统在广泛部署中的经济挑战.

5) 可靠性. 可靠性涵盖了维护操作完整性的广泛考虑[69]. 这包括计划最坏情况下的执行时间, 以适应任何不可预见的延误. 此外, 车辆必须配备中断或紧急干预的协议. 可靠性还涉及车辆组件, 包括传感器、计算和通信设备、算法及其系统集成[88]. 开发应对潜在故障的策略对于车计算系统至关重要.

6) 隐私. 由于车辆收集了大量的周围环境的数据, 隐私问题尤为重要. 保护个人隐私, 如匿名化视频数据中捕获的行人面孔和车牌是至关重要的. 此外, 保护驾驶数据的所有权和查看权, 也要求系统在支持数据访问、存储和传输的时候保证数据的隐私性.

7) 安全. 网络攻击可以针对计算框架的任何组件. 保护传感机制、确保通信安全、维护数据完整性和保障控制系统的安全性至关重要[89]. 系统必须经过严格的安全评估, 以确认其对潜在网络威胁的抵御能力.

7.2 车计算的仿真

1)仿真器. AMoDeus[90] 平台用于模拟和分析自主交通系统, 专注于评估交通管理策略和车辆调度算法. 相比之下, CARLA[91] 和 SUMO[92] 提供了更动态的 3D 和 2D 仿真环境, 并具备部分感知能力, 具有全面接口和用户友好性. 然而, 从仿真到实际应用的过渡仍具挑战, 通常受资源可用性和现实条件精确复制的限制.

2)数字孪生. 数字孪生在道路系统规划、交通管理和算法开发中起到关键作用. 通过创建物理资产或系统的虚拟副本, 数字孪生技术支持深入分析、仿真和决策, 并在以下方面展现显著优势: 在算法测试中, 加速仿真迭代, 实现部署前的微调; 在交通管理中, 可在安全的虚拟环境中测试多种场景; 此外, 数字孪生的高级可视化能力支持远程测试, 使测试灵活、可访问, 能从任何地点详细查看结果.

3)案例研究. 美国特拉华大学的 CAR 实验室通过 D-STAR 项目在校区内创建了一个动态的测试环境, 如图 9 所示. D-STAR 项目包含 2 个核心模块: ① 智能交叉口(OpenIntersection). 该系统集成车联网技术、多种传感器系统(包括高分辨率摄像头和激光雷达)和异构计算平台, 实时处理传感器数据, 利用 AI 技术预测交通流并优化信号, 从而提升交通效率和安全性. ② BlueICE 仿真框架. 该框架提供高精度的车计算测试环境, 模拟物理动力学、复杂交通、天气条件和多样化道路用户行为, 使研发团队在无物理风险的环境中测试和优化算法, 加速实际应用的部署进程.

7.3 车计算的测试平台

1)单车和多车测试. 车计算测试分为单车和多车场景的研究. 单车研究[93-94] 成功地在遥控车上应用 AUTOSAR 设计原则实现了实时动态定位. 单车测试对于测试控制器设计非常有用, 但未考虑对更广泛交通系统的影响. 相比之下, 多车系统研究[95] 显著推进了车辆群体形成战术和自动驾驶系统的发展.

2)案例研究. 如图 10 所示, 特拉华大学的 CAR 实验室推出了室内自动驾驶测试平台 ICAT[96]. ICAT 利用数字孪生技术, 并使用 CARLA 和 SUMO 的高级仿真能力, 在模拟环境和物理测试中促进单车和多车自动驾驶的研究.

8 车计算的技术挑战和机遇

要全面实现车计算, 需要不同领域的研究人员

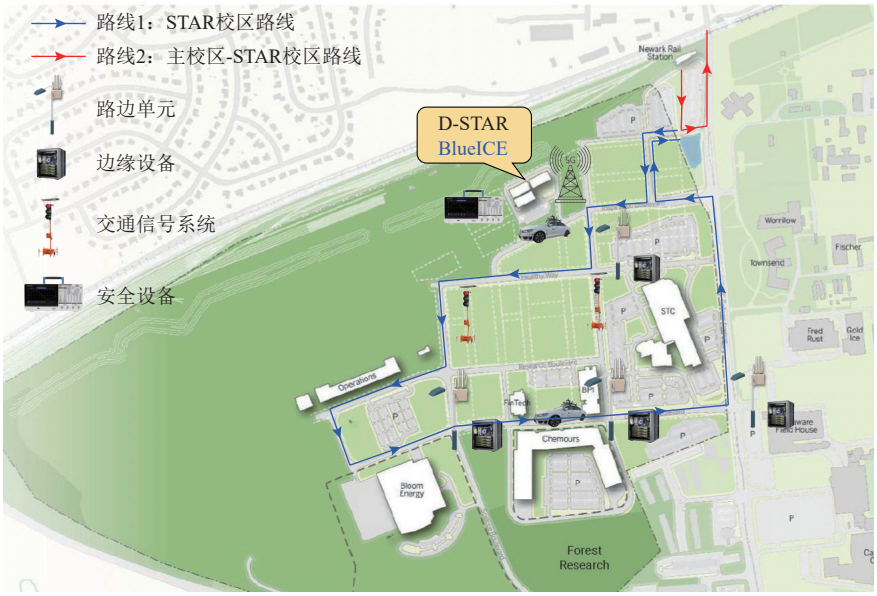


Fig. 9 An overview of the D-STAR

图 9 D-STAR 概览

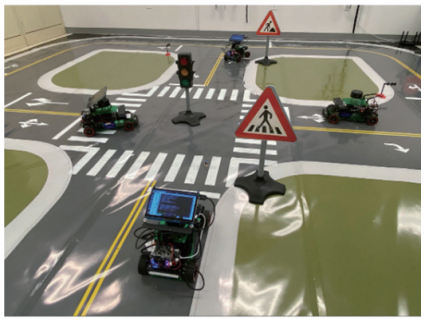


Fig. 10 A panoptic view of ICAT platform[96]

图 10 ICAT 平台概览[96]

协同合作来解决计算、通信、能量管理(包括能量的消耗和存储)、感知和数据存储等方面的技术挑战.

8.1 在车上进行计算所面临的挑战

1)在真实世界场景中应用机器学习

① 安全问题的标准化. 机器学习算法的性能存在可变性, 对相机图像的细微修改, 如裁剪或光照变化, 可能显著影响车辆在物体检测和分割任务中的表现[97]. 然而, ISO 26262[98] 安全标准并未涵盖深度学习技术, 因为它是在 AI 兴起之前制定的. 因此, 需要修订安全标准, 以充分应对 AI 技术在车辆中的集成[99].

② 可扩展训练. 机器学习模型需要在涵盖所有可能应用场景的庞大数据集上进行训练. 这带来了漫长的训练周期和巨大的数据需求. 为应对这些挑战, 已经引入了协同训练[9]、模型压缩技术[100] 和轻量级机器学习算法[101] 等创新方法. 此外, 模型训练所需的多样化对象的精确标注也是一个瓶颈[61].

③ 全面测试. 机器学习从庞大数据集中学习, 并将知识封装在复杂的加权特征组合中, 使得全面测试变得不直观且具有挑战性[102]. 据估计, 要验证自动驾驶系统的灾难性故障率, 需要约 10 亿小时的车辆操作测试, 并需重复测试以达到统计显著性[99].

2)车辆操作系统的挑战. 开发车辆操作系统在整合硬件资源和车载服务方面至关重要, 但兼容性是主要障碍. Autoware 的应用受限于支持的传感器种类, 且仅适用于少数商用车辆. 现代车辆通常配备多达 150 个 ECUs, 通过 CAN 总线通信, 增加了通用操作系统整合的复杂性和安全性[103]. 为应对 CAN 总线上的虚假消息问题, 已有伪造网络攻击检测技术[104], 但由于 CAN 协议的安全性和商业机密性, 统一操作系统的开发仍面临挑战. 为应对这些挑战, AUTOSAR[28] 于 2003 年由领先的汽车制造商和供应商发起, 旨在创建汽车开放系统架构. 尽管 AUTOSAR 有助于推动开源操作系统的发展, 但汽车行业对开放其操作系统持谨慎态度, 限制了其在研究和教育领域的可访问性.

3)硬件成本. 传统车辆与支持车计算的车辆在硬件成本上差异显著. 在美国, 制造一辆标准非豪华车辆的平均成本约为 3 万美元, 而配置车计算系统的成本可能攀升至约 25 万美元[87]. 这种差异主要源于额外的硬件需求, 如专用通信设备、先进计算系统、线控驱动机制、摄像头、激光雷达和雷达等. 此外, 为确保车计算的可靠性和安全性, 通常需要硬件冗余, 如备用电源[105]. 同时, 高昂的维护成本可能抑制

用户兴趣并影响制造商的盈利[106]. 不过，随着技术的成熟和创新，成本有望逐步下降. 例如，谷歌 Waymo 将高端激光雷达的成本降至约 7 500 美元，特斯拉的雷达成本约为 8 000 美元[87].

4）实验平台. 车计算技术的开发和优化需要在现实环境中进行复杂的测试和评估，使得实验平台成为研发过程的关键环节. 然而，构建和维护实验车辆的成本极高，且还需承担停车、保险和维护的持续费用. 此外，现场测试的法律和监管要求也增加了难度. 一个有效的策略是创建更具成本效益的车计算研发平台，适用于室内或低速应用. 例如，HydraOne 是专为室内场景设计的开源平台，提供全面的可编程性，适合研究与开发[107]. 同样，PerceptIn 的 DragonFly 在 40 英里每小时以下速度下运行，价格在 4 万美元以下[108]. 这些平台降低了车计算研究的门槛，促进了广泛的探索与创新.

5）车辆系统组件的异质性. 另外一个重大挑战在于车辆系统组件的多样性和复杂性. 这些组件包括微控制器、实时处理器、微处理器和专用加速器（如 GPU 和 FPGA），其能力和限制各不相同. 某些微控制器设计缺乏重新编程或通用计算调整的灵活性. 例如，许多 ECU 的功能是固定的，并通过数字版权管理系统（包括硬件锁）保护，以防止未经授权的修改. 此外，从不同供应商采购的设备的异质性给建立统一的性能分析方法带来了挑战. 缺乏标准化的方法来衡量和比较这些设备的性能，阻碍了架构级评估和基准测试的有效性，使车辆系统和算法的开发和优化变得复杂.

6）车载应用服务的实时约束. 车载服务中一个重要的挑战在于遵守管理车辆任务执行的实时约束[109]. 这些约束确保关键操作在特定时间内完成，以保证安全和效率. 车辆系统中的实时任务大致分为 3 类：硬实时任务、软实时任务和非实时任务. 未按时完成硬实时任务可能导致严重后果，因此必须优先考虑并确保资源分配. 软实时任务的截止日期未满足可能使结果无关紧要或较少有用，而非实时任务的延迟通常只会降低结果的质量或效用，而不会立即造成伤害.

例如，在正常驾驶条件下，处理车辆摄像头的视频流通常属于软实时任务或非实时任务. 然而，在即将发生碰撞等关键情况下，这些视频流的快速分析则上升为硬实时任务. 这时，车辆系统必须迅速优先分配资源，以确保在规定时间内执行碰撞规避算法，从而突显了车辆计算中任务优先级的动态性质.

7）车辆系统中的安全和隐私保障. 随着车辆系统的复杂性提升以及车计算与边缘和云服务的深度融合，对安全和隐私保护的需求愈发迫切. 基于内生安全的安全架构已被提出，以增强车辆系统的防护[110]，而硬件辅助隔离的软件架构也变得极其重要. 这些架构需符合嵌入式行业的严格安全和隐私标准，确保车辆系统的安全集成. 此外，建立统一的安全协议简化了车辆生态系统的开发和维护，并在分布式车辆环境中实现了可靠的安全和隐私保护操作，促进车载系统与外部计算资源的无缝协作.

8.2 车载通信面临的挑战

1）超可靠的低延迟通信以及提供按照用户需求的差别服务. 推进车计算技术的首要难题之一是确保车辆在移动中的可靠且快速的网络连接，这对于在各种无线网络和云服务之间切换时维持服务连续性至关重要. 目前，LTE/4G/5G 和 WiFi 在高速移动环境下效果较差. DSRC 仍面临带宽和范围的限制. 虽然 C-V2X 作为一种新兴的通信机制，能在高密度的动态交通场景中展现可靠、低延迟和高效通信的潜力，但它目前仍处于初期阶段，成本较高且部署有限. 目前，包括宝马、福特、爱立信和高通在内的领先车厂和芯片制造商正越来越多地投资 C-V2X. 此外，车计算还需要预留特定带宽和低延迟，以实现按需差别服务，如支持紧急情况下的远程操作. 当前网络尚不足以满足此需求，但在下一代通信技术中亟需引入这一机制.

2）传输带宽和经济费用. 随着车计算技术的发展，上传（数据共享）和下载（软件/固件更新）的传输成本变得越来越重要. 预计 1 000 万辆车辆每年可能传输超过 20 PB 的数据，费用可能超过 10 亿美元[111]. 具体来说，单辆车每天平均上传约 8 GB 数据，下载内容主要包括车载信息娱乐系统更新，每次约 500 MB[112]. 移动网络运营商如 AT&T 和 Verizon 提供专用定价计划，包括“按使用量计费”和“无限预付数据计划”，后者每月起价约为 20 美元. 全球 1 GB 移动数据的平均成本约为 8.53 美元，但各国差异显著[112]. 保守估计每季度 1 次的软件或固件更新，1 辆车的年上传费用约为 240 美元，下载费用为 25 美元[112]. 当扩展到数百万辆车时，传输成本将成为巨大的经济负担. 边缘计算能显著减少数据传输需求[113]，降低 10%～30% 的成本并提升 10%～20% 的运营效率[114].

3）网络攻击保护. 车计算面临重放攻击（如模仿电子钥匙）、欺骗攻击（误导行驶路径）和针对机器学习算法的新型威胁[115-116]. 例如，攻击者可通过照片绕

过面部识别系统,或使用欺骗性图像阻碍模型的准确检测[117]. 此外,量子计算的发展对现有加密标准构成威胁,探索后量子密码学因此成为车联网和智能交通的重点[118]. 高级传感器和技术的引入也扩大了潜在攻击面. 应对策略包括多源数据验证(如用摄像头确认雷达检测),以及利用硬件辅助的可信执行环境,在隔离空间中执行关键任务,以提升安全性[119]. 然而,可信执行环境面临内存限制,如何在保证性能的前提下优化资源分配仍是挑战. 建立能够预见攻击、评估安全性并促进恢复的保护系统至关重要.

4)利用智能基础设施提升车辆运营效率. 智能基础设施在以下关键领域对车计算有显著的提升作用:

① 服务提供. 智能基础设施集成无线射频识别传感器,能够高效调度停车服务请求,优化利用率,减少车辆寻找服务位置所需的时间和计算资源[120].

② 交通信息共享. 获取最新交通信息对于安全驾驶至关重要. 配备通信功能的路边单元是向车辆传播和收集交通数据信息的关键节点. 这些数据支持天气警报、地图更新和道路事件检测等功能[121].

③ 任务卸载. 实时管理车内工作负载需要大量计算能力和能源. 将密集计算任务卸载到基础设施可以提升处理速度并节省车辆能量[122]. 确定最优任务卸载顺序以充分利用智能基础设施资源,是一个亟待解决的问题[123].

5)与人类驾驶员和远程操控人员的互动挑战. 在自动驾驶时代,人类驾驶车辆和远程辅助操控驾驶仍不可避免[124],这提出了促进车辆与人类驾驶员或操作员之间有效互动的挑战. 为解决这一问题,许多研究致力于改进速度和控制预测,以增强人机交互效果[125]. 深度强化学习在处理多车复杂情景方面表现出强大潜力,然而,这些方法大多局限于模拟或特定控制环境中. 利用车联网通信是有前景且直观的方法[124],可以显著增强人机交互.

8.3 车载数据管理和存储面临的挑战

1)车辆数据异常检测. 在自动驾驶时代,车计算依赖数据分析做出决策,但面临广泛的攻击面. 除对软硬件接口和运行程序的直接攻击外,车计算系统还易受外部干扰类攻击. 当前的研究展示了多种传感数据攻击方式,数据与算法层面的“入侵”已成为车计算系统特有的安全挑战[115]. 深度学习和大模型在异常检测中可提供一定的安全保障,但也对计算能力提出较高要求. 车计算环境下的算力有限,且网络环境多样、异构且动态,统一模型易造成资源浪费. 因此,设计轻量级机制来及时检测异常数据、确保驾驶安全,仍是一个待解决的关键问题.

2)多传感器数据同步和存储. 车辆数据来源多样,从车辆自身的传感器到其他车辆的传感器、路边单元和外部平台,数据同步是一个显著的挑战. 数据生成频率差异明显,需要战略性的数据同步方法. 此外,不同传感器的时间戳精度差异也加剧了这一问题. 因此数据存储系统是否应处理同步任务是一个关键决定,这直接影响到车载服务中数据利用的效率和效果.

3)车载数据隐私和存储. 现代车辆配备了大量传感器收集海量数据,其中可能包含个人身份信息,如面部特征和车牌号码. 这引发了重大的隐私问题,需要在数据共享前进行匿名化或屏蔽以确保个人隐私不被泄露. 此外,还需要建立稳健的访问控制系统,区分授权和未经授权的数据请求,防止恶意应用篡改关键数据,因为未经授权的访问可能导致错误的驾驶决策,危及乘客安全. 但由于数据的复杂性和访问模式的多样性,实现这一目标非常具有挑战性. 同时,数据存储的可访问性对于事故后的分析、法律调查以及遵守监管标准至关重要,因此开发健全的数据保留和检索机制变得尤为重要.

4)车载数据的安全和存储. 允许外部开发者直接在车辆上部署应用程序带来了复杂的安全挑战. 车辆必须实施机制来保护车辆系统的完整性和运行在其上的应用程序的机密性. 可信执行环境可以通过创建安全隔离的空间来执行第三方应用程序. 首先,需要确保车辆主系统对第三方应用程序执行的具体功能(如与执法相关的操作)保持不知情. 其次,系统必须防范来自第三方应用程序的潜在威胁,以保障车辆主系统或其他共驻应用的安全. 这需要建立一个平衡的框架,既能安全执行外部应用程序,又能保持车辆系统的整体安全和隐私. 此外,制定统一的数据管理协议和有效的安全保证对于实现车计算生态系统中不同系统和实体之间的互操作性至关重要.

5)数据共享的可靠性和冗余. 在车辆系统中提供可靠的数据共享服务面临许多挑战. 分布式系统常见的增强可靠性方法是维护多个数据副本,关键是确定最佳副本数量和存储位置,以在效率和冗余之间取得平衡. 路边单位作为存储车辆数据副本的仓库,能够扩展车辆的数据存储能力. 然而,保持车辆与路边单元之间的数据一致性由于通信链接的不可靠性而更加复杂. 确保在通信不确定性和存储限制下的数据可靠性和一致性,对于提高车计算技术的健壮性至关重要.

8.4 优化能源使用和车辆到电网集成的挑战

1)能源效率. 部署众多传感器和执行自动驾驶所需复杂算法是导致车辆能源消耗增加的原因. 高性能计算单元本身能耗较高. 当为提高车辆应用的可靠性而实施冗余系统时, 这一能耗可能会成倍数增加. 为了解决这一问题, 需要整合并促进能源高效利用的方法, 协调车辆的能源管理、计算基础设施和电池管理系统.

2)V2G 挑战. 尽管 V2G 技术为增强电动汽车与电网的互动带来前景, 但其普及仍面临电池技术、商业模式、可行性和监管框架等多重挑战. 频繁的充放电循环可能加速电池磨损, 因此, 开发耐受 V2G 操作的电池至关重要. 此外, 构建可行的商业模式是推动 V2G 广泛采用的关键. V2G 引入双向能量流动, 需设计合理的补偿机制, 如能源折扣和电池更换条款, 确保车辆向电网供能时得到公平回报. 商业可行性评估同样不可或缺, 利益相关方需判断投资双向充电、能量管理系统和用户平台的回报是否合理. 理解能源供应商和移动服务运营商的经济激励对于实现 V2G 的能源管理潜力至关重要.

9 总　　结

本文系统探讨了车计算这一新兴且前景广阔的领域, 包括其核心概念、技术支持、商业模式、技术挑战和潜在机遇. 首先, 回顾了汽车软件和计算架构的变革, 讨论了车计算范式、四层计算架构及其技术潜力, 展示了车计算在多种实际场景中的应用潜力, 并深入分析了其新兴商业模式, 强调其推动汽车生态系统走向开放多元化发展的作用.

随后, 本文详细分析了车计算的关键技术, 涵盖移动计算、通信技术、传感与数据采集存储、能耗管理等内容, 并进一步讨论了车计算的测试方法、系统度量标准以及仿真技术(如数字孪生)等.

最后, 本文总结了车计算在多个方面的挑战与机遇, 包括机器学习复杂性、操作系统技术、硬件成本控制、实验平台开发、系统异质性、实时服务约束、安全和隐私保障. 讨论重点涉及低延迟通信、传输带宽、网络防护、智能基础设施效率、人机互动、数据异常检测、数据隐私与安全、数据共享的可靠性, 以及能源效率与 V2G 技术等问题.

作者贡献声明: 鲁思迪主要负责论文的撰写和图表的制作, 确保内容的准确性和连贯性, 并对论文进行了多轮编辑和校对, 提升了论文的整体质量; 何元恺负责查阅相关文献, 系统总结前人的研究成果, 深入分析现有研究的不足, 同时对初稿进行了审阅与反馈, 提出了多项关键性改进建议; 施巍松为本研究的整体方向提供了战略性指导, 协调了研究团队的各项工作, 确保研究目标清晰、内容严谨, 并在论文核心思想、研究框架及图表设计方面做出了重要贡献.

参　考　文　献

[1] AECC. Distributed computing in an AECC system[EB/OL]. [2021-08-13] https://aecc.org/resources/publications/

[2] Allied Market Research. Connected car market size, share, growth & trends analysis report by technology, connectivity solution, service, end-use, and segment forecasts, 2020−2027[EB/OL]. [2024-05-15]. https://reports.valuates.com/market-reports/ALLI-Manu-3Z1/connected-car

[3] Lu Sidi, Shi Weisong. Vehicle as a mobile computing platform: Opportunities and challenges[J]. IEEE Network, 2024, 38(6): 493−500

[4] Makke O, Gusikhin O. Connected vehicle prognostics framework for dynamic systems[C]//Proc of the Third Int Scientific Conf “Intelligent Information Technologies for Industry”(IITI’18). Berlin: Springer, 2019: 3−15

[5] Trilla A, Dersin P, Cabre X. Estimating the uncertainty of brake pad prognostics for high-speed rail with a neural network feature ensemble[C]//Proc of Prognostics and Health Management Society. Durham, NC: PHM Society, 2018: 1−7

[6] Guo Hongyan, Shen Chen, Zhang Hui, et al. Simultaneous trajectory planning and tracking using an MPC method for cyber-physical systems: A case study of obstacle avoidance for an intelligent vehicle[J]. IEEE Transactions on Industrial Informatics, 2018, 14(9): 4273−4283

[7] Chien S I J, Kuchipudi C M. Dynamic travel time prediction with real-time and historic data[J]. Journal of Transportation Engineering, 2003, 129(6): 608−616

[8] Gunter G, Gloudemans D, Stern R E, et al. Are commercially implemented adaptive cruise control systems string stable?[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 22(11): 6992−7003

[9] Lu Sidi, Yao Yongtao, Shi Weisong. CLONE: Collaborative learning on the edges[J]. IEEE Internet of Things Journal, 2020, 8(13): 10222−10236

[10] Luo Yichen, Xu Daoxuan, Zhou Gang, et al. Impact of raindrops on camera-based detection in software-defined vehicles[C]//Proc of IEEE Int Conf on Mobility, Operations, Services and Technologies (MOST). Piscataway, NJ: IEEE, 2024: 193–205

[11] Narote S P, Bhujbal P N, Narote A S, et al. A review of recent advances in lane detection and departure warning system[J]. Pattern

Recognition, 2018, 73: 216−234
[12] Lee H, Kang M, Song J, et al. The detection of black ice accidents for preventative automated vehicles using convolutional neural networks[J]. Electronics, 2020, 9(12): 2178−2191
[13] Fan R, Ozgunalp U, Hosking B, et al. Pothole detection based on disparity transformation and road surface modeling[J]. IEEE Transactions on Image Processing, 2019, 29: 897−908
[14] Sumaiya, Jafarpourmarzouni R, Lu Sidi, et al. Enhancing real-time inference performance for time-critical software-defined vehicles[C]//Proc of IEEE Int Conf on Mobility, Operations, Services and Technologies (MOST). Piscataway, NJ: IEEE, 2024: 101−113
[15] Chen Junzhou, Lu Sidi. An advanced driving agent with the multimodal large language model for autonomous vehicles [C]//Proc of IEEE Int Conf on Mobility, Operations, Services and Technologies (MOST). Piscataway, NJ: IEEE, 2024: 1−11
[16] Goh C C, Kamarudin L M, Sudin S, et al. Cloud-based in-vehicle air quality monitoring system with GSM module[J]. Journal of Telecommunication, Electronic and Computer Engineering, 2018, 10(1-14): 77−81
[17] Lu Sidi, Ammar N, Ganlath A, et al. A comparison of end-to-end architectures for connected vehicles[C]//Proc of the Fifth Int Conf on Connected and Autonomous Driving (MetroCAD). Piscataway, NJ: IEEE, 2022: 72−80
[18] Bauer L. Smart vehicle architecture: A sustainable approach to building the next generation of vehicles[J]. APTIV White Paper, 2020
[19] Lu Sidi, Shi Weisong. Vehicle computing: Vision and challenges[J]. Journal of Information and Intelligence, 2023, 1(1): 23−35
[20] Bodei C, De Vincenzi M, Matteucci I. From hardware-functional to software-defined vehicles and their security issues[C]//Proc of IEEE Int Conf on Industrial Informatics (INDIN). Piscataway, NJ: IEEE, 2023: 1−10
[21] Halder S, Ghosal A, Conti M. Secure over-the-air software updates in connected vehicles: A survey[J]. Computer Networks, 2020, 178: 107343
[22] Lu Sidi, Shi Weisong. Vehicle Computing: From Traditional Transportation to Computing on Wheels[M]. Cham, Switzerland: Springer, 2024
[23] Liu Liangkai, Zhang Xingzhou, Qiao Mu, et al. SafeShareRide: Edge-based attack detection in ridesharing services[C]//Proc of IEEE/ACM Symp on Edge Computing (SEC). Piscataway, NJ: IEEE, 2018: 17−29
[24] Lu Sidi, Shi Weisong. The emergence of vehicle computing[J]. IEEE Internet Computing, 2021, 25(3): 18−22
[25] ASCE. 2021 Report Card for America's Infrastructure[EB/OL]. [2024-05-10]. https://infrastructurereportcard.org/cat-item/bridges/
[26] Miller T W, Dickson P R. On-line market research[J]. International Journal of Electronic Commerce, 2001, 5(3): 139−167
[27] Spencer M, Andreozzi M M. Real-time portability in software-defined systems[C]//Proc of Cyber-Physical Systems and Internet of Things Week Workshops. New York: ACM, 2023: 177−178
[28] AUTOSAR. AUTOSAR Website[EB/OL]. [2024-05-15]. https://www.autosar.org/
[29] Zhang Qingyang, Wang Yifan, Zhang Xingzhou, et al. OpenVDAP: An open vehicular data analytics platform for CAVs[C]//Proc of IEEE 38th Int Conf on Distributed Computing Systems (ICDCS). Piscataway, NJ: IEEE, 2018: 1310−1320
[30] Liu Liangkai, Lu Sidi, Zhong Ren, et al. Computing systems for autonomous driving: State of the art and challenges[J]. IEEE Internet of Things Journal, 2020, 8(8): 6469−6486
[31] Xavier M N. A new brain for self-driving, AI, and AR cars[EB/OL]. (2018-01-07)[2019-03-18]. https://www.slashgear.com/meet-nvidia-xavier-a-new-brain-for-self-driving-ai-and-ar-cars-07513987/
[32] Xilinx. Enabling next generation ADAS and AD systems[EB/OL]. [2024-05-17]. https://www.xilinx.com/products/silicon-devices/soc/xa-zynq-ultrascale-mpsoc.html
[33] Xilinx. Zynq UltraScale+ MPSoC ZCU104 Evaluation Kit[EB/OL]. [2024-05-25]. https://www.xilinx.com/products/boards-and-kits/zcu104.html
[34] Texas Instruments. Texas instruments TDA[EB/OL]. [2024-06-14]. http://www.ti.com/processors/automotive-processors/tdax-adas-socs/overview.html
[35] MobileEye. The evolution of EyeQ: The system-on-chip for automotive applications[EB/OL]. [2024-06-14]. https://www.mobileye.com/our-technology/evolution-eyeq-chip/
[36] Nvidia. NVIDIA drive AGX systems[EB/OL]. [2024-05-20]. https://www.nvidia.com/en-us/self-driving-cars/drive-platform/hardware/
[37] Wind River Systems, Inc. VxWorks: Real-time operating system for the intelligent edge[EB/OL]. [2024-06-01]. https://www.windriver.com/products/vxworks/
[38] Sato H, Yakoh T. A real-time communication mechanism for RTLinux[C]//Proc of IEEE Int Conf on Industrial Electronics, Control and Instrumentation (IECON). Piscataway, NJ: IEEE, 2000: 2437−2442
[39] Nvidia. NVIDIA DRIVE - Software[EB/OL]. [2024-05-20]. https://developer.nvidia.com/drive/drive-software
[40] Quigley M, Conley K, Gerkey B, et al. ROS: An open-source robot operating system[C]//Proc of IEEE Int Conf on Robotics and Automation (ICRA) on Open Source Software. Piscataway, NJ: IEEE, 2009: 3(3.2): 5−10
[41] Foote T, Scott K. ROS Community Metrics Report[EB/OL]. [2023-11-23]. http://download.ros.org/downloads/metrics/metrics-report-2020-07.pdf
[42] Teixeira R R, Maurell I P, Drews P L J. Security on ROS: Analyzing and exploiting vulnerabilities of ROS-based systems[C]//Proc of Latin American Robotics Symp (LARS) and Brazilian Symp on Robotics (BSR). Piscataway, NJ: IEEE, 2020: 1−6
[43] Open Robotics. Open Robotics: Home of ROS and Gazebo[EB/OL]. [2024-05-21]. https://www.openrobotics.org
[44] White R, Christensen H I, Caiazza G, et al. Procedurally provisioned access control for robotic systems[C]//Proc of IEEE/RSJ Int Conf on Intelligent Robots and Systems (IROS). Piscataway, NJ: IEEE, 2018: 1−9
[45] Benhamouda F, Lepoint T, Loss J, et al. On the (in) security of ROS[J]. Journal of Cryptology, 2022, 35(4): 25−49
[46] Autoware. Welcome to the Autoware Foundation[EB/OL]. [2023-

05-21]. https://www.autoware.org/

[47] Baidu. Apollo Cyber[EB/OL]. [2024-05-14]. https://github.com/ApolloAuto/apollo/tree/master/cyber

[48] Kenney J B. Dedicated Short-Range Communications (DSRC) standards in the United States[J]. Proceedings of the IEEE, 2011, 99(7): 1162−1182

[49] 5G Automotive Association. V2X Technology Benchmark Testing[EB/OL].[2024-01-31].https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/fcc_usdot_cv2x_-_v2.14_w_video1.pdf

[50] Olaverri-Monreal C. Autonomous vehicles and smart mobility related technologies[J]. Infocommunications Journal, 2016, 8(2): 17−24

[51] Scott-Briggs A. HireGo – Peer to peer car hire and car sharing platform[EB/OL]. [2024-05-17]. https://techbullion.com/hirego-peer-to-peer-car-hire-car-sharing-platform/

[52] Gresham T. Honda, GM-led Group Develops Global Blockchain Standard for EV Grid Integration[EB/OL]. [2024-06-01]. https://www.utilitydive.com/news/honda-gm-led-group-develops-global-blockchain-standard-for-ev-grid-integra/587263/

[53] Shao Wenjuan, Shen Qingguo. Survey of software defined D2D and V2X communication[J]. Journal on Communications, 2019, 40(4): 179−194 (in Chinese)
(邵雯娟, 沈庆国. 软件定义的 D2D 和 V2X 通信研究综述[J]. 通信学报, 2019, 40(4): 179−194)

[54] Zhang Haibo, Wang Zixin, He Xiaofan. V2X offloading and resource allocation under SDN and MEC architecture[J]. Journal on Communications, 2020, 41(1): 114−124 (in Chinese)
(张海波, 王子心, 贺晓帆. SDN 和 MEC 架构下 V2X 卸载与资源分配[J]. 通信学报, 2020, 41(1): 114−124)

[55] Wang Lin, Zhang Qingyang, Li Youhuizi, et al. MobileEdge: Enhancing on-board vehicle computing units using mobile edges for CAVs[C]//Proc of IEEE 25th Int Conf on Parallel and Distributed Systems (ICPADS). Piscataway, NJ: IEEE, 2019: 470−479

[56] Chen Xiao, Huang Muhong, Tian Yifan, et al. Internet of vehicles data sharing scheme via blockchain sharding[J]. Journal of Computer Research and Development, 2024, 61(9): 2246−2260 (in Chinese)
(陈骁, 黄牧鸿, 田一凡, 等. 基于分片区块链的车联网数据共享方案[J]. 计算机研究与发展, 2024, 61(9): 2246−2260)

[57] Liu Xuejiao, Cao Tiancong, Xia Yingjie. Research on efficient and secure cross-domain data sharing of IoV under blockchain architecture[J]. Journal on Communications, 2023, 44(3): 186−197 (in Chinese)
(刘雪娇, 曹天聪, 夏莹杰. 区块链架构下高效的车联网跨域数据安全共享研究[J]. 通信学报, 2023, 44(3): 186−197)

[58] Liu Shaoshan, Li Liyun, Tang Jie, et al. Creating Autonomous Vehicle Systems[M]. Berlin: Springer, 2020

[59] Garcia-Garcia A, Orts-Escolano S, Oprea S, et al. A review on deep learning techniques applied to semantic segmentation[J]. arXiv preprint, arXiv: 1704.06857, 2017

[60] Tampuu A, Matiisen T, Semikin M, et al. A survey of end-to-end driving: Architectures and training methods[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 33(4): 1364−1384

[61] Yu F, Chen Haofeng, Wang Xin, et al. BDD100K: A diverse driving dataset for heterogeneous multitask learning[C]//Proc of IEEE/CVF Conf on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: IEEE, 2020: 2636−2645

[62] Binas J, Neil D, Liu S C, et al. DDD17: End-to-end DAVIS driving dataset[J]. arXiv preprint, arXiv: 1711.01458, 2017

[63] Skalski P. Makesense. ai[EB/OL]. [2024-06-10]. https://www.makesense.ai/

[64] HumanSignal. LabelImg[EB/OL]. [2024-05-21]. https://github.com/tzutalin/labelImg

[65] Dutta A. VGG image annotator[EB/OL]. [2024-02-28]. https://gitlab.com/vgg/via

[66] MIT Computer Science and Artificial Intelligence Laboratory. LabelMe[EB/OL]. [2024-04-12]. http://labelme.csail.mit.edu/Release3.0/

[67] Scalabel. Scalabel: A versatile web-based visual data annotation tool[EB/OL]. [2024-06-03]. https://github.com/scalabel

[68] Kawamura R. RectLabel: An offline image annotation tool for object detection and segmentation[EB/OL]. [2024-06-12]. https://rectlabel.com/

[69] Sabaliauskaite G, Liew L S, Cui J. Integrating autonomous vehicle safety and security analysis using STPA method and the six-step model[J]. International Journal on Advances in Security, 2018, 11(1&2): 160−169

[70] Toderici G, Vincent D, Johnston N, et al. Full resolution image compression with recurrent neural networks[C]//Proc of IEEE Conf on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: IEEE, 2017: 5306−5314

[71] Zhou Yangming, Wang Chengyou, Zhou Xiao. DCT-based color image compression algorithm using an efficient lossless encoder[C]//Proc of the 14th IEEE Int Conf on Signal Processing (ICSP). Piscataway, NJ: IEEE, 2018: 450−454

[72] Di Sheng, Cappello F. Fast error-bounded lossy HPC data compression with SZ[C]//Proc of IEEE Int Parallel and Distributed Processing Symp (IPDPS). Piscataway, NJ: IEEE, 2016: 730−739

[73] Ratnasamy S, Karp B, Yin Li, et al. GHT: A geographic Hash table for data-centric storage[C]//Proc of the 1st ACM Int Workshop on Wireless Sensor Networks and Applications (WSNA). New York: ACM, 2002: 78−87

[74] Zeng Wenying, Zhao Yuelong, Ou Kairi, et al. Research on cloud storage architecture and key technologies[C]//Proc of the 2nd Int Conf on Interaction Sciences: Information Technology, Culture and Human (ICIS). Piscataway, NJ: IEEE, 2009: 1044−1048

[75] Gibson G A, Nagle D F, Amiri K, et al. A cost-effective, high-bandwidth storage architecture[J]. ACM SIGOPS Operating Systems Review, 1998, 32(5): 92−103

[76] Huston L, Sukthankar R, Wickremesinghe R, et al. Diamond: A storage architecture for early discard in interactive search[C]//Proc of the 3rd USENIX Conf on File and Storage Technologies (FAST). Berkeley, CA: USENIX, 2004, 4: 73−86

[77] Wang Ruijun, Liu Liangkai, Shi Weisong. HydraSpace: Computational data storage for autonomous vehicles[C]//Proc of IEEE 6th Int Conf on Collaboration and Internet Computing (CIC). Piscataway, NJ: IEEE, 2020: 70−77

[78] Zhang Qingyang, Zhong Hong, Cui Jie, et al. AC4AV: A flexible and

dynamic access control framework for connected and autonomous vehicles[J]. IEEE Internet of Things Journal, 2020, 8(3): 1946−1958

[79] Wu Baofu, Zhong Ren, Wang Yuxin, et al. VPI: Vehicle programming interface for vehicle computing[J]. Journal of Computer Science and Technology, 2024, 39(1): 22−44

[80] Teraki. Autonomous cars' big problem: The energy consumption of edge processing reduces a car's mileage with up to 30%[EB/OL]. [2020-07-15]. https://medium.com/@teraki/energy-consumption-required-by-edge-computing-reduces-a-autonomous-cars-mileage-with-up-to-30-46b6764ea1b7

[81] Bryce R. How Google powers its monopoly with enough electricity for entire countries (2020)[EB/OL]. [2023-11-25]. https://www.forbes. com/sites/robertbryce/2020/10/21/googles-dominance-is-fueled-by-zambia-size-amounts-of-electricity

[82] Kumar M, Zhang Xingzhou, Liu Liangkai, et al. Energy-efficient machine learning on the edges[C]//Proc of IEEE Int Parallel and Distributed Processing Symp Workshops (IPDPSW). Piscataway, NJ: IEEE, 2020: 912−921

[83] Han Song, Kang Junlong, Mao Huizi, et al. ESE: Efficient speech recognition engine with sparse LSTM on FPGA[C]//Proc of ACM/SIGDA Int Symp on Field-Programmable Gate Arrays (FPGA). New York: ACM, 2017: 75−84

[84] Parsons G R, Hidrue M K, Kempton W, et al. Willingness to pay for vehicle-to-grid (V2G) electric vehicles and their contract terms[J]. Energy Economics, 2014, 42: 313−324

[85] Kempton W, Tomić J. Vehicle-to-grid power fundamentals: Calculating capacity and net revenue[J]. Journal of Power Sources, 2005, 144(1): 268−279

[86] Kato S, Takeuchi E, Ishiguro Y, et al. An open approach to autonomous vehicles[J]. IEEE Micro, 2015, 35(6): 60−68

[87] LeVine S. What it really costs to turn a car into a self-driving vehicle[EB/OL].[2024-05-12].https://qz.com/924212/what-it-really-costs-to-turn-a-car-into-aself-driving-vehicle

[88] Orrick A, McDermott M, Barnett D M, et al. Failure detection in an autonomous underwater vehicle[C]//Proc of IEEE Symp on Autonomous Underwater Vehicle Technology (AUV). Piscataway, NJ: IEEE, 1994: 377−382

[89] Zeng Kexiong, Liu Shinan, Shu Yuanchao, et al. All your GPS are belong to us: Towards stealthy manipulation of road navigation systems[C]//Proc of the 27th USENIX Security Symp (USENIX Security). Berkeley, CA: USENIX, 2018: 1527−1544

[90] Ruch C, Hörl S, Frazzoli E. AMoDeus, a simulation-based testbed for autonomous mobility-on-demand systems[C]//Proc of the 21st Int Conf on Intelligent Transportation Systems (ITSC). Piscataway, NJ: IEEE, 2018: 3639−3644

[91] Dosovitskiy A, Ros G, Codevilla F, et al. CARLA: An open urban driving simulator[C]//Proc of the 1st Annual Conf on Robot Learning (CoRL). Cambridge, MA: PMLR, 2017: 1−16

[92] Krajzewicz D, Hertkorn G, Rössel C, et al. SUMO (Simulation of Urban MObility) - an open-source traffic simulation[C]//Proc of the 4th Middle East Symp on Simulation and Modelling (MESM). San Diego, CA: SCS, 2002: 183−187

[93] Tian Zhaofeng, Shi Weisong. Design and implement an enhanced simulator for autonomous delivery robot[C]//Proc of the Fifth Int Conf on Connected and Autonomous Driving (MetroCAD). Piscataway, NJ: IEEE, 2022: 21−29

[94] Vedder B, Vinter J, Jonsson M. A low - cost model vehicle testbed with accurate positioning for autonomous driving[J]. Journal of Robotics, 2018, 2018(1): 4907536

[95] Pickem D, Glotfelter P, Wang Li, et al. The Robotarium: A remotely accessible swarm robotics research testbed[C]//Proc of IEEE Int Conf on Robotics and Automation (ICRA). Piscataway, NJ: IEEE, 2017: 1699−1706

[96] Tian Zhaofeng, He Yuankai, Tian Boyang, et al. ICAT: An indoor connected and autonomous testbed for vehicle computing[C]//Proc of IEEE Int Conf on Mobility, Operations, Services and Technologies (MOST). Piscataway, NJ: IEEE, 2024: 242−250

[97] Pan Xingang, Shi Jianping, Luo Ping, et al. Spatial as deep: Spatial CNN for traffic scene understanding[C]//Proc of AAAI Conf on Artificial Intelligence (AAAI). Palo Alto, CA: AAAI, 2018, 32(1): 7276−7283

[98] Hommes Q V E. Review and assessment of the ISO 26262 draft road vehicle-functional safety[R]. SAE Technical Paper, 2012

[99] Rao Qing, Frtunikj J. Deep learning for self-driving cars: Chances and challenges[C]//Proc of the 1st Int Workshop on Software Engineering for AI in Autonomous Systems. New York: ACM, 2018: 35−38

[100] Luo Ping, Zhu Zhenyao, Liu Ziwei, et al. Face model compression by distilling knowledge from neurons[C]//Proc of the AAAI Conf on Artificial Intelligence (AAAI). Palo Alto, CA: AAAI, 2016, 30(1): 3560−3566

[101] Howard A G, Zhu Menglong, Chen Bo, et al. MobileNets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint, arXiv: 1704.04861, 2017

[102] Koopman P, Wagner M. Challenges in autonomous vehicle testing and validation[J]. SAE International Journal of Transportation Safety, 2016, 4(1): 15−24

[103] Kato S, Tokunaga S, Maruyama Y, et al. Autoware on board: Enabling autonomous vehicles with embedded systems[C]//Proc of ACM/IEEE 9th Int Conf on Cyber-Physical Systems (ICCPS). Piscataway, NJ: IEEE, 2018: 287−296

[104] Wei Hongqian, Shi Peicheng, Zhang Youtong. Automotive cyber-security: Detection technique of masquerade attacks for the bus network[J]. Journal of Mechanical Engineering, 2024, 60(10): 476−486 (in Chinese)
(魏洪乾, 时培成, 张幽彤. 汽车信息安全: 面向总线网络的伪造攻击检测技术[J]. 机械工程学报, 2024, 60(10): 476−486)

[105] Sedgwick David. When driverless cars call for backup[EB/OL]. [2024-05-24]. https://www.autonews.com/article/20170218/OEM10/302209969/when-driverless-cars-call-for-backup

[106] Luft Alex. The Chevrolet Sonic's Days are numbered[EB/OL]. [2024-05-22]. https://gmauthority.com/blog/2020/07/the-chevrolet-sonics-days-are-numbered/

[107] Wang Yifan, Liu Liangkai, Zhang Xingzhou, et al. HydraOne: An indoor experimental research and education platform for CAVs[C]//Proc of the 2nd USENIX Workshop on Hot Topics in Edge Computing (HotEdge). Berkeley, CA: USENIX, 2019: 1−7

[108] VentureBeat. PerceptIn's self-driving vehicles go on sale in

November for $40, 000[EB/OL]. [2021-10-04]. https://venturebeat.com/2018/09/12/perceptins-self-driving-vehicles-go-on-sale-in-november-for-40000/

[109] Sumaiya, Jafarpourmarzouni R, Luo Yichen, et al. Towards real-time and efficient perception workflows in software-defined vehicles[J]. IEEE Internet of Things Journal, 2024: 1–18. DOI: 10.1109/JIOT.2024.3492801

[110] Wu Jiangxing. Endogenous security problems and countermeasures of intelligent connected vehicle[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2023, 35(3): 383−390 (in Chinese)
(邬江兴. 智能网联汽车内生安全问题与对策[J]. 重庆邮电大学学报(自然科学版), 2023, 35(3): 383−390)

[111] Sam A. Automotive over-the-air updates: A cost consideration guide[EB/OL]. [2022-08-19]. https://www.auroralabs.com/wp-content/uploads/2021/05/OTA_Update_Cost_Consideration_Guide_Apr2021.pdf

[112] Lida Tabibian. Is data transmission the new fuel? (online)[EB/OL]. [2024-05-18]. https://www.auroralabs.com/is-data-transmission-the-new-fuel/

[113] Shi Weisong, Zhang Xingzhou, Wang Yifan, et al. Edge computing: State-of-the-art and future directions[J]. Journal of Computer Research and Development, 2019, 56(1): 69−89 (in Chinese)
(施巍松, 张星洲, 王一帆, 等. 边缘计算: 现状与展望[J]. 计算机研究与发展, 2019, 56(1): 69−89)

[114] Gorkem Y. Edge computing: Operator strategies, use cases and implementation[EB/OL]. [2022-09-12]. https://www.cisco.com/c/dam/en/us/solutions/service-provider/edge-computing/pdf/white-papersp-analysys-mason-research.pdf

[115] Ren Kui, Wang Qian, Wang Cong, et al. The security of autonomous driving: Threats, defenses, and future directions[J]. Proceedings of the IEEE, 2019, 108(2): 357−372

[116] Kamkar S. Drive it like you hacked it: New attacks and tools to wirelessly steal cars[J]. DEF CON, 2015, 23: 10

[117] Komkov S, Petiushko A. AdvHat: Real-world adversarial attack on ArcFace face ID system[C]//Proc of the 25th Int Conf on Pattern Recognition (ICPR). Piscataway, NJ: IEEE, 2021: 819−826

[118] Bernstein D J, Lange T. Post-quantum cryptography[J]. Nature, 2017, 549(7671): 188−194

[119] Ning Zhenyu, Zhang Fengwei, Shi Weisong, et al. Position paper: Challenges towards securing hardware-assisted execution environments[C]//Proc of Hardware and Architectural Support for Security and Privacy (HASP) Workshop. New York: ACM, 2017: 1−8

[120] Pala Z, Inanc N. Smart parking applications using RFID technology[C]//Proc of the 1st Annual RFID Eurasia Conf. Piscataway, NJ: IEEE, 2007: 1−3

[121] Al-Dweik A, Muresan R, Mayhew M, et al. IoT-based multifunctional scalable real-time enhanced road side unit for intelligent transportation systems[C]//Proc of IEEE 30th Canadian Conf on Electrical and Computer Engineering (CCECE). Piscataway, NJ: IEEE, 2017: 1−6

[122] Lin Li, Liao Xiaofei, Jin Hai, et al. Computation offloading toward edge computing[J]. Proceedings of the IEEE, 2019, 107(8): 1584−1607

[123] Dong Zheng, Shi Weisong, Tong Guangmo, et al. Collaborative autonomous driving: Vision and challenges[C]//Proc of 3rd IEEE Int Conf on Connected and Autonomous Driving (MetroCAD). Piscataway, NJ: IEEE, 2020: 17−26

[124] Lu Sidi, Zhong Ren, Shi Weisong. Teleoperation technologies for enhancing connected and autonomous vehicles[C]//Proc of IEEE 19th Int Conf on Mobile Ad Hoc and Smart Systems (MASS). Piscataway, NJ: IEEE, 2022: 435−443

[125] Hubmann C, Becker M, Althoff D, et al. Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles[C]//Proc of IEEE Intelligent Vehicles Symp (IV). Piscataway, NJ: IEEE, 2017: 1671−1678



Lu Sidi, born in 1994. PhD, PhD supervisor, assistant professor of computer science at William & Mary, USA. Her main research interests include edge computing, emerging mobility, and applied AI and data science, aimed at enhancing the reliability, scalability, security, and efficiency of networked, distributed, and autonomous systems.

鲁思迪, 1994 年生. 博士, 博士生导师, 美国威廉玛丽计算机科学助理教授. 主要研究方向为边缘计算、新兴移动性以及应用人工智能和数据科学, 旨在提高网络化、分布式和自治系统的可靠性、可扩展性、安全性和效率.



He Yuankai, born in 1997. PhD candidate. Member of the CAR Lab at the University of Delaware. His main research interests include computer vision, simulation, digital twin, C-V2X, and autonomous driving.

何元恺, 1997 年生. 博士研究生. 美国特拉华大学 CAR 实验室成员. 主要研究方向为计算机视觉、仿真、数字孪生、网联车 (C-V2X) 以及自动驾驶.



Shi Weisong, born in 1974. PhD, IEEE Fellow, Alumni Distinguished Professor of the Department of Computer and Information Sciences at the University of Delaware in USA. His main research interests include edge computing, autonomous driving, and connected health.

施巍松, 1974 年生. 博士, IEEE 会士, 美国特拉华大学计算机与信息科学系冠名教授. 主要研究方向为边缘计算、自动驾驶、智能健康.