



(12) 发明专利申请

(10) 申请公布号 CN 116901948 A

(43) 申请公布日 2023. 10. 20

(21) 申请号 202211537958.5

B60W 40/06 (2012.01)

(22) 申请日 2022.12.01

(30) 优先权数据

17/725,175 2022.04.20 US

(71) 申请人 辉达公司

地址 美国加利福尼亚州

(72) 发明人 D·尼斯特 李汉良 王一舟

R·阿维夫 B·亨克 J·额

A·阿克巴扎德

(74) 专利代理机构 北京市磐华律师事务所

11336

专利代理师 徐爱萍

(51) Int. Cl.

B60W 30/12 (2020.01)

B60W 60/00 (2020.01)

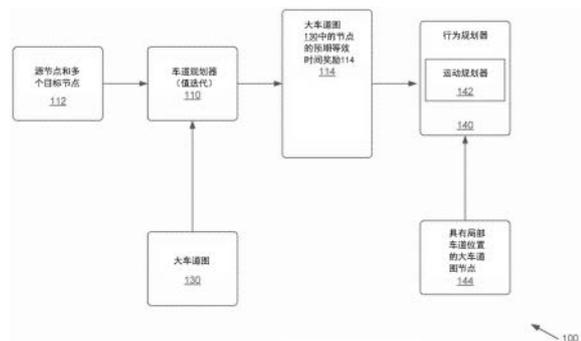
权利要求书4页 说明书39页 附图21页

(54) 发明名称

用于自主机器系统和应用的车道规划架构

(57) 摘要

本公开涉及用于自主机器系统和应用的车道规划架构。在各个示例中,提供了用于基于状态和概率动作空间生成车道规划器输出数据的车道规划器。驾驶系统(基于分层驾驶规划框架进行操作)包括车道规划器和其他规划和控制组件。车道规划器处理车道规划器输入数据(例如,大车道图、源节点、目标节点)以生成车道规划器输出数据(例如,预期时间奖励)。驾驶系统还可以包括路线规划器(例如,第一规划层),其操作以将车道规划器输入数据提供给车道规划器。车道规划器作为第二规划层操作,其至少部分地基于大车道图的状态和概率动作空间来处理车道规划器输入数据并且计算与从大车道图中的源节点导航至目标节点相关联的时间成本。



1. 一种处理器,包括:

一个或多个电路,用于:

生成第一车道图,对于所述第一车道图中的至少一个车道,所述第一车道图包括多个边和在所述至少一个车道内纵向间隔开的多个节点,每个边连接一对节点并且具有与其相关联的关联成本值;

使用确定性算法、所述第一车道图以及每个边的所述关联成本值来生成第二车道图,所述第二车道图包括从源节点到目标节点的一条或更多条路径,所述一条或更多条路径中的至少一条路径包括所述多个节点的子集和所述多个边的子集;

使用随机算法和所述第二车道图来生成第三车道图,所述第三车道图包括用于所述第三车道图的至少一个边的更新的关联成本值;以及

至少部分地基于每个边的所述更新的关联成本值来确定通过所述第三车道图的行驶路线。

2. 根据权利要求1所述的处理器,其中所述行驶路线的确定进一步至少部分地基于实时感知信息。

3. 根据权利要求1所述的处理器,其中至少部分地基于与所述至少一个边相关联的动作被成功完成的概率来计算该边的所述更新的关联成本值。

4. 根据权利要求1所述的处理器,其中所述确定性算法包括最短路径算法。

5. 根据权利要求4所述的处理器,其中所述最短路径算法包括迪杰斯特拉算法或A\*算法中的至少一个。

6. 根据权利要求1所述的处理器,其中所述随机算法包括强化学习算法。

7. 根据权利要求6所述的处理器,其中所述强化学习算法包括马尔可夫决策过程MDP算法,并且所述第三车道图被表示为MDP。

8. 根据权利要求1所述的处理器,其中使用行为规划器来执行最终行驶路线的确定。

9. 根据权利要求1所述的处理器,其中使用路线规划器来生成所述第一车道图,并且使用车道规划器来生成所述第二车道图和所述第三车道图中的至少一个。

10. 根据权利要求1所述的处理器,其中使用从目标奖励、花费的时间、花费的资源、不适、舒适、障碍物安全、路径服从或等待条件服从中选择的至少一个优化类别,来确定所述关联成本值或所述更新的关联成本值中的至少一个。

11. 根据权利要求1所述的处理器,其中所述多个节点中的至少一个节点与车辆状态相对应,并且所述多个边中的至少一个边与车辆动作相对应。

12. 根据权利要求11所述的处理器,其中所述车辆动作包括车道保持、变道、转弯、选择岔路或并线中的至少一个。

13. 根据权利要求1所述的处理器,其中至少部分地基于执行修改值迭代来计算至少一个边的所述更新的关联成本值,所述修改值迭代控制用于执行修改值迭代操作的迭代次数。

14. 根据权利要求1所述的处理器,其中所述至少一个边的关联成本值和所述至少一个边的更新的关联成本值与预期时间奖励相对应,所述预期时间奖励与执行与所述至少一个边相关联的动作相关联。

15. 根据权利要求14所述的处理器,其中用于计算所述预期时间奖励的至少一个输入

至少部分地通过将所述至少一个输入转换成基于时间的输入来计算。

16. 根据权利要求1所述的处理器,其中所述处理器被包括在以下的至少一者中:

用于自主或半自主机器的控制系统;

用于自主或半自主机器的感知系统;

用于执行模拟操作的系统;

用于执行光传输模拟的系统;

用于为3D资产执行协作内容创建的系統;

用于执行深度学习操作的系統;

使用边缘设备实现的系統;

使用机器人实现的系統;

用于执行会话AI操作的系統;

用于生成合成数据的系統;

包含一个或更多个虚拟机VM的系統;

至少部分地在数据中心中实现的系統;或

至少部分地使用云计算资源实现的系統。

17. 一种系統,包括:

一个或更多个处理器,其包括处理电路,所述处理电路用于:

至少部分地基于表示地图的地图数据生成确定性车道表示,所述确定性车道表示包括多个节点和在所述多个节点中的至少两个节点之间延伸的一个或更多个边,所述一个或更多个边中的至少一个边包括初始预期时间奖励;

至少部分地基于所述确定性车道表示生成概率车道表示,所述概率车道表示至少包括所述节点的子集和所述边的子集,所述边的子集中的至少一个边包括更新的预期时间奖励;以及

至少部分地基于所述概率车道表示来执行一个或更多个操作。

18. 根据权利要求17所述的系統,其中至少一个边与车辆动作相关联,并且执行所述一个或更多个操作包括根据一个或更多个车辆动作来控制车辆,所述一个或更多个车辆动作与通过所述概率车道表示的所选路径的一个或更多个边相关联。

19. 根据权利要求18所述的系統,其中所述一个或更多个车辆动作包括车道保持、变道、转弯、选择岔路或者并线中的至少一个。

20. 根据权利要求17所述的系統,其中所述确定性车道表示至少部分地基于对初始车道表示执行最短路径算法来生成。

21. 根据权利要求17所述的系統,其中所述概率车道表示至少部分地基于执行强化学习算法来生成。

22. 根据权利要求17所述的系統,其中所述概率车道表示包括与所述边中的至少一个边相关联的至少一个车辆动作的成功概率。

23. 根据权利要求17所述的系統,其中所述一个或更多个操作的执行进一步至少部分地基于实时感知信息。

24. 根据权利要求17所述的系統,其中所述系統被包括在以下中的至少一个中:

用于自主或半自主机器的控制系统;

用于自主或半自主机器的感知系统；  
用于执行模拟操作的系统；  
用于执行光传输模拟的系统；  
用于为3D资产执行协作内容创建的系统；  
用于执行深度学习操作的系统；  
使用边缘设备实现的系统；  
使用机器人实现的系统；  
用于执行会话AI操作的系统；  
用于生成合成数据的系统；  
包含一个或多个虚拟机VM的系统；  
至少部分地在数据中心中实现的系统；或  
至少部分地使用云计算资源实现的系统。

25. 一种方法,包括:

接收表示与车辆的行驶路线相关联的车道图的至少一部分的数据,所述车道图包括指示所述车道图的一个或多个车道内的一个或多个潜在位置的多个节点和连接所述多个节点的多个边,所述多个边中的至少一个边与所述一个或多个车道内的所述车辆的相应车辆动作相对应;

针对所述多个边中的每个边,至少部分地基于针对穿行由所述车道图表示的起始节点和连接至所述起始节点的节点之间的距离所预期的时间,来计算成本函数;

针对所述多个边中的每个边,计算与所述边相关联的至少一个车辆动作被成功完成的概率;

至少部分地基于与每个边相关联的所述成本函数,计算所述多个节点中的每个节点的时间奖励;以及

至少部分地基于所述多个节点中的每个节点的所述时间奖励,确定包括一个或多个候选车辆动作的一个或多个行驶路线。

26. 根据权利要求25所述的方法,其中计算所述多个节点中的每个节点的所述时间奖励是至少部分地基于执行修改值迭代,所述修改值迭代控制用于执行计算所述时间奖励的修改值迭代操作的迭代次数。

27. 根据权利要求25所述的方法,其中通过最初使用最短路径算法来识别从源节点到所述车道图上的任何节点的时间成本,来计算每个节点的所述时间奖励。

28. 根据权利要求25所述的方法,其中通过以下中的至少一个执行所述方法:

用于自主或半自主机器的控制系统；  
用于自主或半自主机器的感知系统；  
用于执行模拟操作的系统；  
用于执行光传输模拟的系统；  
用于为3D资产执行协作内容创建的系统；  
用于执行深度学习操作的系统；  
使用边缘设备实现的系统；  
使用机器人实现的系统；

用于执行会话AI操作的系统；  
用于生成合成数据的系统；  
包含一个或多个虚拟机VM的系统；  
至少部分地在数据中心中实现的系统；或  
至少部分地使用云计算资源实现的系统。

## 用于自主机器系统和应用的车道规划架构

### 背景技术

[0001] 为了使自主车辆有效地导航通过环境,这些自主车辆需要生成用于在两个或更多个位置之间导航的路线规划。例如,车辆的规划和控制组件可用于识别路线和动作,路线和动作可由车辆针对朝向目标的特定驾驶任务来执行。规划和控制组件(例如,路线规划器、车道规划器和/或行为规划器)可以使用由车辆生成和/或提供给车辆的感知和映射信息。例如,有向图可以包括可用于促进执行与规划和控制组件相关联的操作的感知和映射信息——表示驾驶规划状态和动作。感知和映射信息可在规划和控制组件的接口之间交换以支持导航车辆。

[0002] 驾驶或路线规划中的常规车道规划可以利用支持生成输出(例如,车道规划器输出数据)的输入(例如,车道规划器输入数据)进行操作。输入可包括地图、源和目标,其中源和目标可以是例如GPS位置——可选地与车道相关联——从第一点A到第二点B。输出可以包括可由车辆执行的路线和动作。动作可以包括用于执行的指令,如用于保持在车道中、进行车道变换、跟随车道合并或进行车道分离。传统车道规划器可被配置成使得只有一组有限的动作对于车辆是可用的,并且因此提供精确的动作以导航车辆,而不考虑汽车可能无法执行动作——例如,因为安全相关的问题,或者因为在给定当前道路或交通条件无法执行一个或更多个所指示的动作。

[0003] 以此方式,常规车道规划可以具体地基于确定性方法。在操作上,生成确定性图形,并且使用搜索算法(例如,迪杰斯特拉(Dijkstra)算法,A\*搜索算法)来寻找位置之间的最短路径,同时将动作建模为全确定性的。然而,作为示例,车辆可能尝试变道或至少希望进行变道,但是变道可能被竞争的车辆阻挡,迫使车辆停留在当前车道中。在另一示例中,在车辆进行变道的最后机会,车辆可能被阻挡,使得车辆等待而不是在车道中前进。因此,用于执行动作的输入、输出和控制的当前组合可能是限制性的——并且传统车道规划器不提供复杂的输出在驾驶系统中以不完全确定的方式执行驾驶规划。因此,具有用于执行车道规划操作的替代基础的更全面的驾驶系统可以改进用于驾驶系统的计算操作和接口。

### 发明内容

[0004] 本公开的实施例涉及操作车道规划器以基于状态和概率动作空间生成车道规划器输出数据。车道规划器输出数据对应于驾驶系统的车道检测和/或引导数据,所述驾驶系统基于与车道规划器和其他规划和控制组件相关联的分级驾驶规划框架来操作。车道规划器处理车道规划器输入数据(例如,大车道图、源、目标),以生成车道规划器输出数据(例如,与源和目标之间的大车道图的节点之间的边相对应的预期时间奖励)。驾驶系统还可以包括路线规划器(例如,第一规划层),该路线规划器进行操作以将更高级或粗略的路线信息作为车道规划器输入数据提供至车道规划器——例如以提供源位置、目标位置以及映射源与目标之间的路线的初始高级车道图。车道规划器可作为(例如但不限于)第二规划层进行操作,该第二规划层处理车道规划器输入数据以生成被表示为更细粒度或“大(large)”车道图的状态和概率动作空间,其包括与从源节点导航至一个或更多个目标节点相关联的

时间成本。车道规划器将车道规划器输出数据传送至行为规划器,以引起识别要由车辆执行的至少一个动作,以通过大车道图从源位置(例如,本机的位置)到目标节点从节点到节点地横穿。在实施例中,除了实时感知之外,行为规划器还可使用由车道规划器生成的大车道图(例如,包括具有不同时间奖励的多个可选动作)来作出关于要遵循大车道图的哪些边,以及因此在环境中采取哪条路径的决定。在一些实施例中,行为规划器可将实时感知信息馈送至车道规划器,使得车道规划器可动态更新大车道图,并且行为规划器可仅对填充有实时感知信息的大车道图进行操作。

[0005] 与常规系统(如以上所描述的那些)相比,来自车道规划器的输出可以包括针对到达节点的预期等效时间奖励,其中该预期等效时间奖励(可替代地,在此被称为“预期时间奖励”)是例如转换为时间的最佳或最好当前计算的预期奖励。车道规划器输出数据还可包括具有概率分布的动作。例如,在大车道图边的一边中编码的每个动作可具有正的失败概率。在操作上,可以随机地识别边的目标节点,并且可以通过最初使用诸如(并且不限于)Dijkstra算法的搜索算法来计算预期的等效时间奖励,以识别从源节点到任何大车道图节点的时间成本。时间成本可以用于值迭代(或值迭代的变体)以计算预期的等效时间奖励。因此,不同于常规系统,本技术方案中的预期的等效时间奖励是基于时间奖励的最佳预期值并且进一步考虑了未来的不确定性。

[0006] 更确切地,可以基于强化学习生成车道规划器输出数据。强化学习可以使用与状态集合和动作集合相关联的马尔可夫决策过程(“MDP”)来实现。每个动作将现有状态带到随机输出状态,这触发奖励。作为强化学习的一部分,大车道图可被转化成MDP,其中大车道图节点对应于状态并且边对应于具有随机输出节点的动作。负边成本对应于奖励,并且预期时间奖励可被计算为最佳值。

[0007] 在一些实施例中,修改值迭代方法用于控制迭代次数。修改值迭代方法在计算上打破大车道图的循环,因此图形变成非循环的一并且可创建节点的顺序,使得可向源节点有效地传播预期时间奖励。因此,修改值迭代方法的一次迭代——经由这个顺序——可以返回足够好的结果,这与需要多次迭代才能生成最终结果的常规系统相反,其在计算和时间方面都是密集的。

## 附图说明

[0008] 下面参考附图详细描述用于操作车道规划器以基于状态和概率动作空间生成车道规划器输出数据的本系统和方法,其中:

[0009] 图1A是根据本公开的一些实施例的用于生成车道规划器输出数据的示例系统;

[0010] 图1B是根据本公开的一些实施例的用于生成车道规划器输出数据的示例系统;

[0011] 图1C是根据本公开的一些实施例的用于生成车道规划器输出数据的示例数据流图;

[0012] 图2A至图2G是根据本公开的一些实施例的用于生成车道规划器输出数据的车道规划数据的示例视觉表示;

[0013] 图3A至图3D是根据本公开的一些实施例的用于利用修改值迭代算法生成车道规划器输出数据的计算的示例视觉表示;

[0014] 图4A至图4B是根据本公开的一些实施例的用于生成车道规划器输出数据的车道

规划数据的示例视觉表示；

[0015] 图5至图6是示出根据本公开的一些实施例的用于提供车道规划器输出数据的方法的流程图；

[0016] 图7A是根据本公开的一些实施例的示例自主车辆的图示；

[0017] 图7B是根据本公开的一些实施例的图7A的示例自主车辆的相机位置和视野的示例；

[0018] 图7C是根据本公开的一些实施例的图7A的示例自主车辆的示例系统架构的框图；

[0019] 图7D是根据本公开的一些实施例的基于云的服务器和图7A的示例自主车辆之间的通信的系统示意图；

[0020] 图8是适用于实现本公开的一些实施例的示例计算设备的框图；以及

[0021] 图9是适用于实现本公开的一些实施例的示例数据中心的框图。

### 具体实施方式

[0022] 公开了与操作车道规划器以基于状态和概率动作空间生成车道规划器输出数据相关的系统和方法。虽然可相对于示例性自主车辆700(或者,在本文中称为“车辆700”或“本机700”,相对于图7A至图7D描述了其示例)来描述本公开,但是这并非旨在限制。例如,本文中描述的系统和方法可由但不限于非自主车辆、半自主车辆(例如,在一个或多个自适应驾驶员辅助系统(ADAS)中)、驾驶和未驾驶机器人或机器人平台、仓库车辆、越野车辆、耦合到一个或多个拖车的车辆、飞行船、船、穿梭机、紧急响应车辆、摩托车、电动或机动自行车、飞机、施工车辆、水下飞行器、无人机和/或其他车辆类型使用。此外,尽管可以关于用于自主车辆的车道规划来描述本公开,但是这不旨在是限制性的,并且本文所描述的系统和方法可以用于增强现实、虚拟现实、混合现实、机器人、安全和监视、自主或半自主机器应用和/或可以使用车道或路径规划的任何其他技术空间。

[0023] 在此描述的系统和方法可以被以下各项使用,但不限于:非自主车辆、半自主车辆(例如,在一个或多个自适应驾驶员辅助系统(ADAS)中)、驾驶的和未驾驶的机器人或机器人平台、仓库车辆、越野车辆、耦合到一个或多个拖车上的车辆、飞行船只、船、穿梭车、紧急响应车辆、摩托车、电动或机动自行车、飞机、建筑车辆、水下飞行器、无人机、和/或其他车辆类型。进一步,本文描述的系统和方法可用于各种目的,作为示例而非限制,用于机器控制、机器运动、机器驾驶、合成数据生成、模型训练、感知、增强现实、虚拟现实、混合现实、机器人、安全和监视、自主或半自主机器应用、深度学习、环境模拟、数据中心处理、会话AI、光传输模拟(例如,光线跟踪、路径跟踪等)、用于3D资产的协作内容创建、云计算和/或任何其他合适的应用。

[0024] 所公开的实施例可以包括在多种不同系统中,如汽车系统(例如,用于自主或半自主机器的控制系统、用于自主或半自主机器的感知系统),使用机器人实现的系统、空中系统、内侧系统、划船系统、智能区域监视系统、用于执行深度学习操作的系统、用于执行模拟操作的系统、使用边缘设备实现的系统,合并一个或多个虚拟机(VM)的系统、用于执行合成数据生成操作的系统、至少部分地在数据中心的实现的系统、用于执行会话AI操作的系统,用于执行光传输模拟的系统、用于执行3D资产的协作内容创建的系统、至少部分地使用云计算资源实现的系统、和/或其他类型的系统。

[0025] 本公开的实施例涉及操作车道规划器以基于状态和概率动作空间生成车道规划器输出数据。在操作中,接收表示与车辆的行驶路线相关联的大车道图的至少一部分的数据。大车道图包括指示大车道图的一个或多个车道内的潜在位置的多个节点和连接多个节点的多个边,多个边中的至少一个边对应于车辆在一个或多个车道(例如,局部车道)内的相应车辆动作。对于多个边中的至少一个边,至少部分地基于在起始节点与连接节点之间穿过的预期等效时间来计算成本函数。至少部分地基于一个或多个车道、候选车辆动作以及用于每个候选车辆动作的一个或多个行驶路线。至少部分地基于与每个边相关联的成本函数,为至少一个节点计算预期等效时间奖励,从而作出行驶路线的确定。沿着行驶路线控制车辆。

[0026] 驾驶系统中的分级驾驶规划的概述

[0027] 本公开的实施例涉及一种驾驶系统,该驾驶系统基于优化类别、转换机制、以及状态和概率动作空间来操作。驾驶系统还基于分层规划框架来操作,该分层规划框架包括用于分层规划的状态和动作空间。在实施例中,分级规划可以分解成路线规划、车道规划和行为规划。在实施例中,可分别使用路线规划器、车道规划器和行为规划器来执行路线规划、车道规划和行为规划,并且路线规划器、车道规划器和行为规划器可相互进行通信以确定针对本机的控制决策。

[0028] 如在此更详细地讨论的,实施等效时间或接近等效的时间单位以实现驾驶规划层之间的连接,例如,第一规划层(例如,路线规划)与第二规划层(例如,车道规划)之间的连接。基于等效时间的驾驶规划在对于存在多个候选路线的驾驶任务的驾驶规划中尤其有益。用于在层之间形成连接的实现细节可以变化。在高层次,沿着具有相似预期奖励的候选路线的全球导航卫星系统(“GNSS”)坐标可用作下一规划层的引导目标。以这种方式,第一规划器(例如,路线规划器)支持沿着朝向目标的潜在路线不断地提供引导目标点(例如,几英里外)。引导目标点允许自动驾驶管线中的下一规划器(例如,车道规划器)以目标点作为目标进行操作,因此抽象出更长期的目标并且在更小的规模上提供更简单的规划问题,其可被更详细地评估。

[0029] 参照驾驶系统,该驾驶系统可以被配置成针对一组优化类别进行优化。该组优化类别可以包括目标奖励、花费的时间、花费的资源、不适、舒适、障碍物安全、路径服从、等待条件服从、和/或其他类别。该组优化类别可以被定义为奖励或成本,并且可以被配置为导航至目标时的权衡。为了将优化类别实现为权衡,可以定义优化类别之间的转换机制,使得奖励或成本量化每个优化类别。在一些实施例中,一个或多个优化类别(例如,安全)可以用硬约束而不是成本来定义。

[0030] 在至少一个实施例中,优化类别之间的转换机制可以是时间。例如,目标奖励(如以时间单位定义)可以相对于也以时间单位定义的其他优化类别进行计算。具体地,存在一些可以花费的时间分配或预算(量)(例如,一旦所有成本被考虑进去的等效时间),其将确定是否可以进行至目标的行驶任务。其他优化类别可以从目标奖励中减去等效时间。例如,路径服从(例如,以车道居中规范或环境的标称路径)或等待条件(例如,交通信号灯或通行权)将从目标奖励的等效时间中减去。

[0031] 状态和动作空间可以支持将时间实现为转换机制。在状态和动作空间中,可以实施基于时间的预期奖励(或预期时间奖励)来对具有等效时间的优化类别的构造进行建模。

该模型可以基于在强化学习中通常使用的马尔可夫决策过程(“MDP”)框架。例如,世界的状态由状态和动作空间建模,并且在每个状态中,可以从一组动作中选择特定动作,并且每个动作可以导致下一状态的概率分布和状态转换的奖励。可使用“大”车道图来表示状态和动作空间,该“大”车道图可包括沿着源节点(例如,本机的当前位置)与目标节点(例如,与本机的一个或更多个目的地对应的节点)之间的车道图的不同车道或路径延伸的多个中间节点,其中节点经由表示动作的边连接。

[0032] 状态和动作空间包括概率建模,从而使得单个下一状态不衔接至每个动作。例如,可以采取一个动作来发起变道,但是该动作可能会或可能不会成功,从而导致进入下一车道中或者停留在车道中。此外,状态和动作空间可有利地通过概率或预期(诸如穿行特定路段的预期时间)来建模一些方面。在这方面,奖励根据预期奖励来管理,从而操作目标变成了通过选择动作来最大化预期奖励(例如,奖励减去成本——全部以等效时间表示)。状态和概率动作空间在操作上定义了支持估计从状态开始并且继续最佳策略(例如,最佳动作序列)的预期时间奖励的Q学习或值迭代。

[0033] 驾驶系统进一步基于分层规划框架来操作。基于在分级规划框架中定义的分级分解,可以降低驾驶规划的复杂度。分层框架可基于经建模的状态和动作空间来操作,该经建模的状态和动作空间可在跨越长期模型和短期模型之间的若干层方面被建模。长期模型考虑远离当前状态的大规模状态和动作——这支持更快且更简单的建模。短期模型考虑更接近当前状态的小状态和动作——这支持精确建模,并且还支持经由依靠更简单和可管理的动作的分治方法来实现更长期目标的能力。

[0034] 状态和动作表示支持分层规划,因为可以使用粗略的状态和动作空间来计算预期奖励。例如,可以从粗略状态识别有利状态的子集,使得下一层可以使用粗略状态的终端节点。粗略状态然后可以链接到更详细的状态和动作表示,但是尺度更小。可以定义切换,其中最大的规划层跨越整个规划问题,其中每个下一层有效地计算更精细的规划问题。

[0035] 驾驶规划可分解为路线规划、车道规划和行为规划。路线规划可以表示最大且最粗的规划尺度(例如,大陆尺度的道路级别粗度)。车道规划——中间规划级别——可以表示车道选择、变道、以及在个体车道细节(例如,以几公里的尺度)的选择岔路和转弯选择。行为规划可表示包含车道保持、变道、选择岔路和通过将它们转换成运动规划的转弯的实施,且最后控制,其将运动规划转换成致动动作。

[0036] 参考图1A至图1C,图1A至图1C包括根据本公开的一些实施例的包括车道规划器的示例系统100。例如,系统100可以操作车道规划器以基于状态和概率动作空间生成车道规划器输出数据。应当理解,本文所述的这种和其他布置仅作为示例进行阐述。除了或代替所示的布置和元件,可使用其他布置和元件(例如,机器、接口、功能、次序、功能分组等),并且可完全省略一些元件。进一步,本文描述的许多元件是可以被实现为分立或分布式组件或与其他组件结合、且以任何合适的组合和位置的功能实体。本文描述为由实体执行的各个功能可由硬件、固件和/或软件执行。例如,各个功能可以通过处理器执行存储在存储器中的指令来实现。在一些实施例中,系统100可以包括与图7A-7D的示例自主车辆700、图8的示例计算设备800、和/或图9的示例数据中心900类似的组件、特征、和/或功能。

[0037] 系统100提供用于基于状态和概率动作空间提供车道规划器输出数据的组件和操作。车道规划器输出数据对应于驾驶系统的车道引导数据,该驾驶系统基于与车道规划器

和其他规划和控制组件相关联的分级驾驶规划框架来操作。如图1A中所示(并且在下面更详细地讨论),组件包括以下各项:车道规划器110(值迭代)、源节点和多个目标节点112、大车道图130中的节点的预期等效时间奖励114、包括运动规划器142的行为规划器140以及与局部车道位置144链接的大车道图节点。图1A中的组件和操作对应于本文所述的图1B和图1C中的组件和操作。

[0038] 路线规划器

[0039] 转向路线规划器120(图1B),在一个或多个实施例中,路线规划器120可以被实现为最粗和最大的规划层。路线规划器120负责以例如大陆的尺度进行规划,或者与定义的大地图区域一样大。路线规划器120可操作以提供允许路线规划器120被灵活替换的规范输出。路线规划器120可以以类似于现代导航路线的方式操作。在一些实施例中,路线规划器120以足够的频率执行操作以能够适应当前位置、目标变化或其他条件。

[0040] 在高层,路线规划器120接收道路网络122作为路线规划器输入数据。道路网络识别源和目标(被包括在任务124中)。可以确定从源到目标的几(k)个不同的短路径,其中每条短路径具有从源到目标的预期的等效时间114。可将预期等效时间114作为路线规划器输出数据传送到行为规划器140。

[0041] 路线规划器120提供用于传送路线规划器输出数据的接口。路线规划器输出数据可以是定义数量(例如,k)的全球定位系统(“GPS”)或GNSS轨迹,每个坐标具有对应的预期等效时间。在一个示例中,GPS轨迹可以是世界测地系统1984(“WGS84”)坐标系中的位置列表。每个GPS位置具有附加的预期等效时间,该预期等效时间是如果在当前GPS位置开始对驾驶任务的剩余部分的预期奖励。以这种方式,路线规划器120使用第一地图(例如,可用的最佳地图,诸如HD地图、GPS地图等)构建道路网络的有向图近似(“图近似”或“有向图数据结构”)。图近似可以包括与道路网络相关联的节点和边。图中的节点可以对应于道路网络的简单版本中的GPS坐标。边可以将节点连接在一起。每个边可以用预期的等效时间(例如,穿过与相应边相关联的节点的时间量)来注释。因此,可将边上的预期等效时间设定为等于穿过该边所花费的平均预期成本。图近似可以包括辅助信息,诸如交通状况、转弯困难等。辅助信息可包括来自车队的关于交通状况的信息,并且路线规划器120可在车辆中的驾驶系统中或在云计算位置中执行。图近似可以考虑一个或多个优化类别。

[0042] 在操作上,举例来讲,用户可以经由GPS坐标或地址指定目标。目标可被转换成道路网络122的节点之一,并且该节点被作为目标节点来处理。以等效时间表示的目标奖励被分配给目标节点。原则上,奖励可被分配给多个目标,或者表示中止任务而没有奖励的识别节点;然而,在简化的实施方式中,奖励可基于用于路线规划器120的单个目标节点。在状态和动作空间中捕获由路线规划器120执行的操作。具体地,每个节点对应于一个状态,并且来自节点的每个出边表示从该节点行进至下个节点的动作。

[0043] 动作导致确定性结果,并且边携带正成本(例如,所花费的预期等效时间),并且存在单个目标和目标奖励。驾驶规划(例如,一个完整的规划问题)包括计算从任何节点开始的预期奖励的操作。该计算对应于找到从每个节点到目标的最短路径(例如,将所花费的预期等效时间视为每个边的距离)。然后,在每个节点处的预期奖励是目标奖励减去从该节点到目标的最短路径的成本。如果预期奖励是负的,则最短路径比目标奖励更昂贵。在这种情况下,预期的目标奖励可以被截断为零,对应于如果驾驶规划到达该节点则中止任务。

[0044] 经由路线规划器120的驾驶规划可以实现若干效率。例如,道路网络122中的每个节点的预期奖励对于计算或通过路线规划器接口传输是不必要的和昂贵的。相反,路线规划器120基于特定源节点(道路网络122上的当前位置)来操作。路线规划器120的用户可诸如通过将当前GPS位置对齐到最近的节点或通过沿着道路网络122跟踪来指定当前的源节点。路线规划器120有效地操作,因为路线规划器120确定从源节点到目标节点(例如,在任务124中被识别)的最短路径并计算沿该最短路径的每个节点的预期奖励114。存在用于发现通过道路网络122的最短路径的另外的有效解决方案。例如,在给定源和目标的情况下,收缩层级数据结构支持以毫秒速度在移动计算平台上以大陆尺度进行路由。在操作上,在道路网络122的预处理步骤中构造紧凑的数据结构(例如,收缩层级结构)。使用收缩层级,实况路由处理变成双向Dijkstra的搜索,限制是将搜索的每一方向限制为在收缩层级中向上。以此方式,在下一较大道路上(例如,在高速公路上)搜索源和目标的最短路径,且搜索操作收敛于高层级。在一个或更多个实施例中,层级是连续的,每个节点占据其自身的级别。

[0045] 路线规划器120可以被配置成将不同类型的实现方式用于路线规划。可使用可以为优选路线产生GPS轨迹的任何路线规划器120。例如,外部路线规划器可传送GPS轨迹或单个GPS轨迹而没有预期奖励。与GPS轨迹相关联的行进距离的曲线积分可用作等效时间的替代。如果给出了沿该路线的预期时间,则可以用于为每个节点计算剩余的预期时间,并且可以从目标奖励中减去它以创建等效时间。

[0046] 参照图1C中的路线规划器120,在高层,路线规划器120在C122接收道路网络122和任务(源、中间路点和目标)124作为路线规划器输入数据。任务124识别关于道路网络122的源和目标。在C124,可以确定从源到目标的几(k)个不同的最短路径。每条最短路径在从源至目标的每个节点处具有预期的等效时间奖励124。在C126,具有预期等效时间奖励的最短路径可作为路线规划器输出数据传送出(例如,具有预期等效时间奖励的GPS轨迹126)。

[0047] 路线规划器120将车道规划器输入数据传送至车道规划器110。车道规划器输入数据122可以包括具有多个带注释的节点和多个带注释的边的大车道图130的至少一部分。具体地,路线规划器120提供车道规划器输入数据,车道规划器输入数据在车道规划器110处被处理以生成车道规划器输出数据,车道规划器输出数据传送给行为规划器140以引起对车辆要执行的至少一个动作的识别。

[0048] 参考图2A,图2A图示了基于路线规划的示意性表示包括路线点(例如,路线点202A、路线点204A和路线点204C)。路线点可具体地对应于大车道图裁剪(例如,大车道图130的子图)。如所讨论的,大车道图130的子图可包括由路线规划器120提供的目标节点的预期时间奖励114(例如,经由车道规划器支持的源节点到加油站中间目标节点,以及由路线规划器120支持的加油站到家—最终目标节点)。以这种方式,路线规划器120使用第一地图(例如,可用的最佳地图)构建道路网络122的有向图近似(“图近似”或“有向图数据结构”)。图近似可以包括与道路网络122相关联的节点和边(例如,多个带注释的节点和多个带注释的边)。图中的节点对应于道路网络122的简单版本中的GPS坐标。边连接节点,并且每个边可以用预期的等效时间114来注释。例如,边上的预期等效时间114可以是平均预期花费在穿过该边的成本。有向图可以包括辅助信息,诸如交通状况、转弯困难等。辅助信息可包括来自车队的关于交通状况的信息,并且路线规划器120可在车辆中的驾驶系统中或

在云计算位置中执行。图近似可以考虑一个或更多个优化类别。

[0049] 车道规划器

[0050] 参考图1B,图1B示出了示例系统100,其进行操作以提供车道规划器110,车道规划器110基于车道规划器输入数据(例如,图1A的源节点和多个目标节点112或从路线规划器(例如,路线规划器120)接收的具有预期等效时间奖励126的GPS轨迹)中的大车道图130的状态和概率动作空间来生成车道规划器输出数据(例如,图1A的预期等效时间奖励114)。车道规划器输出数据对应于驾驶系统的车道引导数据,该驾驶系统基于与车道规划器和其他规划和控制组件相关联的分层驾驶规划框架来操作。车道规划器110提供车道规划器输出数据114,以使行为规划器(例如,行为规划器140)确定将由车辆在行驶路线上执行的动作(例如,经由图1A的运动规划器142)。

[0051] 车道规划器输入数据112可以包括源、目标和大车道图130。可从路线规划器接收至少一些输入。例如,路线规划器传送与车辆的驾驶规划相关联的路线点(例如,表示为全球定位系统(“GPS”)坐标的GPS轨迹或路途点或位置)。路线点用于识别车道规划器利用的大车道图130的子部分。路线规划器120还为包括驾驶规划的最终目标节点的路线点提供预期的等效时间奖励126。例如,路线点可以表示驾驶规划的中间目标(像加油站)或最终目标(如家)。此外,当车道规划器的几何范围受到限制时,路线点还可用于设置引导目标。车道规划器消耗引导目标的预期时间奖励,以估计在其范围内的大车道图节点的预期时间奖励。例如,如果在车道规划器的范围的边界内仅存在一个具有1000秒的预期时间奖励的引导目标,并且如果从大车道图节点到引导目标的最佳时间成本是300秒,那么 $1000-300=700$ 秒是该大车道图节点的预期时间奖励。路线点可以进一步用于对边等效时间成本或节点预期等效时间奖励施加惩罚,以便车道规划器110更喜欢紧跟路途点的动作。

[0052] 因此,车道规划器110在路线规划器120下的下一细节级别进行连接。车道规划器110支持明确地操作沿着道路的每个车道。与道路相比,状态连接到特定车道中和沿着特定车道的位置。车道规划器110基于表示车道感知状态和概率动作空间的数据结构在下一细节级别操作。

[0053] 车道规划器110还使用数据结构和数据结构中表示的数据来执行操作。在一个实施例中,数据结构是大车道图130。大车道图130可不同于局部车道图,局部车道图与和感知映射相关的更小范围有关(例如,经由行为规划器140)。大车道图130可根据多个节点来表示,其中节点可以是在某纵向行进的车道中心的顶点。大车道图130表示状态和动作空间,因为节点表示沿着车道和在车道中的车辆700的潜在位置。

[0054] 节点还可以具有附接到其上的附加信息(例如,附加注释数据),特别是在至少一个GNSS坐标。当节点表示直至下一节点的车道的更密集采样(例如,车道数据的更密集采样)时,附加GNSS坐标可与该节点相关联。动作可以由节点之间的边表示,并且动作可以对应于:在车道中向前移动到下一节点;车道向左或向右改变到下一车道;或者选择岔路或转弯的各种选择——全部由边的目标节点表示。边还可具有包括该边表示的操控类型的信息,并且每个边可与表示在穿行该边时花费的预期平均成本的预期等效时间114相关联。可将等效时间114定义为平均值,但在替代方案中,可进一步由当前交通状况通知。在一些实施例中,也可以通过操控类型来调节边。

[0055] 用于形成大车道图130的主要机制包括映射和/或感知操作。可以用边上的值(例

如,由平均值通知的值、由一天中的时间适配的平均值、由从车辆车队学习的当前状况通知的平均值)生成大车道图130,并且可以使用由多个因素(例如,速度、车道占用和交通灯的状态)的实况感知通知的平均值来进一步增强车辆700的当前位置附近的边。

[0056] 大车道图130可从路线规划器120移交给车道规划器110。例如,路线规划器120生成大车道图130的第一表示并且将大车道图130传送至车道规划器110。路线规划器120可使用当前路线规划作为引导。举例而言,路线规划可用k个GPS轨迹来表示,以使得从大车道图130中提取大车道图130的子图。路线规划器120随后执行映射操作以生成大车道图130的第一表示。大车道图130包括可在大车道图形130的子图中使用或表示的大车道图数据(例如,状态和动作,或等效的节点和具有等效时间的边)。

[0057] 大车道图130的子图可以是围绕经由路线规划提供的k个GPS轨迹中的每个轨迹的具有某一固定宽度(例如,300m)的狭长带(corridor)内的一部分,从车辆的当前位置开始到与当前位置相比所花费的某一预期等效时间。对于每个GPS轨迹,路线规划器120被配置为处理相同的预期等效时间114。以这种方式处理每个GPS轨迹允许车道规划和路线规划之间的转换发生在沿着每个GPS轨迹的相似花费的等效时间。

[0058] 总时间可以通过将被考虑用于车道规划的子图中的固定数量的节点作为目标来限制。如果沿GPS轨迹的范围到达目标,则狭长带基于到目标的当前位置。如果沿GPS轨迹的范围未到达目标,则狭长带可以延伸到沿轨迹考虑的最远GPS点。接下来,可以识别大车道图130中定义的半径(例如,100m)内的所有节点。这些节点被分配最靠近它们的GPS点的预期等效时间奖励,加上到它们最近的GPS点的距离的惩罚。沿着每条轨迹靠近更远的GPS点的节点被视为车道规划的目标节点。这些目标节点可以被称为代理目标或引导目标。以这种方式,这些操作在计算上将大车道图130与路线规划拼接在一起,使得总规划的质量可被计算为目标节点之一处的预期时间奖励114减去车道规划中为了到达目标节点而花费的预期等效时间114。

[0059] 如所讨论的,搜索半径限制(例如,100m)可以由车道规划器110实施。搜索半径限制的原因涉及当执行路线规划和车道规划的交接时地图中的限制差异。并且在GPS轨迹周围使用狭长带的原因是允许车道规划比在路由中使用的地图所允许的做出更详细的规划,同时使用车道规划和路线规划之间的偏差不应该需要超过狭长带宽度的一半的启发法,这样比在所有方向上搜索效率更高。

[0060] 在操作上,可以针对沿着每个GPS轨迹的定义距离(例如,几英里)执行车道规划。定义的距离可以被配置成使得定义的距离足够长,从而不需要超出该距离的详细的车道规划。为了更精确地支持穿行现有路线规划,操作可以可选地通过基于节点或边与GPS轨迹的距离向节点或边添加惩罚来惩罚与GPS轨迹的偏离。然而,惩罚可能不是从初始的一组奖励类别中得到的。

[0061] 可生成大车道图130(例如,大车道图130的子图),使得计算过程的目标是确定大车道图130中每个节点的预期等效时间114。即,如果在该节点处开始并且遵循最佳动作序列,则预期作为奖励的等效时间。大车道图上的动作可能无法被建模为完全确定性的。具体地,车辆700可尝试变道或至少希望进行变道,但是该变道可能被竞争车辆阻挡,迫使车辆留在其车道中。在最后一次进行变道的机会,车辆700可能被阻挡,使车辆700等待并且不在车道中向前移动,而是最后回到同一节点处。

[0062] 因此,与常规的大车道图不同,一些边被升级到尝试但可能不成功的动作。并且,代替直接和确定性地引向某个节点,动作和边还可以基于一些概率导致到达其他节点。因此,车道规划不再简单地基于确定图中的最短路径(例如,利用确定性图,使用例如但不限于Dijkstra算法来从任何节点找到最短路径和预期的等效时间)。可适合的替代算法包括但不限于A\*、最短路径、以及广度优先算法以及它们的变体。相反,在状态和概率动作空间中,可以执行值迭代方法以从任何节点找到最佳动作和预期等效时间。

[0063] 可基于在图上最初执行搜索算法(例如,Dijkstra的搜索算法)来高效地执行寻找最佳动作和预期等效物。可以假设所有动作都将成功并且导致在每个节点处生成预期等效时间值的近似值。近似的预期等效时间可以用作用于对概率图执行值迭代的起始点。在另一个实施例中,可实施额外技术以使值迭代更快。在操作是,将大车道图130的较小范围限制到接近车辆的当前位置的范围。这有效地在规划层次中创建来一个或更多个级别,将部分留在常规的图搜索中更远的地方,在值迭代期间锁定那些节点的值。

[0064] 大车道图130可以对应于车道规划器输入数据的一部分,并且可以包括经由路线规划器120填充的信息(如在此所描述的)并且可以可选地通过实时感知信息来增强(例如,在存在来自行为规划器140的反馈以更新大车道图130的情况下)。大车道图130在车道规划器110处被处理以生成包括在大车道图的多个节点处的等效时间114的车道规划器输出数据。使用等效时间114,可以将动作的值计算为对该动作进入下一节点的概率结果的预期值。车道规划器输出数据可进一步包括与指示来自每个节点的最佳动作的动作相关联的概率。进一步设想,可以通过假设动作将成功来生成单个标称规划——例如作为可视化或用于提示运动规划器。

[0065] 车道规划器输出数据(例如,车道规划)可以用于针对变道做出确定。例如,让我们假设特定车道是优选车道(例如,剥离其他车道)。在这样的示例中,给定另外等效的并行车道,车道规划输出数据可以指示尽快改变到那个车道。实际上,车辆离剥离点越近,变道不及时成功的风险就越高。因此,车辆离剥离点越近,压力就越高,并且这反映在节点的预期的等效时间中。另一方面,该目标车道可能在某种程度上不利,例如反映在更高的岔路或合并出该车道的风险——在那些边的等效时间内受到碰撞风险的惩罚——或者由于该车道移动较慢而反映在更高的等效时间。在这种情况下,车道规划可以指示延迟变道或在变到目标车道之前绕过较慢移动的交通。此外,现在或将来,更高的车辆密度和速度可能会降低变道成功的可能性。这可以通过通知变道边的编辑的实时感知来反映。在实施例中,更高的车辆速度或不存在车辆可使一个车道(或其节点)优于另一车道,这可通过通知车道保持边的编辑的实时感知来反映。在一些示例中,最近的一组边可由行为规划器基于其直接可行性来编辑。

[0066] 因此,车道规划器110接收车道规划器输入数据112并且生成车道规划器输出数据(例如,图1A的车道规划器输出数据114)。如本文所讨论的,车道规划器输出数据114包括预期等效时间奖励(例如,图1A中的预期等效时间奖励114)。状态和概率动作空间促进执行值迭代(例如,经由图1A的值迭代引擎110)以找到任何大车道图节点的最佳动作和预期等效时间奖励。车道规划器110生成车道引导数据:每个大车道图节点的预期等效时间奖励。计算预期等效时间奖励可包括确定具有指示执行动作的可能性的概率分数的动作。因此,代替对直接和确定性地引导至某个节点的动作建模,这些动作还可以基于其他概率允许车辆

到达不同的节点。

[0067] 因此,在操作上且相对于图1C,在C112,接收表示与车辆的行驶路线相关联的车道图130的至少一部分的数据。大车道图130包括指示大车道图130的一个或多个车道(例如,局部车道)内的潜在位置的多个节点和连接多个节点的多个边,多个边中的每个边对应于一个或多个车道内的车辆的相应车辆动作。对于多个边中的每个边,至少部分地基于在起始节点和连接节点之间穿行的预期等效时间114来计算成本函数。至少部分地基于一个或多个车道,可生成候选车辆动作以及用于每个候选车辆动作的一个或多个行驶路线。至少部分地基于与每个边相关联的成本函数,为每个节点计算预期的等效时间奖励,从而使得做出行驶路线的确定。沿着行驶路线控制车辆。在C114,车道规划器110提供车道规划器输出数据,以引起确定(例如,经由行为规划器140)车辆将在行驶路线上执行的识别动作。

[0068] 参照图2B至图2F,图2B至图2F示出了大车道图130的特征。大车道图130包括行驶车道(例如,图2B的车道202B、车道204B以及车道206B)的位置。大车道图130可以是具有作为行驶车道的位置的节点(例如,图2C的节点202C、节点204C和节点206C)的有向图。这些位置可以从与驾驶规划相关联的地图中的车道表示进行采样。有向图可进一步包括连接两个节点并且描述动作的边(例如,边202D和动作208D、边204D和动作210D、以及边206D和动作212D)。动作可以包括以下中的任一个:笔直的、左车道转弯、右车道变换等,(例如,分别为动作208D、动作210D和动作212D)。

[0069] 车道规划器输入数据可以包括与大车道图130(例如,有向图)相关联的状态和概率动作空间,其中节点和边对应于行驶路线的地图的至少子部分的状态和动作。状态和概率空间还支持表示节点的预期等效时间奖励114。预期的等效时间奖励114可以指完成行驶路线的量化的时间奖励或成本。

[0070] 车道规划器110作为具有分级规划框架的驾驶系统中的规划组件来操作。车道规划器110提供用于生成车道规划器输出数据的操作,车道规划器输出数据可以包括在车道规划器110处生成的数据以支持确定将由车辆在行驶路线上执行的动作。车道规划器输出数据可被称为传送至下游组件(例如,行为规划器140)的车道引导或车道引导数据。预期具有等效奖励时间114的大车道图节点可与局部车道图(例如,从用于构建大车道图的同一地图或从诸如实况路径感知的其他源生成)链接以支持运动规划。车道引导数据可识别车辆应在其上行驶以完成行驶规划(例如,朝向目标的行驶路线或行驶任务)的车道。

[0071] 参考图2E,在操作上,车道规划器输出数据可以基于强化学习生成。可以使用与状态集合和动作集合(例如,图2E中的状态和概率动作)相关联的马尔可夫决策过程(“MDP”)来实现强化学习。每个动作将现有状态带到随机输出状态,这触发等效的预期时间成本。作为强化学习的一部分,大车道图130可变换成MDP,其中大车道图节点表示状态,而大车道图边表示具有随机输出节点的动作。负边成本被表示为奖励。预期时间奖励114可指最佳预期值,并且每个大车道图节点可被分配预期等效时间奖励(例如,图1A的预期等效时间奖励114)。节点的预期等效时间奖励以时间的形式衡量从该节点开始到达目标的好坏程度。预期等效时间奖励114可以指对于行驶路线,驾驶员愿意花费的时间减去预期等效时间成本。

[0072] 参考图2F,可创建指示左车道变换动作的大车道图130的边。边可以开始于定位在位置202F的大车道图节点,并且结束于两个可能的大车道图节点,一个定位在位置204F并

且另一个定位在位置206F。如果边在位置204F处结束,则左车道变换成功。如果变道失败,则位置202F处的车辆可以不经车道分隔线,可以返回至其原始车道并且可以在位置206F处结束。预期等效时间成本被分配给每个大车道图边,并且预期等效时间成本可以基于地图中的车道的长度和地图速度(例如,车辆正在移动的检测速度,标记在地图中)。可以基于动作类型来进一步调整每个边的成本。例如,变道可能产生额外成本。

[0073] 在车道规划器110中使用的MDP(图2E)还可以包括具有概率分布的动作。例如,在大车道图边中编码的每个动作可具有失败概率(例如,图2F的位置204F的变道成功概率PROB 0.9和位置206F的变道失败概率PROB.0.1)。边的目标节点可以随机标识(例如,位置204F或位置206F)。与常规系统不同,该技术方案中的预期等效时间奖励114是基于时间奖励的最佳预期值并且进一步考虑未来不确定性。

[0074] 参考图2G,在操作上,可以通过最初使用搜索算法(例如,Dijkstra算法)识别从源节点到任何大车道图节点的时间成本来更高效地计算预期等效时间奖励114。时间成本可以用于值迭代或值迭代的变体(例如,经由图1A的值迭代引擎118)以计算预期的等效时间奖励114。可以在热图渲染中表示预期的等效时间奖励。热图渲染是用于描绘预期等效时间奖励数据的强度(例如,高、中、低)的可视化。例如,热图(例如,图2G中的具有源节点202G、目标节点204G、第一颜色维度206G、第二颜色维度208G和第三颜色维度210G的热图)可用将出现在地图上的彩色叠加层来启用,并且具有较高预期等效时间奖励的区域将以第一颜色着色,并且具有较低预期等效时间奖励的区域将以第二颜色出现。第三颜色(例如,第一颜色和第二颜色的混合)可以描绘中等的预期等效时间奖励。以此方式,热图可向观看者提供关于预期等效时间奖励如何在空间上群集或变化的明显视觉提示。例如,具有较高预期时间奖励的节点通常靠近目标节点;具有较低预期时间奖励的节点通常靠近源节点;任务失败节点(例如,不具有连接至目标节点的路径的节点)具有零个预期时间奖励。

[0075] 修改值迭代

[0076] 可以基于值迭代来生成被定义为最佳预期值的预期等效时间奖励114;然而,可以替代地使用修改值迭代方法来控制迭代次数。在高层,因为大车道图通常是循环的,使用经典值迭代,迭代次数不受控制(例如,经典值迭代可执行大量迭代直到收敛),所以计算量可能很大。相反,修改值迭代方法打破了大车道图的循环,因此大车道图130变成非循环的,并且可创建节点的顺序,从而可向源节点快速传播预期时间奖励。因此,修改值迭代方法经由此顺序的一次迭代可返回足够好的结果。在车道规划器评估由于来自节点的边已经被消除而不具有正的预期时间奖励的节点的情况下,可以加回边并且可以更新节点奖励。

[0077] 例如,在(由大车道图130表示的)状态和概率动作空间中,可以执行值迭代方法以从任何节点找到最佳动作和预期的等效时间奖励114。以下描述了用于计算大车道图中的节点的预期时间奖励的值迭代的变体。计算预期时间奖励可基于强化学习。在操作上,可以将大车道图130转换成马尔可夫决策过程(“MDP”),其中大车道图130的节点是状态并且大车道图的边是动作。每个动作 $a$ 将一个状态 $n$ 带到随机输出状态 $m$ ,并且转换概率可以由 $p(m|n,a)$ 表示。

[0078] 一般而言,对于任何MDP,每个状态携带衡量处于该状态的可能性的值。在大车道图130的情况下,大车道图节点的预期时间奖励114是节点的值表示。值函数可经由称为值迭代的迭代过程来计算。在每次迭代时,每个节点的值根据该节点所依赖的其他节点的

值进行更新。理论上,在充分迭代之后,节点的值变成常数,其是预期的时间奖励。在大车道图是非循环的情况下,一次迭代就足够了,因为可对给定顺序的节点执行计算以计算值,使得任何额外迭代都不会修改这些值。

[0079] 大车道图通常是循环的,并且如果迭代次数不受控制,则计算量可能非常大。然而,可处理(例如,中断循环)大车道图130,因此大车道图变成非循环的,同时,计算确保值的最佳传播的节点顺序。因此,经由此顺序的一次迭代可以返回足够好的结果。如果一些节点因为来自这些节点的边消失而不具有正的预期时间奖励,则可加回边并且可更新这些节点的奖励。

[0080] 在操作中,假设从大车道图130中识别一个源节点和一个或多个目标节点。在操作1:对于每个大车道图节点,执行重置操作以将节点的预期时间奖励(例如,奖励属性)设置为零,将节点的状态(例如,状态属性)设置为未解决的,并且设置依赖性计数(例如,依赖性计数属性),其为未解决的直接后继者的数量。

[0081] 在操作2:在确定性大车道图(例如,假设其所有动作都成功的大车道图)上使用二进制堆运行搜索算法(例如,Dijkstra算法)操作。使用该堆,可基于从源节点起的成本的降序来计算大车道图节点的顺序。生成该节点顺序(例如,NodeOrder)的数组。

[0082] 在操作3:初始化空队列(“Queue”)、存储节点的第一空数组(例如,SettleOrder)、以及第二空数组(例如,RemovedEdge)。在一个或多个实施例中,第一空数组可以用于存储节点并且第二空数组可以用于存储将被移除的边,并且SettleOrder数组中的任何节点可以是已解决节点。车道规划器过程在NodeOrder数组、Queue、SettleOrder数组以及RemovedEdges数组上工作。

[0083] 在操作4:建立SettleOrder数组并计算其节点的预期时间奖励。例如,对于一个节点被推入Queue的每个计算,该节点变为已解决节点,然后更新该节点的所有直接前驱节点的依赖性计数。确定已满足停止标准是基于执行以下描述的操作4a然后是操作4b。在一个或多个实施例中,可以重复地执行一个或多个操作4a和4b,直到满足停止标准。

[0084] 在操作4a,可以执行以下三个操作中的一个。第一操作包括将目标节点推入Queue中,第二操作包括将具有零依赖性计数的节点推入Queue中,前提是目标节点已解决,并且第三操作包括如果第一操作和第二操作中提及的节点已解决,则执行其他操作。操作4a的第三操作可包括以下的一个或多个:遍历数组NodeOrder以获得终止于NodeOrder数组中的未解决节点处的大车道图边;忽略NodeOrder数组中在该结束节点之前的所有节点;或从大车道图中移除边并更新该边的起始节点的依赖性计数。如果其依赖性计数为零,那么将该节点推到Queue中,且将此移除的边保存在RemovedEdges数组中。

[0085] 在操作4b,可重复执行一个或多个操作直至Queue为空。例如,操作4b的这些操作可包括弹出Queue的前端节点,将该节点推到SettleOrder数组中并计算该节点的预期时间奖励,或查找该节点的直接前驱节点并将具有零依赖性计数的未解决节点推到Queue中。

[0086] 在操作5,对于任何选择的迭代次数,可以将RemovedEdges数组中的所有边添加回大车道图,并且可以基于SettleOrder数组来更新预期时间奖励。

[0087] 参考算法的其他考虑包括:对于操作4中的停止标准,停止标准可以是任何其他合理定义的标准,例如,解决源节点,解决源节点附近的节点,和/或解决每个大车道图节点。操作4可被定义为需要查找大车道图节点的直接前驱节点,并且为了更高效的实现,用于每

个大车道图节点的信息可被预先缓存。

[0088] 可以如下计算预期时间奖励114:对于每个目标节点,奖励是确定的正值(例如,固定的大正值,从由路线规划器计算的引导目标的预期等效时间奖励导出的值);对于不具有任何直接后继者的节点,奖励为零;并且对于其他节点,节点n的奖励Reward(n)可以经由Bellman方程计算,如下:

$$[0089] \quad \text{Reward}(n) = \max_{\{\text{actions } a \text{ from } n\}} \text{Reward}(a)$$

[0090] 其中

$$[0091] \quad \text{Reward}(a) = \sum_{\{\text{output nodes } m\}} [p(m|n, a) [\text{Reward}(m) - \text{ActionCost}(a, n \rightarrow m)]]$$

[0092] 其中 $p(m|n, a)$ 是通过动作a将节点n带到m的转换概率,并且 $\text{ActionCost}(a, n \rightarrow m)$ 是采取将n带到m的动作a的成本。注意,假设的折扣率被设置为一。

[0093] 可以假设每个目标节点的预期时间奖励114足够大,并且当节点具有正奖励时,这意味着存在从该节点到目标节点中的至少一个的路径。否则,不存在这样的路径。

[0094] 对于算法的时间复杂度,N和E可以分别是大车道图节点和边的数量。总之,在大车道图节点和边中复杂度是线性的,不包括Dijkstra算法的复杂度。具体地,操作2采用 $O(\log N)$ 。对于操作4,创建SettleOrder数组采用 $O(N+E)$ ,并且奖励计算采用 $O(N+E)$ 。操作5采用 $O(N+E)$ ,因为迭代次数是固定参数。

[0095] 参考图3A至图3D,提供了修改值迭代方法的示例图示。图中的数字表示以秒为单位的边成本。源节点为A,目标节点为K。边EF和IJ表示可能失败的两个动作。如果动作EF失败,则输出变为J而不是F。如果动作IJ失败,则输出变为F而不是J。EF或IJ成功的概率是0.9。采取失败动作的成本与成功动作的成本相同。除了EF和IJ之外的任何边总是成功的。在本示例中,词语“奖励”表示预期的等效时间奖励。

[0096] 执行修改值迭代方法包括以下操作:操作1:进行初始配置,使得所有节点未被解决且携带零奖励。依赖性计数在图3B中示出。操作2:在确定性大车道图130(其假设动作EF和IJ不会失败)上,基于从A:NodeOrder=[L,K,J,I,H,G,F,D,E,C,B,A]的成本降序来执行搜索算法(例如,Dijkstra算法)以得到NodeOrder数组。操作3:停止标准对应于所有大车道图节点何时被解决。

[0097] 在操作上,在第一操作中,目标节点K被推至Queue中。将目标节点推到Queue中会使目标节点K的直接前驱节点J的依赖计数减一。因此,节点J具有零依赖性计数,并且Queue=[K]。在第二步,仅节点K从Queue弹出,然后K被推入SettleOrder数组。由于K是目标节点,因此其预期时间奖励114可以被设置为大的固定常数,比如600秒。

[0098] 节点J是K的唯一直接前驱。其具有零依赖性计数,因此可被推到Queue中。因此,Queue=[J]。E和I的依赖计数变成1。然后,只有节点J从Queue弹出,并且J被推到SettleOrder数组中。这导致:

$$[0099] \quad \text{Reward}(J) = \text{Reward}(K) - \text{ActionCost}(JK) = 600 - 20 = 580,$$

[0100] 以及SettleOrder=[K,J]。

[0101] 由于E和I的依赖性计数不为零,因此它们不被推送到Queue中。Queue为空。现在,节点L是具有零依赖性计数的唯一节点,因此将L推至Queue中。然后从Queue弹出L,并且L被

推入SettleOrder数组。L的奖励为零。L的所有直接前驱被解决，并且Queue再次为空，其中SettleOrder=[K,J,L]。

[0102] 修改值迭代方法进一步包括从图中移除一条边。然后检查NodeOrder数组。不移除指向已解决节点K、J、L的边，但是可移除指向未解决节点I的任何边。因此，该方法还包括移除边HI。

[0103] 接下来，节点H的依赖性计数变为零，因此其可以被推到Queue中。执行该过程直至节点E被解决，得到SettleOrder=[K,J,L,H,G,F,I,E]，Queue为空，并且H、G、F的奖励分别为0、-10、-20。有两个动作从I、IJ和IF开始。动作IF总是成功的，因此其具有奖励-25。IJ的奖励是：

[0104]  $0.9(\text{Reward}(J) - 20) + 0.1(\text{Reward}(F) - 20) = 500$ 。

[0105] 这意味着 $\text{Reward}(I) = 500$  (500和-25中的最大值)。

[0106] 类似地， $\text{Reward}(E) = 280$ 。B的依赖性计数是1。在此操作中，从NodeOrder数组移除边CD。这确保节点C和B分别以奖励0和255解决。节点A和节点D分别以奖励245和230来解决。所有节点被解决，因此操作4结束。总结操作4，SettleOrder数组是[K,J,L,H,G,F,I,E,C,B,A,D]，移除两条边HI和CD，并且节点奖励如图3C所示。

[0107] 在操作5，使用SettleOrder数组再次更新每个节点的奖励并且计算图3D中所示的结果，将边HI和CD添加回图。除了L，所有节点携带正的奖励。它暗示对于除L之外的任何节点，存在从该节点行进到目标节点K的路径。

[0108] 行为规划器

[0109] 转向行为规划器140，行为规划器140规划更直接的行为。行为可以包括以下项中的任一项：用于实现车道保持的运动、用于实现变道的运动、用于转向的运动、选择岔路的运动、或用于并线的运动。运动可被定义达固定时间段或被定义达固定距离。在操作上，世界模型输出可以用于障碍物、路径和等待条件以将这些优化类别作为奖励来处理。例如，行为规划器140可以使用针对不平稳运动如何影响舒适度—以及不平稳运动规划如何在等效时间方面带来预期的舒适度惩罚的模型。行为规划器140还可以使用即将到来的节点的预期等效时间来考虑向目标奖励的进展，并且可以对到达其附近的节点所花费的时间和资源打折。行为规划器140进行操作以到达非常接近的节点—比更接近目标的当前节点具有更高的预期奖励—花费更少的时间。

[0110] 行为规划器140与车道规划器对接以支持行为规划。在操作上，行为规划器140接收包括大车道图130的节点的预期等效时间奖励114的车道规划器输出数据，并且包括将节点链接到行为规划中更精确的局部几何和即时动作。行为规划器140可利用由链接到大车道图130的局部世界模型定义的数据结构中更详细的车道图(例如，局部车道图)来操作。局部车道图(例如，基于地图的局部车道图)可从用于创建大车道图130的相同底层地图数据生成。这允许大车道图130的节点与基于地图的局部车道图中的车道段之间的链路被精确地知道。可通过精确定位来生成基于地图的局部车道图，使得给定车道规划器输出数据，局部车道图包括用于任何规划的精确几何形状。以这种方式，车道规划器输出数据可以转化为例如变道或分岔指令。

[0111] 车道规划器输出数据还可以用于依次与任何其他局部车道图(如纯粹通过实时感知生成的那些局部车道图)执行匹配。通过那些链路，用于行驶的任何车道图可以与来自车

道规划器输出数据的预期等效时间相关联。此外,车道规划器输出数据的几何形状可以用于与实时感知进行匹配,并且几何形状比一组固定的车道名称(诸如左岔路或右岔路)更灵活,如果它是三向岔路则可能崩溃。因此,车道规划器输出数据可用于生成用于详细运动规划的车道图的精确几何结构,在沿着车道中心具有规则间距的顶点处具有相关联的预期等效时间。

[0112] 行为规划器140可以被配置成通过固定在一些靠近节点处的预期等效时间并且将它们视为目标节点来以与先前层相似的方式创建规划层级的一个或更多个层。然后,行为规划器的运动规划器142被应用于多个可能的动作—诸如,车道保持、变道、变道中止、转弯等。在操作上,对于每个动作,运动规划器测试多个运动规划,并针对成本和奖励项对它们进行评估。障碍物可以通过将安全系统或安全协议和指南集(如由NVIDIA公司开发的安全力场)视为硬约束和/或通过支付惩罚(例如,对于急动)来处理。可通过惩罚与车道中心的偏离来处理路径,并且除了产生约束之外,等待条件还可作为纵向进度上的边界约束来处理。可以通过从我们到达的节点的预期等效时间减去所花费的时间和资源来处理进度,和/或可以通过惩罚不平稳的运动规划来鼓励舒适。这样,来自不同动作的运动规划原则上可以在共同点上比较它们的优点,并且可以用作例如是否选择来自变道或车道保持动作的运动规划的基础。

[0113] 参见图1C中的行为规划器140,车道规划器输出数据可以用于行为规划器140中(例如,经由图1A中的运动规划器),例如,用于选择要行驶的车道。在C142,行为规划器140进行操作以处理车道规划器输出数据,以确定将由车辆在行驶路线上执行的识别动作。以这种方式,行为规划器140可以支持针对多个车道进行车道选择。行为规划器140接收多个局部车道(例如,图4A-4B中的车道402C、车道404C和车道406C)并且生成候选车辆动作和对应的行驶路线(例如,车道402C对应于左变道,车道404C对应于直行,并且车道406C对应于采取右岔路)。车道规划器110提供链接到局部世界地图车道的大车道图节点的预期等效时间奖励。行为规划器140识别多个车道并且在地图中理解它们的连接。然后,行为规划器140接收大车道图130中的节点(例如,节点402A、节点404A和节点406A)到多个车道中的每个车道(例如,分别对应于节点402B、节点404B和节点406B)的分配(例如,图1B中具有局部车道位置的大车道图节点144)。这种分配是可能的,因为局部车道可从用于创建大车道图130的相同底层地图数据生成。如果通过其他方法生成局部车道,诸如纯粹地通过实时感知生成的那些,则可以在基于地图的车道和实时感知的车道之间执行匹配,使得可以将大车道图节点分配给实时感知的车道。执行将节点分配至多个车道是为了导出多个车道中的每个车道的预期等效时间奖励,并且预期等效时间奖励114(例如,40秒、0秒、0秒)可以是节点预期等效时间奖励114减去在当前车辆位置(图4B中具有字母e的框)与节点之间的等效时间成本,在零秒处截断。当前车辆位置和节点之间的等效时间成本可以基于车道规划器110未考虑的局部实时感知来确定。例如,车辆车道404C具有零的预期等效时间奖励,因为车辆正前方的静止障碍物正阻碍车辆向前行驶(或前进),即使在节点404B处的预期时间奖励被车道规划器110确定为正的。因此,行为规划器140选择最大化预期时间奖励的车辆动作,至少部分地基于车辆动作来控制车辆沿着行驶路线。在该示例中,车辆可以执行左变道(例如,选择车道402C),因为车辆的车道402C左侧的车道具有最高的预期等效时间奖励。

[0114] 现在参见图5,在此描述的方法500的每个框包括可以使用硬件、固件和/或软件的

任何组合执行的计算过程。例如,不同功能可以通过处理器执行存储在存储器中的指令来实现。所述方法还可体现为存储于计算机存储介质上的计算机可用指令。仅举几例,所述方法可由独立应用、服务或托管服务(独立或与其他托管服务组合)或另一产品的插件提供。此外,通过示例的方式,关于图1的系统100描述了方法500。然而,这些方法可以另外地或可替代地由任何一个系统或系统的任何组合来执行,这些系统包括但不限于在此描述的那些系统。然而,这些方法可以另外地或可替代地由任何一个系统或系统的任何组合来执行,包括但不限于在此描述的那些系统。

[0115] 图5是示出根据本公开的一些实施例的用于操作车道规划器以生成车道规划器输出数据的方法500的流程图。在框B502处,方法500包括接收表示与车辆的行驶路线相关联的车道图的至少一部分的数据。大车道图包括指示在大车道图的一个或多个车道内的潜在位置的多个节点以及连接多个节点的多个边。如为了清楚起见描绘的,多个边中的每个边对应于一个或多个车道(即,局部车道)内的车辆的相应车辆动作。大车道图是有向图数据结构,所述有向图数据结构至少包括多个节点,所述多个节点包括目标节点和源节点。多个节点和多个边对应于定义用于确定行驶路线的状态和动作空间的动作。

[0116] 在框B504处,方法500包括为对应于车辆动作的多个边中的每个边,至少部分地基于用于在起始节点与起始节点的连接节点之间穿行的时间成本和与节点成功相关联的车辆动作的概率,来计算成本函数。在一个或多个实施例中,可以针对多个边中的每个边执行这种计算。在框506处,方法500包括计算多个节点中的每个节点的预期时间奖励是至少部分地基于执行修改值迭代,该修改值迭代控制用于执行计算预期等效时间奖励的修改值迭代操作的迭代次数。控制用于执行修改值迭代操作的迭代次数是基于至少一个停止标准。

[0117] 在框508处,方法500包括至少部分地基于一个或多个车道,生成候选车辆动作和针对每个候选车辆动作的一个或多个行驶路线。在框510处,方法500包括至少部分地基于与每个节点相关联的预期等效时间奖励,确定行驶路线以使得控制车辆沿着该行驶路线。

[0118] 现在参见图6,图6是示出根据本公开的一些实施例的用于操作驾驶系统以提供用于驾驶规划的车道规划器输出数据的方法600的流程图。在框B602处,方法600包括基于至少一个停止标准执行修改值迭代操作。在框B604处,方法600包括基于至少一个停止标准控制用于执行修改值迭代操作的迭代次数。在框B606处,方法600包括:至少部分地基于执行控制修改值迭代的迭代次数的修改值迭代,计算包括多个节点中的源节点和目标节点的多个节点中的每个节点的预期时间奖励;以及将具有预期时间等效时间奖励的多个节点中的一个或多个节点分配至与候选车辆动作相关联的行驶路线。与多个节点相关联的多个边中的每个边的预期等效时间成本是至少部分地基于在起始节点与节点的连接节点之间穿行的时间成本以及与节点成功相关联的车辆动作成功的概率。此外,在多次迭代的每次迭代中更新每个节点的预期等效时间奖励。

[0119] 示例自主车辆

[0120] 图7A为根据本公开一些实施例的示例自主车辆700的图示。自主车辆700(可替代地,在本文称为“车辆700”)可以包括但不限于乘用车,例如汽车、卡车、公共汽车、急救车、穿梭车、电动或机动自行车、摩托车、消防车、警车、救护车、船、工程车辆、水下船只、无人

机、连接拖车的车辆和/或其他类型的车辆(例如,无人驾驶和/或可容纳一名或多名乘客)。自主车辆通常按照美国运输部的一个部门——国家公路交通安全管理局(NHTSA)以及汽车工程师协会(SAE)“Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles”(2018年6月15日发布的标准No.J3016-201806,2016年9月30日发布的标准No.J3016-201609,以及该标准的先前和未来的版本)定义的自动化级别进行描述。车辆700能够实现符合自动驾驶级别的3级-5级中的一者或更多者的功能。车辆700能够实现符合自动驾驶级别的1级-5级中的一者或更多者的功能。例如,取决于实施例,车辆700能够实现驾驶员辅助(1级)、部分自动化(2级)、有条件的自动化(3级)、高度自动化(4级)和/或全自动化(5级)的能力。这里所使用的“自主”一词,可以包括700或其他机器的任何和/或所有类型的自主,如完全自主、高度自主、有条件的自主、部分自主、提供辅助自主、半自主、主要自主或其他名称。

[0121] 车辆700可以包括诸如底盘、车身、车轮(例如2个、4个、6个、8个、18个等)、轮胎、车轴之类的部件以及车辆的其他部件。车辆700可以包括推进系统750,例如内燃机、混合动力发电厂、全电动发动机和/或另一种推进系统类型。推进系统750可以连接到可以包括变速器的车辆700的传动系以便实现车辆700的推进。可以响应于接收到来自油门/加速器752的信号而控制推进系统750。

[0122] 可以包括方向盘的转向系统754可以用来在推进系统750操作时(例如在车辆运动时)使车辆700转向(例如沿着希望的路径或路线)。转向系统754可以接收来自转向致动器756的信号。对于全自动(5级)功能而言,方向盘可以是可选的。

[0123] 制动传感器系统746可以用来响应于接收到来自制动致动器748和/或制动传感器的信号而操作车辆制动器。

[0124] 可以包括一个或更多个片上系统(SoC)704(图7C)和/或一个或更多个GPU的一个或更多个控制器736可以向车辆700的一个或更多个部件和/或系统提供(例如表示命令的信号)。例如,一个或更多个控制器可以发送经由一个或更多个制动致动器748操作车辆制动器、经由一个或更多个转向致动器756操作转向系统754、经由一个或更多个油门/加速器752操作推进系统750的信号。一个或更多个控制器736可以包括一个或更多个板载(例如集成)计算设备(例如超级计算机),所述计算设备处理传感器信号并且输出操作命令(例如表示命令的信号),以实现自动驾驶和/或辅助人类驾驶员驾驶车辆700。一个或更多个控制器736可以包括用于自动驾驶功能的第一控制器736、用于功能性安全功能的第二控制器736、用于人工智能功能(例如计算机视觉)的第三控制器736、用于信息娱乐功能的第四控制器736、用于紧急情况下的冗余的第五控制器736和/或其他控制器。在一些示例中,单个控制器736可以处理上述功能中的两个或更多,两个或更多控制器736可以处理单个功能,和/或其任意组合。

[0125] 一个或更多个控制器736可以响应于接收自一个或更多个传感器的传感器数据(例如传感器输入),提供用于控制车辆700的一个或更多个部件和/或系统的信号。传感器数据可以接收自例如且不限于全球导航卫星系统传感器758(例如全球定位系统传感器)、RADAR传感器760、超声传感器762、LIDAR传感器764、惯性测量单元(IMU)传感器766(例如加速度计、陀螺仪、磁罗盘、磁力计等)、麦克风796、立体相机768、广角相机770(例如鱼眼相机)、红外相机772、环绕相机774(例如360度相机)、远程和/或中程相机798、速度传感器744

(例如用于测量车辆700的速率)、振动传感器742、转向传感器740、制动传感器(例如作为制动传感器系统546的部分)和/或其他传感器类型。

[0126] 控制器736中的一个或多个可以接收来自车辆700的仪表组732的输入(例如由输入数据表示),并且经由人机接口(HMI)显示器734、听觉信号器、扬声器和/或经由车辆700的其他部件提供输出(例如输出数据、显示数据等表示的)。这些输出可以包括诸如车辆速度、速率、时间、地图数据(例如图7C的HD地图722)、位置数据(例如,车辆700例如在地图上的位置)、方向、其他车辆的位置(例如占用网格)之类的信息,如控制器736所感知的关于对象和对象状态的信息等等。例如,HMI显示器734可以显示关于一个或多个对象(例如街道指示牌、警示牌、交通灯变化等)的存在性的信息和/或关于车辆已经做出、正在做出或者将会做出的驾驶机动的信息(例如现在变道、两英里后离开34B,等等)。

[0127] 车辆700进一步包括网络接口724,其可以使用一个或多个无线天线726和/或调制解调器通过一个或多个网络通信。例如,网络接口724可能通过LTE、WCDMA、UMTS、GSM、CDMA2000等通信。一个或多个无线天线726也可以使用诸如蓝牙、蓝牙LE、Z波、ZigBee等等之类的一个或多个局域网和/或诸如LoRaWAN、SigFox等等之类的一个或多个低功率广域网(LPWAN)实现环境中的对象(例如车辆、移动设备等等)之间的通信。

[0128] 图7B为根据本公开一些实施例的用于图7A的示例自主车辆700的相机位置和视场的示例。相机和各自的视场是一个示例实施例,并不意图是限制性的。例如,可以包括附加的和/或可替换的相机,和/或这些相机可以位于车辆700上的不同位置。

[0129] 用于相机的相机类型可以包括但不限于可以适于与车辆700的部件和/或系统一起使用的数字相机。所述相机可以在汽车安全完整性级别(ASIL)B下和/或在另一个ASIL下操作。相机类型可以具有任何图像捕获率,例如60帧每秒(fps)、120fps、240fps等等,这取决于实施例。相机可能使用滚动快门、全局快门、另一种类型的快门或者其组合。在一些示例中,滤色器阵列可以包括红白白白(RCCC)滤色器阵列、红白白蓝(RCCB)滤色器阵列、红蓝绿白(RBGC)滤色器阵列、Foveon X3滤色器阵列、拜耳传感器(RGGB)滤色器阵列、单色传感器滤色器阵列和/或另一种类型的滤色器阵列。在一些实施例中,诸如具有RCCC、RCCB和/或RBGC滤色器阵列的相机之类的清晰像素相机可以用在提高光敏感度的努力中。

[0130] 在一些示例中,所述相机中的一个或多个可以用来执行高级驾驶员辅助系统(ADAS)功能(例如作为冗余或故障安全设计的部分)。例如,可以安装多功能单目相机以提供包括车道偏离警告、交通指示牌辅助和智能前照灯控制在内的功能。所述相机中的一个或多个(例如全部相机)可以同时记录和提供图像数据(例如视频)。

[0131] 所述相机中的一个或多个可以安装在诸如定制设计的(3-D打印的)组件之类的安装组件中,以便切断可能干扰相机的图像数据捕获能力的杂散光和来自汽车内的反射(例如挡风玻璃镜中反射的来自仪表板的反射)。关于翼镜安装组件,翼镜组件可以是定制3-D打印的,使得相机安装板匹配翼镜的形状。在一些示例中,一个或多个相机可以集成到翼镜中。对于侧视相机而言,一个或多个相机也可以集成到驾驶室每个拐角的四根柱子内。

[0132] 具有包括车辆700前面的环境部分的视场的相机(例如前置相机)可以用于环视,以帮助识别前向路径和障碍,以及在一个或多个控制器736和/或控制SoC的帮助下辅助提供对于生成占用网格和/或确定优选车辆路径至关重要的信息。前置相机可以用来执行

许多与LIDAR相同的ADAS功能,包括紧急制动、行人检测和碰撞避免。前置相机也可以用于ADAS功能和系统,包括车道偏离警告(LDW)、自主巡航控制(ACC),和/或诸如交通指示牌识别之类的其他功能。

[0133] 各种各样的相机可以用于前置配置中,包括例如包括CMOS(互补金属氧化物半导体)彩色成像仪在内的单目相机平台。另一个示例可以是广角相机770,其可以用来感知从周边进入视野的对象(例如行人、十字路口交通或者自行车)。尽管图7B中图示出仅仅一个广角相机,但是在车辆700上可以存在任意数量的广角相机770。此外,远程相机798(例如长视立体相机对)可以用于基于深度的对象检测,尤其是用于尚未针对其训练神经网络的对象。远程相机798也可以用于对象检测和分类以及基本的对象跟踪。

[0134] 一个或更多个立体相机768也可以包括在前置配置中。立体相机768可以包括集成控制单元,该单元包括可扩展处理单元,其可以提供在单个芯片上具有集成的CAN或以太网接口的多核微处理器和可编程逻辑(FPGA)。这样的单元可以用来生成车辆环境的3-D地图,包括针对图像中的所有点的距离估计。可替代的立体相机768可以包括紧凑型立体视觉传感器,其可以包括两个相机镜头(左右各一个)以及可以测量从车辆到目标对象的距离并且使用生成的信息(例如元数据)激活自主紧急制动和车道偏离警告功能的图像处理芯片。除了本文所描述的那些之外或者可替代地,可以使用其他类型的立体相机768。

[0135] 具有包括车辆700的侧面的环境部分的视场的相机(例如侧视相机)可以用于环视,提供用来创建和更新占用网格以及生成侧撞击碰撞警告的信息。例如,环绕相机774(例如如图7B中所示的四个环绕相机774)可以置于车辆700上。环绕相机774可以包括广角相机770、鱼眼相机、360度相机和/或类似物。四个示例,四个鱼眼相机可以置于车辆的前面、后面和侧面。在一种可替代的布置中,车辆可以使用三个环绕相机774(例如左边、右边和后面),并且可以利用一个或更多个其他相机(例如前向相机)作为第四环视相机。

[0136] 具有包括车辆700的后面的环境部分的视场的相机(例如后视相机)可以用于辅助停车、环视、后面碰撞警告以及创建和更新占用网格。可以使用各种各样的相机,包括但不限于也适合作为如本文所描述的前置相机(例如远程和/或中程相机798、立体相机768、红外相机772等等)的相机。

[0137] 图7C为根据本公开一些实施例的用于图7A的示例自主车辆700的示例系统架构的框图。应当理解,这种布置和本文描述的其他布置仅仅作为示例而被阐述。除了所示的那些之外或者代替它们的是,可以使用其他的布置和元素(例如机器、接口、功能、顺序、功能分组等等),并且一些元素可以完全省略。进一步,许多本文描述的元素是功能实体,其可以实现为分立的或分布式部件或者结合其他部件实现,以及以任何适当的组合和位置实现。本文描述为由实体执行的各个功能可以通过硬件、固件和/或软件实现。例如,各个功能可以通过处理器执行存储在内存中的指令而实现。

[0138] 图7C中车辆700的部件、特征和系统中的每一个被图示为经由总线702连接。总线702可以包括控制器区域网络(CAN)数据接口(可替代地,本文称为“CAN总线”)。CAN可以是车辆700内部的网络,用来辅助控制车辆700的各种特征和功能,例如制动器、加速、制动、转向、挡风玻璃雨刷等等的驱动。CAN总线可以被配置为具有数十或者甚至数百个节点,每个节点具有其自己的唯一标识符(例如CAN ID)。可以读取CAN总线以找到方向盘角度、车速、每分钟发动机转速(RPM)、按钮位置和/或其他车辆状态指示符。CAN总线可以是ASIL B兼容

的。

[0139] 尽管这里将总线702描述为CAN总线,但是这并不意图是限制性的。例如,除了CAN总线之外或者可替代地,可以使用FlexRay和/或以太网。此外,尽管用单条线来表示总线702,但是这并不意图是限制性的。例如,可以存在任意数量的总线702,其可以包括一条或更多条CAN总线、一条或更多条FlexRay总线、一条或更多条以太网总线和/或一条或更多条使用不同协议的其他类型的总线。在一些示例中,两条或更多总线702可以用来执行不同的功能,和/或可以用于冗余。例如,第一总线702可以用于碰撞避免功能,并且第二总线702可以用于驱动控制。在任何示例中,每条总线702可以与车辆700的任何部件通信,并且两条或更多总线702可以与相同的部件通信。在一些示例中,车辆内的每个SoC 704、每个控制器736和/或每个计算机可以有权访问相同的输入数据(例如来自车辆700的传感器的输入),并且可以连接到诸如CAN总线之类的公共总线。

[0140] 车辆700可以包括一个或更多个控制器736,例如本文关于图7A所描述的那些控制器。控制器736可以用于各种各样的功能。控制器736可以耦合到车辆700的任何其他不同的部件和系统,并且可以用于车辆700的控制、车辆700的人工智能、用于车辆700的信息娱乐和/或类似物。

[0141] 车辆700可以包括一个或更多个片上系统(SoC)704。SoC 704可以包括CPU 706、GPU 708、处理器710、高速缓存712、加速器714、数据存储716和/或未图示出的其他部件和特征。在各种各样的平台和系统中,SoC 704可以用来控制车辆700。例如,一个或更多个SoC 704可以在系统(例如车辆700的系统)中与HD地图722结合,所述HD地图可以经由网络接口724从一个或更多个服务器(例如图7D的一个或更多个服务器778)获得地图刷新和/或更新。

[0142] CPU 706可以包括CPU簇或者CPU复合体(可替代地,本文称为“CCPLEX”)。CPU 706可以包括多个核和/或L2高速缓存。例如,在一些实施例中,CPU 706在一致性多处理器配置中可以包括八个核。在一些实施例中,CPU 706可以包括四个双核簇,其中每个簇具有专用的L2高速缓存(例如2MB L2高速缓存)。CPU 706(例如CCPLEX)可以被配置为支持同时簇操作,使得CPU 706的簇的任意组合能够在任何给定时间是活动的。

[0143] CPU 706可以实现包括以下特征中的一个或更多个的功率管理能力:各硬件块在空闲时可以自动进行时钟门控以节省动态功率;由于WFI/WFE指令的执行,每个核时钟可以在该核不主动地执行指令时进行门控;每个核可以独立地进行功率门控;当所有核都进行时钟门控或者功率门控时,可以独立地对每个核簇进行时钟门控;和/或当所有核都进行功率门控时,可以独立地对每个核簇进行功率门控。CPU 706可以进一步实现用于管理功率状态的增强算法,其中指定允许的功率状态和预期的唤醒时间,并且硬件/微代码为所述核、簇和CCPLEX确定要进入的最佳的功率状态。处理核可以在软件中支持简化的功率状态进入序列,该工作被卸载到微代码。

[0144] GPU 708可以包括集成的GPU(可替代地,本文称为“iGPU”)。GPU708可以是可编程的,并且对于并行工作负载而言是高效的。在一些示例中,GPU 708可以使用增强张量指令集。GPU 708可以包括一个或更多个流式微处理器,其中每个流式微处理器可以包括L1高速缓存(例如具有至少96KB存储能力的L1高速缓存),并且这些流式微处理器中的两个或更多可以共享L2高速缓存(例如具有512KB存储能力的L2高速缓存)。在一些实施例中,GPU 708

可以包括至少八个流式微处理器。GPU 708可以使用计算应用编程接口 (API)。此外, GPU 708可以使用一个或更多个并行计算平台和/或编程模型 (例如NVIDIA的CUDA)。

[0145] 在汽车和嵌入式使用的情况下,可以对GPU 708进行功率优化以实现最佳性能。例如,可以在鳍式场效应晶体管 (FinFET) 上制造GPU 708。然而,这并不意图是限制性的,并且GPU 708可以使用其他半导体制造工艺来制造。每个流式微处理器可以合并划分成多个块的若干混合精度处理核。例如且非限制性地,可以将64个PF32核和32个PF64核划分成四个处理块。在这样的示例中,每个处理块可以分配16个FP32核、8个FP64核、16个INT32核、用于深度学习矩阵算术的两个混合精度NVIDIA张量核、L0指令高速缓存、线程束 (warp) 调度器、分派单元和/或64KB寄存器文件。此外,流式微处理器可以包括独立的并行整数和浮点数据路径,以利用计算和寻址计算的混合提供工作负载的高效执行。流式微处理器可以包括独立线程调度能力,以允许实现并行线程之间的更细粒度的同步和协作。流式微处理器可以包括组合的L1数据高速缓存和共享内存单元,以便在简化编程的同时提高性能。

[0146] GPU 708可以包括在一些示例中提供大约900GB/s的峰值内存带宽的高带宽内存 (HBM) 和/或16GB HBM2内存子系统。在一些示例中,除了HBM内存之外或者可替代地,可以使用同步图形随机存取存储器 (SGRAM),例如第五代图形双倍数据速率同步随机存取存储器 (GDDR5)。

[0147] GPU 708可以包括统一内存技术,其包括访问计数器以允许内存页面更精确地迁移到最频繁地访问它们的处理器,从而提高处理器之间共享的内存范围的效率。在一些示例中,地址转换服务 (ATS) 支持可以用来允许GPU 708直接访问CPU 706页表。在这样的示例中,当GPU 708内存管理单元 (MMU) 经历遗漏时,可以将地址转换请求传输至CPU 706。作为响应,CPU 706可以在其页表中寻找用于地址的虚拟-物理映射,并且将转换传输回GPU 708。这样,统一内存技术可以允许单个统一虚拟地址空间用于CPU 706和GPU 708二者的内存,从而简化了GPU 708编程和将应用程序移 (port) 到GPU 708。

[0148] 此外, GPU 708可以包括访问计数器,其可以跟踪GPU 708访问其他处理器的内存的频率。访问计数器可以帮助确保内存页面移至最频繁地访问这些页面的处理器的物理内存。

[0149] SoC 704可以包括任意数量的高速缓存712,包括本文描述的那些高速缓存。例如,高速缓存712可以包括CPU 706和GPU 708二者可用的L3高速缓存 (例如,其连接到CPU 706和GPU 708二者)。高速缓存712可以包括回写高速缓存,其可以例如通过使用高速缓存一致性协议 (例如MEI、MESI、MSI等) 跟踪行的状态。取决于实施例,L3高速缓存可以包括4MB或者更多,但是也可以使用更小的高速缓存大小。

[0150] SoC 704可包括一个或更多个算术逻辑单元 (ALU),其可用于执行关于车辆700的各种任务或操作中的任一个的处理,例如处理DNN。此外,SoC 704可以包括浮点单元 (FPU) 或其他数学协处理器或数字协处理器类型,用于在系统内执行数学运算。例如,SoC 704可以包括一个或更多个集成为CPU 706和/或GPU 708内的执行单元的FPU。

[0151] SoC 704可以包括一个或更多个加速器714 (例如硬件加速器、软件加速器或者其组合)。例如,SoC 704可以包括硬件加速簇,其可以包括优化的硬件加速器和/或大型片上内存。该大型片上内存 (例如4MB SRAM) 可以使得硬件加速簇能够加速神经网络和其他计算。硬件加速簇可以用来补充GPU 708,并且卸载GPU 708的一些任务 (例如释放GPU 708的

更多周期以用于执行其他任务)。作为一个示例,加速器714可以用于足够稳定以易于控制加速的有针对性的工作负载(例如感知、卷积神经网络(CNN)等等)。当在本文中使用时,术语“CNN”可以包括所有类型的CNN,包括基于区域的或者区域卷积神经网络(RCNN)和快速RCNN(例如用于对象检测)。

[0152] 加速器714(例如硬件加速簇)可以包括深度学习加速器(DLA)。DLA可以包括可以被配置成为深度学习应用和推理提供额外的每秒10万亿次操作的一个或更多个张量处理单元(TPU)。TPU可以是被配置为执行图像处理功能(例如用于CNN、RCNN等)且针对执行图像处理功能而优化的加速器。DLA可以进一步针对特定的一组神经网络类型和浮点运算以及推理进行优化。DLA的设计可以比通用GPU提供每毫米更高的性能,并且远远超过CPU的性能。TPU可以执行若干功能,包括单实例卷积函数,支持例如用于特征和权重二者的INT8、INT16和FP16数据类型,以及后处理器功能。

[0153] DLA可以在处理的或者未处理的数据上针对各种各样的功能中的任何功能快速且高效地执行神经网络,尤其是CNN,例如且不限于:用于使用来自相机传感器的数据进行对象识别和检测的CNN;用于使用来自相机传感器的数据进行距离估计的CNN;用于使用来自麦克风的数据进行应急车辆检测和识别与检测的CNN;用于使用来自相机传感器的数据进行面部识别和车主识别的CNN;和/或用于安全和/或安全相关事件的CNN。

[0154] DLA可以执行GPU 708的任何功能,并且通过使用推理加速器,例如,设计者可以使DLA或GPU 708针对任何功能。例如,设计者可以将CNN的处理和浮点运算聚焦在DLA上,并且将其他功能留给GPU 708和/或其他加速器714。

[0155] 加速器714(例如硬件加速簇)可以包括可编程视觉加速器(PVA),其在本文中可以被替代地称为计算机视觉加速器。PVA可以被设计和配置为加速用于高级驾驶员辅助系统(ADAS)、自动驾驶和/或增强现实(AR)和/或虚拟现实(VR)应用的计算机视觉算法。PVA可以提供性能与灵活性之间的平衡。例如,每个PVA可以包括例如且不限于任意数量的精简指令集计算机(RISC)核、直接内存访问(DMA)和/或任意数量的向量处理器。

[0156] RISC核可以与图像传感器(例如本文描述的任何相机的图像传感器)、图像信号处理器和/或类似物交互。这些RISC核中的每一个可以包括任意数量的内存。取决于实施例,RISC核可以使用若干协议中的任何协议。在一些示例中,RISC核可以执行实时操作系统(RTOS)。RISC核可以使用一个或更多个集成电路设备、专用集成电路(ASIC)和/或存储设备实现。例如,RISC核可以包括指令高速缓存和/或紧密耦合的RAM。

[0157] DMA可以使得PVA的部件能够独立于CPU 706访问系统内存。DMA可以支持用来向PVA提供优化的任意数量的特征,包括但不限于支持多维寻址和/或循环寻址。在一些示例中,DMA可以支持高达六个或更多维度的寻址,其可以包括块宽度、块高度、块深度、水平块步进、竖直块步进和/或深度步进。

[0158] 向量处理器可以是可编程处理器,其可以被设计为高效且灵活地执行用于计算机视觉算法的编程并且提供信号处理能力。在一些示例中,PVA可以包括PVA核和两个向量处理子系统分区。PVA核可以包括处理器子系统、一个或更多个DMA引擎(例如两个DMA引擎)和/或其他外围设备。向量处理子系统可以作为PVA的主处理引擎而操作,并且可以包括向量处理单元(VPU)、指令高速缓存和/或向量内存(例如VMEM)。VPU核可以包括数字信号处理器,诸如例如单指令多数据(SIMD)、超长指令字(VLIW)数字信号处理器。SIMD和VLIW的组合

可以增强吞吐量和速率。

[0159] 向量处理器中的每一个可以包括指令高速缓存并且可以耦合到专用内存。结果, 在一些示例中, 向量处理器中的每一个可以被配置为独立于其他向量处理器执行。在其他示例中, 包括在特定PVA中的向量处理器可以被配置为采用数据并行化。例如, 在一些实施例中, 包括在单个PVA中的多个向量处理器可以执行相同的计算机视觉算法, 但是在图像的不同区域上执行。在其他示例中, 包括在特定PVA中的向量处理器可以在相同的图像上同时执行不同的计算机视觉算法, 或者甚至在序列图像或者图像的部分上执行不同的算法。除其他的以外, 任意数量的PVA可以包括在硬件加速簇中, 并且任意数量的向量处理器可以包括在这些PVA中的每一个中。此外, PVA可以包括附加的纠错码 (ECC) 内存, 以增强总体系统安全性。

[0160] 加速器714 (例如硬件加速簇) 可以包括片上计算机视觉网络和SRAM, 以提供用于加速器714的高带宽、低延迟SRAM。在一些示例中, 片上内存可以包括由例如且不限于八个现场可配置的内存块组成的至少4MB SRAM, 其可以由PVA和DLA二者访问。每对内存块可以包括高级外围总线 (APB) 接口、配置电路系统、控制器和复用器。可以使用任何类型的内存。PVA和DLA可以经由向PVA和DLA提供高速内存访问的主干 (backbone) 访问内存。主干可以包括 (例如使用APB) 将PVA和DLA互连到内存的片上计算机视觉网络。

[0161] 片上计算机视觉网络可以包括在传输任何控制信号/地址/数据之前确定PVA和DLA二者都提供就绪且有效的信号的接口。这样的接口可以提供用于传输控制信号/地址/数据的单独相位和单独信道, 以及用于连续数据传输的突发式通信。这种类型的接口可以符合ISO 26262或者IEC 61508标准, 但是也可以使用其他标准和协议。

[0162] 在一些示例中, SoC 704可以包括例如在2018年8月10日提交的美国专利申请No. 16/101, 232中描述的实时光线追踪硬件加速器。该实时光线追踪硬件加速器可以用来快速且高效地确定 (例如世界模型内的) 对象的位置和范围, 以便生成实时可视化仿真, 以用于RADAR信号解释、用于声音传播合成和/或分析、用于SONAR系统仿真、用于一般波传播仿真、用于为了定位和/或其他功能的目的与LIDAR数据相比较和/或用于其他用途。在一些实施例中, 一个或更多个树遍历单元 (TTU) 可用于执行一个或更多个光线追踪相关操作。

[0163] 加速器714 (例如硬件加速器簇) 具有广泛的自动驾驶用途。PVA可以是可编程视觉加速器, 其可以用于ADAS和自主车辆中的关键处理阶段。PVA的能力是需要可预测处理、低功率和低延迟的算法域的良好匹配。换言之, PVA在半密集或者密集规则计算上, 甚至在需要具有低延迟和低功率的可预测运行时间的小数据集上都表现良好。因此, 在用于自主车辆的平台的背景下, PVA被设计为运行经典计算机视觉算法, 因为它们在对象检测和整数数学运算方面很有效。

[0164] 例如, 根据该技术的一个实施例, PVA用来执行计算机立体视觉。在一些示例中, 可以使用基于半全局匹配的算法, 但是这并不意图是限制性的。许多用于3级-5级自动驾驶的应用都需要即时运动估计/立体匹配 (例如来自运动的结构、行人识别、车道检测等等)。PVA可以在来自两个单目相机的输入上执行计算机立体视觉功能。

[0165] 在一些示例中, PVA可以用来执行密集的光流。根据过程原始RADAR数据 (例如使用4D快速傅立叶变换) 以提供经处理的RADAR。在其他示例中, PVA用于飞行时间深度处理, 其例如通过处理原始飞行时间数据以提供经处理的飞行时间数据。

[0166] DLA可以用来运行任何类型的网络以增强控制和驾驶安全性,包括例如输出用于每个对象检测的置信度度量的神经网络。这样的置信度值可以解释为概率,或者解释为提供每个检测与其他检测相比的相对“权重”。该置信度值使得系统能够做出关于哪些检测应当被认为是真阳性检测而不是假阳性检测的进一步决策。例如,系统可以为置信度设置阈值,并且仅仅将超过阈值的检测看作真阳性检测。在自动紧急制动(AEB)系统中,假阳性检测会使得车辆自动地执行紧急制动,这显然是不希望的。因此,只有最确信的检测才应当被认为是AEB的触发因素。DLA可以运行用于回归置信度值的神经网络。该神经网络可以将至少一些参数子集作为其输入,例如边界框维度,(例如从另一个子系统)获得的地平面估计,与车辆700取向、距离相关的惯性测量单元(IMU)传感器766输出,从神经网络和/或其他传感器(例如LIDAR传感器764或RADAR传感器760)获得的对象的3D位置估计等。

[0167] SoC 704可以包括一个或更多个数据存储716(例如内存)。数据存储716可以是SoC 704的片上内存,其可以存储要在GPU和/或DLA上执行的神经网络。在一些示例中,为了冗余和安全,数据存储716可以容量足够大以存储神经网络的多个实例。数据存储712可以包括L2或L3高速缓存712。对数据存储716的引用可以包括对与如本文所描述的PVA、DLA和/或其他加速器714关联的内存的引用。

[0168] SoC 704可以包括一个或更多个处理器710(例如嵌入式处理器)。处理器710可以包括启动和功率管理处理器,其可以是用于处理启动功率和管理功能以及有关安全实施的专用处理器和子系统。启动和功率管理处理器可以是SoC 704启动序列的一部分,并且可以提供运行时间功率管理服务。启动功率和管理处理器可以提供时钟和电压编程、辅助系统低功率状态转换、SoC 704热和温度传感器管理和/或SoC 704功率状态管理。每个温度传感器可以实现为环形振荡器,其输出频率与温度成比例,并且SoC704可以使用环形振荡器检测CPU 706、GPU 708和/或加速器714的温度。如果确定温度超过阈值,那么启动和功率管理处理器可以进入温度故障例程并且将SoC 704置于较低功率状态和/或将车辆700置于司机安全停车模式(例如使车辆700安全停车)。

[0169] 处理器710可以进一步包括可以用作音频处理引擎的一组嵌入式处理器。音频处理引擎可以是一种音频子系统,其允许实现对于通过多个接口的多声道音频的完全硬件支持以及一系列广泛而灵活的音频I/O接口。在一些示例中,音频处理引擎是具有带有专用RAM的数字信号处理器的专用处理器核。

[0170] 处理器710可以进一步包括始终在处理器上的引擎,其可以提供必要的硬件特征以支持低功率传感器管理和唤醒用例。该始终在处理器上的引擎可以包括处理器核、紧密耦合的RAM、支持外围设备(例如定时器和中断控制器)、各种I/O控制器外围设备和路由逻辑。

[0171] 处理器710可以进一步包括安全簇引擎,其包括处理汽车应用的安全管理的专用处理器子系统。安全簇引擎可以包括两个或更多处理器核、紧密耦合的RAM、支持外围设备(例如定时器、中断控制器等等)和/或路由逻辑。在安全模式下,所述两个或更多核可以操作于锁步模式下,并且用作具有检测它们的操作之间的任何差异的比较逻辑的单核。

[0172] 处理器710可以进一步包括实时相机引擎,其可以包括用于处理实时相机管理的专用处理器子系统。

[0173] 处理器710可以进一步包括高动态范围信号处理器,其可以包括图像信号处理器,

该图像信号处理器是一种硬件引擎,该硬件引擎是相机处理管线的部分。

[0174] 处理器710可以包括可以是(例如微处理器上实现的)处理块的视频图像复合器,其实现视频回放应用程序产生用于播放器窗口的最终图像所需的视频后处理功能。视频图像复合器可以对广角相机770、环绕相机774和/或对驾驶室内监控相机传感器执行镜头畸变校正。驾驶室内监控相机传感器优选地由运行在高级SoC的另一个实例上的神经网络监控,被配置为识别驾驶室内事件并且相对地做出响应。驾驶室内系统可以执行唇读,以激活移动电话服务并拨打电话、口述电子邮件、改变车辆目的地、激活或改变车辆的信息娱乐系统和设置或者提供语音激活的网上冲浪。某些功能仅在车辆操作于自主模式下时对于驾驶员可用,并且在其他情况下被禁用。

[0175] 视频图像复合器可以包括用于空间和时间降噪的增强时间降噪。例如,在视频中出现运动的情况下,降噪适当地对空间信息加权,降低邻近帧提供的信息的权重。在图像或者图像的部分不包括运动的情况下,视频图像复合器执行的时间降噪可以使用来自先前的图像的信息以降低当前图像中的噪声。

[0176] 视频图像复合器也可以被配置为对输入立体镜头帧执行立体校正。当操作系统桌面正在使用并且GPU 708无需连续地渲染(render)新的表面时,视频图像复合器可以进一步用于用户接口组成。甚至在GPU 708上电并且激活,进行3D渲染时,视频图像复合器可以用来减轻GPU 708的负担以提高性能和响应能力。

[0177] SoC 704可以进一步包括用于从相机接收视频和输入的移动行业处理器接口(MIPI)相机串行接口、高速接口和/或可以用于相机和有关像素输入功能的视频输入块。SoC 704可以进一步包括可以由软件控制并且可以用于接收未提交到特定角色的I/O信号的输入/输出控制器。

[0178] SoC 704可以进一步包括大范围的外围设备接口,以使能与外围设备、音频编解码器、功率管理和/或其他设备通信。SoC 704可以用来处理来自(通过千兆多媒体串行链路和以太网连接的)相机、传感器(例如可以通过以太网连接的LIDAR传感器764、RADAR传感器760等等)的数据,来自总线702的数据(例如车辆700的速率、方向盘位置等等),来自(通过以太网或CAN总线连接的)GNSS传感器758的数据。SoC 704可以进一步包括专用高性能大容量存储控制器,其可以包括它们自己的DMA引擎,并且其可以用来从日常数据管理任务中释放CPU 706。

[0179] SoC 704可以是具有灵活架构的端到端平台,该架构跨越自动化3级-5级,从而提供利用和高效使用计算机视觉和ADAS技术以实现多样性和冗余、连同深度学习工具一起提供用于灵活可靠驾驶软件堆栈的平台的综合功能安全架构。SoC 704可以比常规的系统更快、更可靠,甚至更加能量高效和空间高效。例如,当与CPU 706、GPU 708和数据存储716结合时,加速器714可以提供用于3级-5级自主车辆的快速高效平台。

[0180] 因此该技术提供了不能通过常规系统实现的能力和函数。例如,计算机视觉算法可以在CPU上执行,这些CPU可以使用诸如C编程语言之类的高级编程语言配置为跨各种各样的视觉数据执行各种各样的处理算法。然而,CPU常常不能满足许多计算机视觉应用的性能要求,诸如与例如执行时间和功耗有关的那些要求。特别地,许多CPU不能实时地执行复杂的对象检测算法,这是车载ADAS应用的要求和实用3-5级自主车辆的要求。

[0181] 与常规系统形成对比的是,通过提供CPU复合体、GPU复合体和硬件加速簇,本文描

述的技术允许同时和/或顺序地执行多个神经网络,并且将结果组合在一起以实现3-5级自动驾驶功能。例如,在DLA或dGPU(例如GPU 720)上执行的CNN可以包括文本和单词识别,允许超级计算机读取和理解交通指示牌,包括尚未针对其特别地训练神经网络的指示牌。DLA可以进一步包括能够识别、解释和提供对指示牌的语义理解,并且将该语义理解传递给运行在CPU复合体上的路径规划模块的神经网络。

[0182] 作为另一个示例,如3、4或5级驾驶所需的,多个神经网络可以同时运行。例如,由“注意:闪烁的灯指示结冰条件”组成的警告指示牌连同电灯可以由若干神经网络独立地或者共同地进行解释。指示牌本身可以由部署的第一神经网络(例如经过训练的神经网络)识别为交通指示牌,文本“闪烁的灯指示结冰条件”可以由部署的第二神经网络解释,该部署的第二神经网络告知车辆的路径规划软件(优选地在CPU复合体上执行)当检测到闪烁的灯时,存在结冰条件。闪烁的灯可以通过在多个帧上操作部署的第三神经网络而识别,该神经网络告知车辆的路径规划软件闪烁的灯的存在(或不存在)。所有三个神经网络可以例如在DLA内和/或在GPU 708上同时运行。

[0183] 在一些示例中,用于面部识别和车主识别的CNN可以使用来自相机传感器的数据识别车辆700的授权的驾驶员和/或车主的存在。始终在传感器上的处理引擎可以用来在车主接近驾驶员车门时解锁车辆并且打开灯,并且在安全模式下,在车主离开车辆时禁用车辆。按照这种方式,SoC 704提供了防范盗窃和/或劫车的安全性。

[0184] 在另一个示例中,用于应急车辆检测和识别的CNN可以使用来自麦克风796的数据来检测并且识别应急车辆警报(siren)。与使用通用分类器检测警报并且手动地提取特征的常规系统形成对比的是,SoC 704使用CNN以对环境 and 城市声音分类以及对视觉数据分类。在优选的实施例中,运行在DLA上的CNN被训练为识别应急车辆的相对关闭速率(例如通过使用多普勒效应)。CNN也可以被训练为识别如GNSS传感器758所识别的特定于车辆在其中操作的局部区域的应急车辆。因此,例如,当在欧洲操作时,CNN将寻求检测欧洲警报,并且当在美国时,CNN将寻求识别仅仅北美的警报。一旦检测到应急车辆,在超声传感器762的辅助下,控制程序可以用来执行应急车辆安全例程,使车辆放慢速度,开到路边,停下车辆,和/或使车辆空转,直到应急车辆通过。

[0185] 车辆可以包括可以经由高速互连(例如PCIe)耦合到SoC 704的CPU 718(例如分立的CPU或dCPU)。CPU 718可以包括例如X86处理器。CPU 718可以用来执行各种各样的功能中的任何功能,包括例如仲裁ADAS传感器与SoC 704之间潜在地不一致的结果,和/或监控控制器736和/或信息娱乐SoC 730的状态和健康状况。

[0186] 车辆700可以包括可以经由高速互连(例如NVIDIA的NVLINK)耦合到SoC 704的GPU 720(例如分立的GPU或dGPU)。GPU 720可以例如通过执行冗余的和/或不同的神经网络而提供附加的人工智能功能,并且可以用来基于来自车辆700的传感器的输入(例如传感器数据)来训练和/或更新神经网络。

[0187] 车辆700可以进一步包括网络接口724,该网络接口可以包括一个或更多个无线天线726(例如用于不同通信协议的一个或更多个无线天线,例如蜂窝天线、蓝牙天线等等)。网络接口724可以用来使能通过因特网与云(例如与服务器778和/或其他网络设备)、与其他车辆和/或与计算设备(例如乘客的客户端设备)的无线连接。为了与其他车辆通信,可以在这两辆车之间建立直接链接,和/或可以建立间接链接(例如跨网络以及通过因特网)。直

接链接可以使用车对车通信链路提供。车对车通信链路可以向车辆700提供关于接近车辆700的车辆(例如车辆700前面、侧面和/或后面的车辆)的信息。该功能可以是车辆700的协作自适应巡航控制功能的部分。

[0188] 网络接口724可以包括提供调制和解调功能并且使得控制器736能够通过无线网络通信的SoC。网络接口724可以包括用于从基带到射频的上转换以及从射频到基带的下转换的射频前端。频率转换可以通过公知的过程执行,和/或可以使用超外差(super-heterodyne)过程执行。在一些示例中,射频前端功能可以由单独的芯片提供。网络接口可以包括用于通过LTE、WCDMA、UMTS、GSM、CDMA2000、蓝牙、蓝牙LE、Wi-Fi、Z波、ZigBee、LoRaWAN和/或其他无线协议通信的无线功能。

[0189] 车辆700可以进一步包括可包括片外(例如SoC 704外)存储装置的数据存储728。数据存储728可以包括一个或多个存储元件,包括RAM、SRAM、DRAM、VRAM、闪存、硬盘和/或可以存储至少一个比特的数据的其他部件和/或设备。

[0190] 车辆700可以进一步包括GNSS传感器758。GNSS传感器758(例如GPS、辅助GPS传感器、差分GPS(DGPS)传感器等)用于辅助映射、感知、占用网格生成和/或路径规划功能。可以使用任意数量的GNSS传感器758,包括例如且不限于使用带有以太网到串行(RS-232)网桥的USB连接器的GPS。

[0191] 车辆700可以进一步包括RADAR传感器760。RADAR传感器760可以甚至在黑暗和/或恶劣天气条件下也由车辆700用于远程车辆检测。RADAR功能安全级别可以是ASIL B。RADAR传感器760可以使用CAN和/或总线702(例如以传输RADAR传感器760生成的数据)以用于控制以及访问对象跟踪数据,在一些示例中接入以太网以访问原始数据。可以使用各种各样的RADAR传感器类型。例如且非限制性地,RADAR传感器760可以适合前面、后面和侧面RADAR使用。在一些示例中,使用脉冲多普勒RADAR传感器。

[0192] RADAR传感器760可以包括不同的配置,例如具有窄视场的远程、具有宽视场的短程、短程侧覆盖等等。在一些示例中,远程RADAR可以用于自适应巡航控制功能。远程RADAR系统可以提供通过两个或更多独立浏览实现的广阔视场(例如250m范围内)。RADAR传感器760可以帮助区分静态对象和运动对象,并且可以由ADAS系统用于紧急制动辅助和前方碰撞警告。远程RADAR传感器可以包括具有多根(例如六根或更多)固定RADAR天线以及高速CAN和FlexRay接口的单站多模RADAR。在具有六根天线的示例中,中央四根天线可以创建聚焦的波束图案,其被设计为在更高速率下以来自邻近车道的最小交通干扰记录车辆700的周围环境。其他两根天线可以扩展视场,使得快速地检测进入或离开车辆700的车道的车辆成为可能。

[0193] 作为一个示例,中程RADAR系统可以包括高达760m(前面)或80m(后面)的范围以及高达42度(前面)或750度(后面)的视场。短程RADAR系统可以包括但不限于被设计为安装在后保险杠两端的RADAR传感器。当安装在后保险杠两端时,这样的RADAR传感器系统可以创建持续地监控后方和车辆旁边的盲点的两个波束。

[0194] 短程RADAR系统可以在ADAS系统中用于盲点检测和/或变道辅助。

[0195] 车辆700可以进一步包括超声传感器762。可以置于车辆700的前面、后面和/或侧面的超声传感器762可以用于停车辅助和/或创建和更新占用网格。可以使用各种各样的超声传感器762,并且不同的超声传感器762可以用于不同的检测范围(例如2.5m、4m)。超声传

感器762可以操作于功能安全级别的ASIL B。

[0196] 车辆700可以包括LIDAR传感器764。LIDAR传感器764可以用于对象和行人检测、紧急制动、碰撞避免和/或其他功能。LIDAR传感器764可以为功能安全级别的ASIL B。在一些示例中,车辆700可以包括可以使用以太网(例如以将数据提供给千兆以太网交换机)的多个LIDAR传感器764(例如两个、四个、六个等等)。

[0197] 在一些示例中,LIDAR传感器764可能能够对360度视场提供对象列表及其距离。商业上可用的LIDAR传感器764可以具有例如近似700m的广告范围,精度为2cm-3cm,支持700Mbps以太网连接。在一些示例中,可以使用一个或更多个非突出的LIDAR传感器764。在这样的示例中,LIDAR传感器764可以实现为可以嵌入到车辆700的前面、后面、侧面和/或拐角的小设备。在这样的示例中,LIDAR传感器764可以甚至对于低反射率对象提供高达120度水平的和35度竖直的视场,具有200m的范围。前面安装的LIDAR传感器764可以被配置用于45度与135度之间的水平视场。

[0198] 在一些示例中,也可以使用诸如3D闪光LIDAR之类的LIDAR技术。3D闪光LIDAR使用激光的闪光作为发射源,以照亮高达约200m的车辆周围环境。闪光LIDAR单元包括接受器,该接受器将激光脉冲传输时间和反射光记录在每个像素上,其进而与从车辆到对象的范围相对应。闪光LIDAR可以允许利用每个激光闪光生成周围环境的高度精确且无失真的图像。在一些示例中,可以部署四个闪光LIDAR传感器,车辆700的每一侧一个。可用的3D闪光LIDAR系统包括没有风扇以外的运动部件(moving part)的固态3D凝视阵列LIDAR相机(例如非浏览LIDAR设备)。闪光LIDAR设备可以使用每帧5纳秒I类(眼睛安全)激光脉冲,并且可以以3D范围点云和共同寄存的强度数据的形式捕获反射的激光。通过使用闪光LIDAR,并且因为闪光LIDAR是没有运动部件的固态设备,LIDAR传感器764可以不太容易受到运动模糊、振动和/或震动的影响。

[0199] 该车辆可以进一步包括IMU传感器766。在一些示例中,IMU传感器766可以位于车辆700的后轴的中心。IMU传感器766可以包括例如且不限于加速度计、磁力计、陀螺仪、磁罗盘和/或其他传感器类型。在一些示例中,例如在六轴应用中,IMU传感器766可以包括加速度计和陀螺仪,而在九轴应用中,IMU传感器766可以包括加速度计、陀螺仪和磁力计。

[0200] 在一些实施例中,IMU传感器766可以实现为微型高性能GPS辅助惯性导航系统(GPS/INS),其结合微机电系统(MEMS)惯性传感器、高灵敏度GPS接收器和高级卡尔曼滤波算法以提供位置、速度和姿态的估计。这样,在一些示例中,IMU传感器766可以使得车辆700能够在无需来自磁传感器的输入的情况下通过直接观察从GPS到IMU传感器766的速度变化并且将其相关来估计方向(heading)。在一些示例中,IMU传感器766和GNSS传感器758可以结合到单个集成单元中。

[0201] 该车辆可以包括置于车辆700中和/或车辆700周围的麦克风796。除别的以外,麦克风796可以用于应急车辆检测和识别。

[0202] 该车辆可以进一步包括任意数量的相机类型,包括立体相机768、广角相机770、红外相机772、环绕相机774、远程和/或中程相机798和/或其他相机类型。这些相机可以用来捕获车辆700整个外围周围的图像数据。使用的相机类型取决于实施例和车辆700的要求,并且相机类型的任意组合可以用来提供车辆700周围的必要覆盖。此外,相机的数量可以根据实施例而不同。例如,该车辆可以包括六个相机、七个相机、十个相机、十二个相机和/或

另一数量的相机。作为一个示例且非限制性地,这些相机可以支持千兆多媒体串行链路(GMSL)和/或千兆以太网。所述相机中的每一个在本文关于图7A和图7B更详细地进行了描述。

[0203] 车辆700可以进一步包括振动传感器742。振动传感器742可以测量车辆的诸如车轴之类的部件的振动。例如,振动的变化可以指示道路表面的变化。在另一个示例中,当使用两个或更多振动传感器742时,振动之间的差异可以用来确定道路表面的摩擦或滑移(例如当动力驱动轴与自由旋转轴之间存在振动差异时)。

[0204] 车辆700可以包括ADAS系统738。在一些示例中,ADAS系统738可以包括SoC。ADAS系统738可以包括自主/自适应/自动巡航控制(ACC)、协作自适应巡航控制(CACC)、前方撞车警告(FCW)、自动紧急制动(AEB)、车道偏离警告(LDW)、车道保持辅助(LKA)、盲点警告(BSW)、后方穿越交通警告(RCTW)、碰撞警告系统(CWS)、车道居中(LC)和/或其他特征和功能。

[0205] ACC系统可以使用RADAR传感器760、LIDAR传感器764和/或相机。ACC系统可以包括纵向ACC和/或横向ACC。纵向ACC监控并控制到紧接在车辆700前方的车辆的距离,并且自动地调节车速以维持离前方车辆的安全距离。横向ACC执行距离保持,并且在必要时建议车辆700改变车道。横向ACC与诸如LCA和CWS之类的其他ADAS应用程序有关。

[0206] CACC使用来自其他车辆的信息,该信息可以经由网络接口724和/或无线天线726经由无线链路或者通过网络连接(例如通过因特网)间接地从其他车辆接收。直接链接可以由车对车(V2V)通信链路提供,而间接链接可以是基础设施到车辆(I2V)的通信链路。通常,V2V通信概念提供关于紧接在前的车辆(例如紧接在车辆700前方且与其处于相同车道的车辆)的信息,而I2V通信概念提供关于前方更远处的交通的信息。CACC系统可以包括I2V和V2V信息源中的任一个或者二者。给定车辆700前方车辆的信息,CACC可以更加可靠,并且它有可能提高交通流的畅通性且降低道路拥堵。

[0207] FCW系统被设计为提醒驾驶员注意危险,使得驾驶员可以采取纠正措施。FCW系统使用耦合到专用处理器、DSP、FPGA和/或ASIC的前置相机和/或RADAR传感器760,该专用处理器、DSP、FPGA和/或ASIC电耦合至诸如显示器、扬声器和/或振动部件之类的驾驶员反馈。FCW系统可以提供例如声音、视觉警告、振动和/或快速制动脉冲形式的警告。

[0208] AEB系统检测即将发生的与另一车辆或其他对象的前方碰撞,并且可以在驾驶员在指定的时间或距离参数内没有采取纠正措施的情况下自动地应用制动器。AEB系统可以使用耦合到专用处理器、DSP、FPGA和/或ASIC的前置相机和/或RADAR传感器760。当AEB系统检测到危险时,它典型地首先提醒(alert)驾驶员采取纠正措施以避免碰撞,并且如果驾驶员没有采取纠正措施,那么AEB系统可以自动地应用制动器以努力防止或者至少减轻预测的碰撞的影响。AEB系统可以包括诸如动态制动支持和/或碰撞迫近制动之类的技术。

[0209] LDW系统提供了诸如方向盘或座位振动之类的视觉、听觉和/或触觉警告,以在车辆700穿过车道标记时提醒驾驶员。当驾驶员指示有意偏离车道时,通过激活转弯信号,不激活LDW系统。LDW系统可以使用耦合到专用处理器、DSP、FPGA和/或ASIC的前侧朝向相机,该专用处理器、DSP、FPGA和/或ASIC电耦合至诸如显示器、扬声器和/或振动部件之类的驾驶员反馈。

[0210] LKA系统是LDW系统的变型。如果车辆700开始离开车道,那么LKA系统提供纠正车

辆700的转向输入或制动。

[0211] BSW系统检测并向驾驶员警告汽车盲点中的车辆。BSW系统可以提供视觉、听觉和/或触觉警报以指示合并或改变车道是不安全的。系统可以在驾驶员使用转弯信号时提供附加的警告。BSW系统可以使用耦合到专用处理器、DSP、FPGA和/或ASIC的后侧朝向相机和/或RADAR传感器760,该专用处理器、DSP、FPGA和/或ASIC电耦合至诸如显示器、扬声器和/或振动部件之类的驾驶员反馈。

[0212] RCTW系统可以在车辆700倒车时在后置相机范围之外检测到对象时提供视觉、听觉和/或触觉通知。一些RCTW系统包括AEB以确保应用车辆制动器以避免撞车。RCTW系统可以使用耦合到专用处理器、DSP、FPGA和/或ASIC的一个或更多个后置RADAR传感器760,该专用处理器、DSP、FPGA和/或ASIC电耦合至诸如显示器、扬声器和/或振动部件之类的驾驶员反馈。

[0213] 常规的ADAS系统可能易于出现假阳性结果,这可能会让驾驶员烦恼并分散注意力,但是典型地不是灾难性的,因为ADAS系统提醒驾驶员并且允许驾驶员决定安全条件是否真正存在并且相对应地采取行动。然而,在自主车辆700中,在冲突结果的情况下,车辆700本身必须决定是否注意(heed)来自主计算机或者辅助计算机(例如第一控制器736或第二控制器736)的结果。例如,在一些实施例中,ADAS系统738可以是用于向备用计算机合理性模块提供感知信息的备用和/或辅助计算机。备用计算机合理性监视器可以在硬件部件上运行冗余多样的软件,以检测感知和动态驾驶任务中的故障。来自ADAS系统738的输出可以提供给监督MCU。如果来自主计算机和辅助计算机的输出冲突,那么监督MCU必须确定如何协调该冲突以确保安全操作。

[0214] 在一些示例中,主计算机可以被配置为向监督MCU提供置信度评分,指示主计算机对所选结果的置信度。如果置信度评分超过阈值,那么监督MCU可以遵循主计算机的方向,而不管辅助计算机是否提供冲突或不一致的结果。在置信度评分不满足阈值的情况下并且在主计算机和辅助计算机指示不同的结果(例如冲突)的情况下,监督MCU可以在这些计算机之间进行仲裁以确定适当的结果。

[0215] 监督MCU可以被配置为运行神经网络,所述神经网络被训练并且被配置为基于来自主计算机和辅助计算机的输出,确定辅助计算机提供假警报的条件。因此,监督MCU中的神经网络可以了解何时可以信任辅助计算机的输出以及何时不能。例如,当辅助计算机是基于RADAR的FCW系统时,监督MCU中的神经网络可以了解FCW系统何时正在识别事实上不是危险的金属对象,例如触发警报的排水栅格或井盖。类似地,当辅助计算机是基于相机的LDW系统时,监督MCU中的神经网络可以学习在骑车者或行人在场并且车道偏离实际上是最安全的策略时无视该LDW。在包括运行在监督MCU上的神经网络的实施例中,监督MCU可以包括适合于利用关联的内存运行神经网络的DLA或GPU中的至少一个。在优选的实施例中,监督MCU可以包括SoC 704的部件和/或作为SoC 704的部件而被包括。

[0216] 在其他示例中,ADAS系统738可以包括使用传统计算机视觉规则执行ADAS功能的辅助计算机。这样,辅助计算机可以使用经典的计算机视觉规则(如果-那么),并且在监督MCU中存在神经网络可以提高可靠性、安全性和性能。例如,多样化的实现方式和有意的非完全相同(non-identity)使得整个系统更加容错,对于软件(或者软件-硬件接口)功能造成的故障而言尤其如此。例如,如果在主计算机上运行的软件中存在软件漏洞或错误并且

运行在辅助计算机上的非完全相同的软件代码提供相同的总体结果,那么监督MCU可以更加确信总体结果是正确的,并且主计算机上的软件或硬件中的漏洞不造成实质性的错误。

[0217] 在一些示例中,ADAS系统738的输出可以馈送至主计算机的感知块和/或主计算机的动态驾驶任务块。例如,如果ADAS系统738由于对象紧接在前的原因而指示前方碰撞警告,那么感知块可以在识别对象时使用该信息。在其他示例中,辅助计算机可以具有它自己的神经网络,其被训练并且因此如本文所描述的降低假阳性的风险。

[0218] 车辆700可以进一步包括信息娱乐SoC 730(例如车载信息娱乐系统(IVI))。尽管被图示和描述为SoC,但是信息娱乐系统可以不是SoC,并且可以包括两个或更多分立的部件。信息娱乐SoC 730可以包括可以用来向车辆700提供音频(例如音乐、个人数字助理、导航指令、新闻、广播等等)、视频(例如TV、电影、流媒体等等)、电话(例如免提呼叫)、网络连接(例如LTE、Wi-Fi等等)和/或信息服务(例如导航系统,后停车援助,无线电数据系统,诸如燃油水平、覆盖的总距离、制动燃油水平、油位、车门开/关、空气过滤器信息之类的车辆有关信息,等等)的硬件和软件的组合。例如,信息娱乐SoC 730可以包括收音机、盘播放器、导航系统、视频播放器、USB和蓝牙连接、车载电脑、车载娱乐、Wi-Fi、方向盘音频控件、免提语音控件、平视显示器(HUD)、HMI显示器734、远程信息处理设备、控制面板(例如用于控制各种部件、特征和/或系统,和/或与其交互)和/或其他部件。信息娱乐SoC 730可以进一步用来向车辆的用户提供信息(例如视觉的和/或听觉的),例如来自ADAS系统738的信息,诸如规划的车辆机动、轨迹、周围环境信息(例如交叉路口信息、车辆信息、道路信息等等)之类的自动驾驶信息,和/或其他信息。

[0219] 信息娱乐SoC 730可以包括GPU功能。信息娱乐SoC 730可以通过总线702(例如CAN总线、以太网等)与车辆700的其他设备、系统和/或部件通信。在一些示例中,信息娱乐SoC 730可以耦合至监督MCU,使得在主控制器736(例如车辆700的主和/或备用计算机)出现故障的情况下,信息娱乐系统的GPU可以执行一些自动驾驶功能。在这样的示例中,信息娱乐SoC 730可以如本文所描述的将车辆700置于司机安全停车模式。

[0220] 车辆700可以进一步包括仪表组732(例如数字仪表盘、电子仪表组、数字仪表表面板等等)。仪表组732可以包括控制器和/或超级计算机(例如分立的控制器或超级计算机)。仪表组732可以包括一套仪器,例如车速表、燃油水平、油压、转速表、里程表、转弯指示器、换挡位置指示器、安全带警告灯、停车制动警告灯、发动机故障灯、安全气囊(SRS)系统信息、照明控件、安全系统控件、导航信息等等。在一些示例中,信息可以被显示和/或在信息娱乐SoC 730和仪表组732之间共享。换言之,仪表组732可以作为信息娱乐SoC 730的部分而被包括,或者反之亦然。

[0221] 图7D为根据本公开一些实施例的基于云的服务器与图7A的示例自主车辆700之间的通信的系统示意图。系统776可以包括服务器778、网络790以及包括车辆700在内的车辆。服务器778可以包括多个GPU 784(A)-784(H)(这里统称为GPU 784)、PCIe交换机782(A)-782(H)(这里统称为PCIe交换机782)和/或CPU 780(A)-780(B)(这里统称为CPU 780)。GPU 784、CPU 780和PCIe交换机可以与诸如例如但不限于NVIDIA开发的NVLink接口788之类的高速互连和/或PCIe连接786互连。在一些示例中,GPU 784经由NVLink和/或NVSwitch SoC连接,并且GPU 784和PCIe交换机782经由PCIe互连连接。尽管图示出八个GPU 784、两个CPU 780和两个PCIe交换机,但是这并不意图是限制性的。取决于实施例,服务器778中的每一个

可以包括任意数量的GPU 784、CPU780和/或PCIe交换机。例如，服务器778中的每一个可以包括八个、十六个、三十二个和/或更多GPU 784。

[0222] 服务器778可以通过网络790并且从车辆接收图像数据，该图像数据表示出诸如最近开始的道路工程之类的意外或改变的道路状况的图像。服务器778可以通过网络790并且向车辆传输神经网络792、更新的神经网络792和/或地图信息794，包括关于交通和道路状况的信息。对地图信息794的更新可以包括对于HD地图722的更新，例如关于建筑工地、坑洼、弯道、洪水或其他障碍物的信息。在一些示例中，神经网络792、更新的神经网络792和/或地图信息794可以已从新的训练和/或从环境中的任意数量的车辆接收的数据中表示和/或基于数据中心处执行的训练（例如使用服务器778和/或其他服务器）的经验产生。

[0223] 服务器778可以用来基于训练数据训练机器学习模型（例如神经网络）。训练数据可以由车辆生成，和/或可以在仿真中生成（例如使用游戏引擎）。在一些示例中，训练数据被标记（例如在神经网络受益于有监督学习的情况下）和/或经历其他预处理，而在其他示例中，训练数据不被标记和/或预处理（例如在神经网络无需有监督学习的情况下）。训练可以根据任何一类或更多类机器学习技术来执行，包括但不限于以下类：监督训练、半监督训练、无监督训练、自学习、强化学习、联合学习、迁移学习、特征学习（包括主成分和聚类分析）、多线性子空间学习、流形学习、表示学习（包括备用字典学习）、基于规则的机器学习、异常检测及其任何变体或组合。一旦机器学习模型被训练，机器学习模型可以由车辆使用（例如通过网络790传输至车辆），和/或机器学习模型可以由服务器778用来远程地监控车辆。

[0224] 在一些示例中，服务器778可以接收来自车辆的数据，并且将该数据应用到最新的实时神经网络以用于实时智能推理。服务器778可以包括由GPU 784供电的深度学习超级计算机和/或专用AI计算机，例如NVIDIA开发的DGX和DGX站机器。然而，在一些示例中，服务器778可以包括仅使用CPU供电的数据中心的深度学习基础设施。

[0225] 服务器778的深度学习基础设施能够快速实时推理，并且可以使用该能力来评估并验证车辆700中的处理器、软件和/或关联硬件的健康状况。例如，深度学习基础设施可以接收来自车辆700的定期更新，例如图像序列和/或车辆700已经定位的位于该图像序列中的对象（例如经由计算机视觉和/或其他机器学习对象分类技术）。深度学习基础设施可以运行它自己的神经网络以识别对象并且将它们与车辆700识别的对象相比较，如果结果不匹配并且该基础设施得出车辆700中的AI发生故障的结论，那么服务器778可以向车辆700传输信号，指示车辆700的故障保护计算机进行控制，通知乘客，并且完成安全停车操作。

[0226] 为了推理，服务器778可以包括GPU 784和一个或多个可编程推理加速器（例如NVIDIA的TensorRT 3）。GPU供电的服务器和推理加速的组合可以使得实时响应成为可能。在其他示例中，例如在性能不那么重要的情况下，CPU、FPGA和其他处理器供电的服务器可以用于推理。

[0227] 示例计算设备

[0228] 图8为适合用于实现本公开一些实施例的示例计算设备800的框图。计算设备800可以包括直接或间接耦合下列设备的互连系统802：内存804，一个或多个中央处理单元（CPU）806，一个或多个图形处理单元（GPU）808，通信接口810，输入/输出（I/O）端口812，

输入/输出组件814,电源816,一个或更多个呈现组件818(例如显示器)和一个或更多个逻辑单元820。在至少一个实施例中,计算设备800可以包括一个或更多个虚拟机(VM),和/或其任何组件可以包括虚拟组件(例如,虚拟硬件组件)。对于非限制性示例,一个或更多个GPU 808可以包括一个或更多个vGPU,一个或更多个CPU 806可以包括一个或更多个vCPU,和/或一个或更多个逻辑单元820可以包括一个或更多个虚拟逻辑单元。因此,计算设备800可以包括分立组件(例如,专用于计算设备800的完整GPU)、虚拟组件(例如,专用于计算设备800的GPU的一部分),或其组合。

[0229] 尽管图8的各个框被示为经由具有线路的互连系统802连接,但是这并不意图是限制性的,并且仅仅为了清楚起见。例如,在一些实施例中,诸如显示设备之类的呈现组件818可以被认为是I/O组件814(例如如果显示器为触摸屏)。作为另一个示例,CPU 806和/或GPU 808可以包括内存(例如,内存804可以表示除了GPU 808、CPU 806和/或其他组件的内存以外的存储设备)。换言之,图8的计算设备仅仅是说明性的。在诸如“工作站”、“服务器”、“膝上型电脑”、“台式机”、“平板电脑”、“客户端设备”、“移动设备”、“手持式设备”、“游戏控制台”、“电子控制单元(ECU)”、“虚拟现实系统”和/或其他设备或系统类型之类的类别之间不进行区分,因为所有这些都考虑在图8的计算设备的范围内。

[0230] 互连系统802可以表示一条或更多条链路或总线,例如地址总线、数据总线、控制总线或者其组合。互连系统802可以包括一种或更多种链路或总线类型,例如行业标准架构(ISA)总线、扩展行业标准架构(EISA)总线、视频电子标准协会(VESA)总线、外围组件互连(PCI)总线、外围组件互连快速(PCIe)总线,和/或另一种类型的总线或链路。在一些实施例中,组件之间存在直接连接。作为示例,CPU 806可以直接连接到存储器804。此外,CPU 806可以直接连接到GPU 808。在组件之间存在直接或点对点连接的情况下,互连系统802可以包括PCIe链路来执行该连接。在这些示例中,计算设备800中不需要包括PCI总线。

[0231] 内存804可以包括各种各样的计算机可读介质中的任何介质。计算机可读介质可以是可由计算设备800访问的任何可用介质。计算机可读介质可以包括易失性和非易失性介质以及可移除和不可移除介质。举例而言且非限制性地,计算机可读介质可以包括计算机存储介质和通信介质。

[0232] 计算机存储介质可以包括易失性和非易失性介质和/或可移除和不可移除介质,其以用于存储诸如计算机可读指令、数据结构、程序模块和/或其他数据类型之类的信息的任何方法或技术实现。例如,内存804可以存储计算机可读指令(例如其表示程序和/或程序元素,例如操作系统)。计算机存储介质可以包括但不限于RAM、ROM、EEPROM、闪存或者其他存储技术、CD-ROM、数字多功能盘(DVD)或其他光盘存储装置、磁带盒、磁带、磁盘存储装置或其他磁存储设备,或者可以用来存储预期的信息且可以由计算设备800访问的任何其他介质。当在本文使用时,计算机存储介质并不包括信号本身。

[0233] 计算机存储介质可以在诸如载波之类的调制数据信号或其他传输机制中包含计算机可读指令、数据结构、程序模块和/或其他数据类型,并且包括任何信息输送介质。术语“调制数据信号”可以指这样的信号,该信号使它的特性中的一个或更多个以这样的将信息编码到该信号中的方式设置或改变。举例而言且非限制性地,计算机存储介质可以包括诸如有线网络或直接有线连接之类的有线介质,以及诸如声音、RF、红外和其他无线介质之类的无线介质。任何以上所述的组合也应当包含在计算机可读介质的范围内。

[0234] CPU 806可以被配置为执行计算机可读指令中的至少一些,以便控制计算设备800的一个或更多个组件执行本文描述的方法和/或过程中的一个或更多个。CPU 806中的每一个可以包括能够同时处理大量软件线程的一个或更多个核(例如一个、两个、四个、八个、二十八个、七十二个等等)。CPU 806可以包括任何类型的处理器,并且可以包括不同类型的处理器,这取决于实现的计算设备800的类型(例如具有用于移动设备的较少核的处理器以及具有用于服务器的更多核的处理器)。例如,取决于计算设备800的类型,处理器可以是使用精简指令集计算(RISC)实现的高级RISC机制(ARM)处理器或者使用复杂指令集计算(CISC)实现的x86处理器。除了一个或更多个微处理器或者诸如数学协处理器之类的补充协处理器之外,计算设备800还可以包括一个或更多个CPU 806。

[0235] 除了或替代CPU 806,GPU 808还可以被配置为执行至少一些计算机可读指令,以控制计算设备800的一个或更多个组件执行一个或更多个本文所述的方法和/或过程。一个或更多个GPU 808可以是集成GPU(例如,具有一个或更多个CPU 806)和/或一个或更多个GPU 808可以是离散GPU。在实施例中,一个或更多个GPU 808可以是一个或更多个CPU 806的协处理器。计算设备800可以使用GPU 808来渲染图形(例如,3D图形)或执行通用计算。例如,GPU 808可用于GPU上的通用计算(GPGPU)。GPU808可以包括能够同时处理数百或数千个软件线程的数百或数千个核。GPU 808可以响应于渲染命令(例如经由主机接口接收的来自CPU 806的渲染命令)而生成用于输出图像的像素数据。GPU 808可以包括诸如显示内存之类的用于存储像素数据或任何其他合适的数据(例如GPGPU数据)的图形内存。显示内存可以作为内存804的部分而被包括。GPU 808可以包括(例如经由链路)并行操作的两个或更多GPU。链路可以直接连接GPU(例如,使用NVLINK),也可以通过交换机(例如,使用NVSwitch)连接GPU。当组合在一起时,每个GPU 808可以生成用于输出的不同部分或者用于不同输出的像素数据或GPGPU数据(例如,第一GPU用于第一图像,第二GPU用于第二图像)。每个GPU可以包括它自己的内存,或者可以与其他GPU共享内存。

[0236] 除了或替代CPU 806和/或GPU 808,逻辑单元820可以被配置为执行至少一些计算机可读指令,以控制计算设备800的一个或更多个组件来执行本文描述的一种或更多种方法和/或过程。在实施例中,CPU 806、GPU808和/或逻辑单元820可以离散地或联合地执行方法、过程和/或其部分的任何组合。一个或更多个逻辑单元820可以是一个或更多个CPU 806和/或一个或更多个GPU 808的一部分和/或集成在其中,和/或一个或更多个逻辑单元820可以是CPU 806和/或GPU 808的离散组件或以其他方式在其外部。在实施例中,一个或更多个逻辑单元820可以是一个或更多个CPU806和/或一个或更多个GPU 808的处理器。

[0237] 逻辑单元820的示例包括一个或更多个处理核心和/或其组件,例如数据处理单元(DPU)、张量核心(TC)、张量处理单元(TPU)、像素视觉核心(PVC)、视觉处理单元(VPU)、图形处理集群(GPC)、纹理处理集群(TPC)、流式多处理器(SM)、树遍历单元(TTU)、人工智能加速器(AIA)、深度学习加速器(DLA)、算术逻辑单元(ALU)、专用集成电路(ASIC)、浮点单元(FPU)、输入/输出(I/O)元件、外围组件互连(PCI)或外围组件互连快速(PCIe)元件等。

[0238] 通信接口810可以包括一个或更多个接收器、发送器和/或收发器,其使得计算设备800能够经由电子通信网络与其他计算设备通信,包括有线和/或无线通信。通信接口810可以包括使能通过若干不同网络中的任何网络进行通信的组件和功能,所述网络例如无线网络(例如Wi-Fi、Z波、蓝牙、蓝牙LE、ZigBee等等)、有线网络(例如通过以太网或

InfiniBand通信)、低功率广域网(例如LoRaWAN、SigFox等等)和/或因特网。在一个或多个实施例中,逻辑单元820和/或通信接口810可以包括一个或多个数据处理单元(DPU),以通过网络和/或通过互连系统802接收的数据直接传输到一个或多个GPU 808(例如, GPU 808中的存储器)。

[0239] I/O端口812可以使得计算设备800能够逻辑地耦合到包括I/O组件814、呈现组件818和/或其他组件在内的其他设备,其中一些可以内置到(例如集成到)计算设备800中。说明性I/O组件814包括麦克风、鼠标、键盘、操纵杆、游戏垫、游戏控制器、碟形卫星天线、浏览仪、打印机、无线设备等等。I/O组件814可以提供处理用户生成的空中手势、语音或其他生理输入的自然用户接口(NUI)。在一些实例中,输入可以传输至适当的网络元件以便进一步处理。NUI可以实现语音识别、手写笔识别、面部识别、生物特征识别、屏幕上和邻近屏幕的手势识别、空中手势、头部和眼睛跟踪以及与计算设备800的显示器关联的触摸识别(如下文更详细地描述的)的任意组合。计算设备800可以包括诸如立体相机系统之类的深度相机、红外相机系统、RGB相机系统、触摸屏技术以及这些的组合,以用于手势检测和识别。此外,计算设备800可以包括使能运动检测的加速度计或陀螺仪(例如作为惯性测量单元(IMU)的部分)。在一些示例中,加速度计或陀螺仪的输出可以由计算设备800用来渲染沉浸式增强现实或者虚拟现实。

[0240] 电源816可以包括硬接线电源、电池电源或者其组合。电源816可以向计算设备800供电以使得计算设备800的组件能够操作。

[0241] 呈现组件818可以包括显示器(例如监视器、触摸屏、电视屏幕、平视显示器(HUD)、其他显示器类型或者其组合)、扬声器和/或其他呈现组件。呈现组件818可以接收来自其他组件(例如GPU 808、CPU 806、DPU等等)的数据,并且输出该数据(例如作为图像、视频、声音等等)。

[0242] 示例数据中心

[0243] 图9示出了示例数据中心900,其可用于本公开的至少一个实施例中。数据中心900可以包括数据中心基础设施层910、框架层920、软件层930和应用层940。

[0244] 如图9所示,数据中心基础设施层910可以包括资源协调器912、分组的计算资源914和节点计算资源(“节点C.R.”)916(1)-916(N),其中“N”代表任何完整的正整数。在至少一个实施例中,节点C.R.916(1)-916(N)可以包括但不限于任何数量的中央处理单元(CPU)或其他处理器(包括DPU、加速器、现场可编程门阵列(FPGA)、图形处理器或图形处理单元(GPU)等),内存设备(例如动态只读内存),存储设备(例如固态硬盘或磁盘驱动器),网络输入/输出(NW I/O)设备,网络交换机,虚拟机(VM),电源模块和冷却模块等。在一些实施例中,节点C.R.916(1)-916(N)中的一个或多个节点C.R.可以对应于具有一个或多个上述计算资源的服务器。此外,在一些实施例中,节点C.R.916(1)-916(N)可以包括一个或多个虚拟组件,例如vGPU、vCPU等,和/或节点C.R.916(1)-916(N)中的一个或多个可以对应于虚拟机(VM)。

[0245] 在至少一个实施例中,分组的计算资源914可以包括容纳在一个或多个机架内的节点C.R.916的单独分组(未示出),或者容纳在各个地理位置的数据中心内的许多机架(也未示出)。分组的计算资源914内的节点C.R.916的单独分组可以包括可以被配置或分配为支持一个或多个工作负载的分组的计算、网络、内存或存储资源。在至少一个实施例

中,可以将包括CPU、GPU、DPU和/或其他处理器的几个节点C.R.916分组在一个或更多个机架内,以提供计算资源来支持一个或更多个工作负载。一个或更多个机架还可以包括以任意组合的任何数量的电源模块、冷却模块和/或网络交换机。

[0246] 资源协调器912可以配置或以其他方式控制一个或更多个节点C.R.916(1)-916(N)和/或分组的计算资源914。在至少一个实施例中,资源协调器912可以包括用于数据中心900的软件设计基础结构(SDI)管理实体。资源协调器912可以包括硬件、软件或其某种组合。

[0247] 在至少一个实施例中,如图9所示,框架层920可以包括作业调度器933、配置管理器934、资源管理器936和分布式文件系统938。框架层920可以包括支持软件层930的软件932和/或应用程序层940的一个或更多个应用程序942的框架。软件932或应用程序942可以分别包括基于Web的服务软件或应用程序,例如由Amazon Web Services,Google Cloud和Microsoft Azure提供的服务软件或应用程序。框架层920可以是但不限于一种免费且开源的软件网络应用框架,例如可以利用分布式文件系统938来进行大规模数据处理(例如“大数据”)的Apache Spark™(以下称为“Spark”)。在至少一个实施例中,作业调度器933可以包括Spark驱动器,用于促进对数据中心900的各个层所支持的工作负载进行调度。在至少一个实施例中,配置管理器934可以能够配置不同的层,例如软件层930和包括Spark和用于支持大规模数据处理的分布式文件系统938的框架层920。资源管理器936能够管理映射到或分配用于支持分布式文件系统938和作业调度器933的集群或分组计算资源。在至少一个实施例中,集群或分组计算资源可以包括数据中心基础设施层910处的分组的计算资源914。资源管理器936可以与资源协调器912协调以管理这些映射的或分配的计算资源。

[0248] 在至少一个实施例中,包括在软件层930中的软件932可以包括由节点C.R.916(1)-916(N)的至少部分,分组的计算资源914和/或框架层920的分布式文件系统938使用的软件。一种或更多种类型的软件可以包括但不限于Internet网页搜索软件、电子邮件病毒浏览软件、数据库软件和流视频内容软件。

[0249] 在至少一个实施例中,应用层940中包括的一个或更多个应用程序942可以包括由节点C.R.916(1)-916(N)的至少部分、分组的计算资源914和/或框架层920的分布式文件系统938使用的一种或更多种类型的应用程序。一种或更多种类型的应用程序可以包括但不限于任何数量的基因组学应用程序,认知计算和机器学习应用程序,包括训练或推理软件,机器学习框架软件(例如PyTorch、TensorFlow、Caffe等)和/或其他与一个或更多个实施例结合使用的机器学习应用程序。

[0250] 在至少一个实施例中,配置管理器934、资源管理器936和资源协调器912中的任何一个可以基于以任何技术上可行的方式获取的任何数量和类型的数据来实现任何数量和类型的自我修改动作。自我修改动作可以减轻数据中心900的数据中心操作员做出可能不好的配置决定并且可以避免数据中心的未充分利用和/或行差的部分。

[0251] 数据中心900可以包括工具、服务、软件或其他资源,用于根据本文所述的一个或更多个实施例来训练一个或更多个机器学习模型或者使用一个或更多个机器学习模型来预测或推理信息。例如,可以通过使用上文关于数据中心900描述的软件和计算资源,根据神经网络架构计算权重参数来训练机器学习模型。在至少一个实施例中,通过使用通过一种或更多种训练技术计算出的权重参数,可以使用上面与关于数据中心900所描述的资源,

使用对应于一个或更多个神经网络的经训练的机器学习模型来推理或预测信息,例如但不限于本文所述的那些。

[0252] 在至少一个实施例中,数据中心900可以使用CPU、专用集成电路(ASIC)、GPU、FPGA和/或其他硬件(或与之对应的虚拟计算资源)来使用上述资源来执行训练和/或推理。此外,上述的一个或更多个软件和/或硬件资源可以配置成一种服务,以允许用户训练或执行信息推理,例如图像识别、语音识别或其他人工智能服务。

[0253] 示例网络环境

[0254] 适用于实现本公开的实施例的网络环境可以包括一个或更多个客户端设备、服务器、网络附加存储(NAS)、其他后端设备和/或其他设备类型。客户端设备、服务器和/或其他设备类型(例如,每个设备)可以在图8的计算设备800的一个或更多个实例上实现—例如,每个设备可以包括计算设备800的类似组件、特征和/或功能。此外,在实现后端设备(例如,服务器、NAS等)的情况下,后端设备可以是作为数据中心900的一部分而被包括的,其示例在此关于图9更详细地描述。

[0255] 网络环境的组件可以通过网络彼此通信,网络可以是有线、无线或两者。网络可以包括多个网络,或者多个网络中的网络。举例来说,网络可以包括一个或更多个广域网(WAN)、一个或更多个局域网(LAN)、一个或更多个公共网络(例如因特网和/或公共交换电话网(PSTN))、和/或一个或更多个专用网络。在网络包括无线电信网的情况下,诸如基站、通信塔或甚至接入点(以及其他组件)之类的组件可以提供无线连接。

[0256] 兼容的网络环境可以包括一个或更多个对等网络环境(在这种情况下服务器可能不包括在网络环境中),以及一个或更多个客户端-服务器网络环境(在这种情况下一个或更多个服务器可以包含在网络环境中)。在对等网络环境中,本文描述的关于服务器的功能可以在任意数量的客户端设备上实现。

[0257] 在至少一个实施例中,网络环境可以包括一个或更多个基于云的网络环境、分布式计算环境、它们的组合等。基于云的网络环境可以包括框架层、作业调度器、资源管理器,以及在一个或更多个服务器上实现的分布式文件系统,这些服务器可以包括一个或更多个核心网络服务器和/或边缘服务器。框架层可以包括用于支持软件层的软件和/或应用层的一个或更多个应用程序的框架。软件或应用程序可分别包括基于网络的服务软件或应用程序。在实施例中,一个或更多个客户端设备可以使用基于网络的服务软件或应用程序(例如,通过经由一个或更多个应用程序编程接口(API)访问服务软件和/或应用程序)。框架层可以是但不限于一种类型的免费且开源软件网络应用程序框架,例如可以使用分布式文件系统进行大规模数据处理(例如,“大数据”)。

[0258] 基于云的网络环境可以提供执行本文描述的计算和/或数据存储功能(或其一个或更多个部分)的任何组合的云计算和/或云存储。这些各种功能中的任何一个都可以分布在来自(例如,可以分布在州、地区、国家、全球等地的一个或更多个数据中心的)中央或核心服务器的多个位置。如果到用户(例如,客户端设备)的连接相对靠近边缘服务器,则核心服务器可以将功能的至少一部分指定给边缘服务器。基于云的网络环境可以是私有的(例如,仅限于单个组织),可以是公共的(例如,对许多组织可用),和/或其组合(例如,混合云环境)。

[0259] 客户端设备可以包括本文关于图8描述的示例计算设备800的至少一些组件、特征

和功能。作为示例而非限制,客户端设备可以体现为个人计算机(PC)、膝上型计算机、移动设备、智能手机、平板计算机、智能手表、可穿戴计算机、个人数码助理(PDA)、MP3播放器、虚拟现实头戴式显示器、全球定位系统(GPS)或设备、视频播放器、摄像机、监视设备或系统、车辆、船只、飞行器、虚拟机器、无人机、机器人、手持通信设备、医院设备、游戏设备或系统、娱乐系统、车载计算机系统、嵌入式系统控制器、遥控器、电器、消费电子设备、工作站、边缘设备、这些描述设备的任意组合或任何其他合适的设备。

[0260] 本公开可以在由计算机或者诸如个人数字助理或其他手持式设备之类的其他机器执行的、包括诸如程序模块之类的计算机可执行指令的机器可使用指令或者计算机代码的一般背景下进行描述。通常,包括例程、程序、对象、组件、数据结构等等的程序模块指的是执行特定任务或者实现特定抽象数据类型的代码。本公开可以在各种各样的系统配置中实践,这些配置包括手持式设备、消费电子器件、通用计算机、更专业的计算设备等等。本公开也可以在其中任务由通过通信网络链接的远程处理设备执行的分布式计算环境中实践。

[0261] 如在本文中使用的,“和/或”关于两个或更多元素的叙述应当解释为仅指一个元素或者元素组合。例如,“元素A、元素B和/或元素C”可以包括仅仅元素A,仅仅元素B,仅仅元素C,元素A和元素B,元素A和元素C,元素B和元素C,或者元素A、B和C。此外,“元素A或元素B中的至少一个”可以包括元素A中的至少一个,元素B中的至少一个,或者元素A中的至少一个和元素B中的至少一个。进一步,“元素A和元素B中的至少一个”可以包括元素A中的至少一个,元素B中的至少一个,或者元素A中的至少一个和元素B中的至少一个。

[0262] 这里详细地描述了本公开的主题以满足法定要求。然而,描述本身并非意在限制本公开的范围。相反地,本发明人已经设想到,要求保护的主题也可以以其他的方式具体化,以包括与本文中结合其他当前或未来技术描述的步骤不同的步骤或者相似的步骤的组合。而且,尽管术语“步骤”和/或“框”在本文中可以用来隐含采用的方法的不同元素,但是这些术语不应当被解释为暗示本文公开的各个步骤之中或之间的任何特定顺序,除非明确描述了各步骤的顺序。

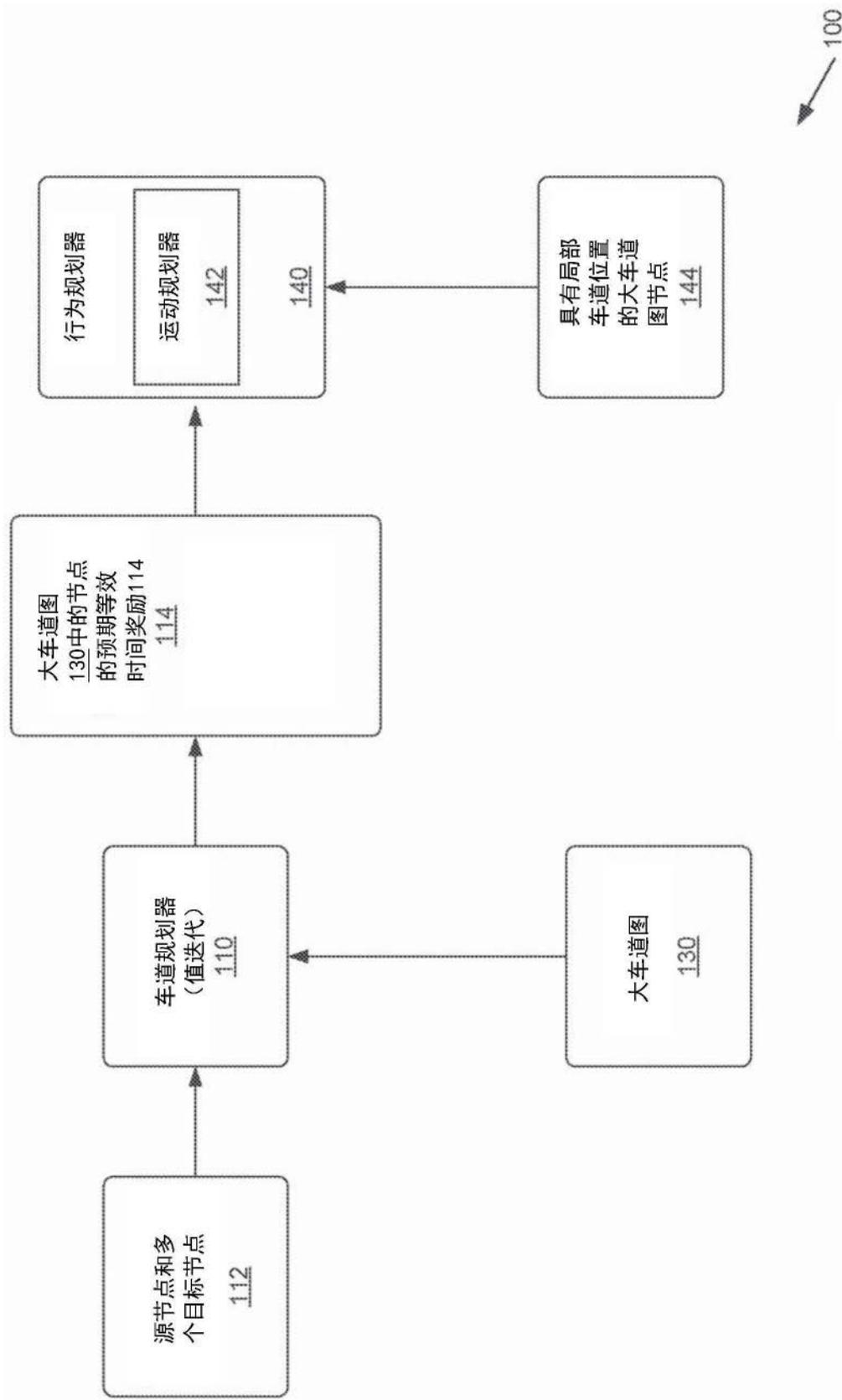


图1A

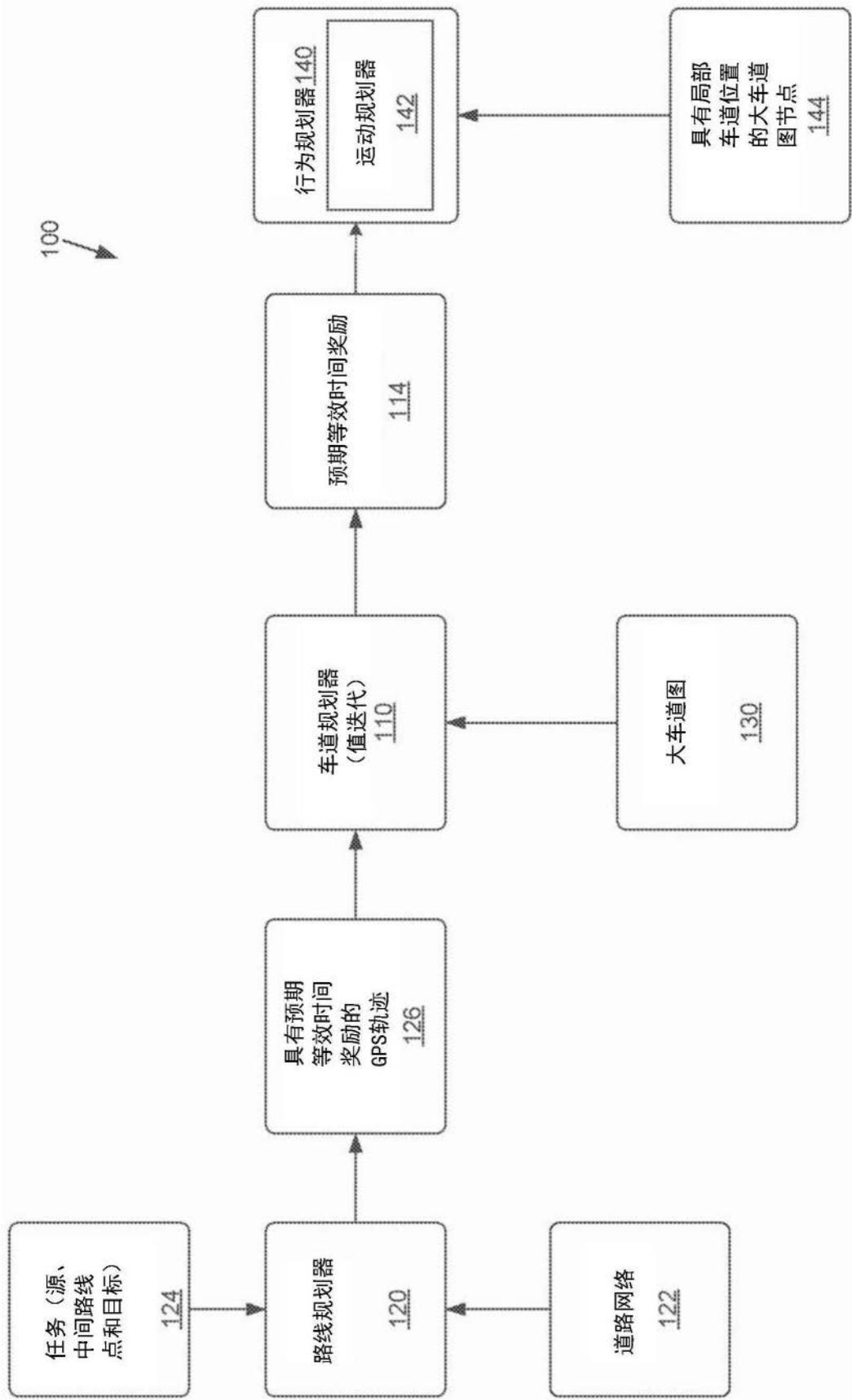


图1B

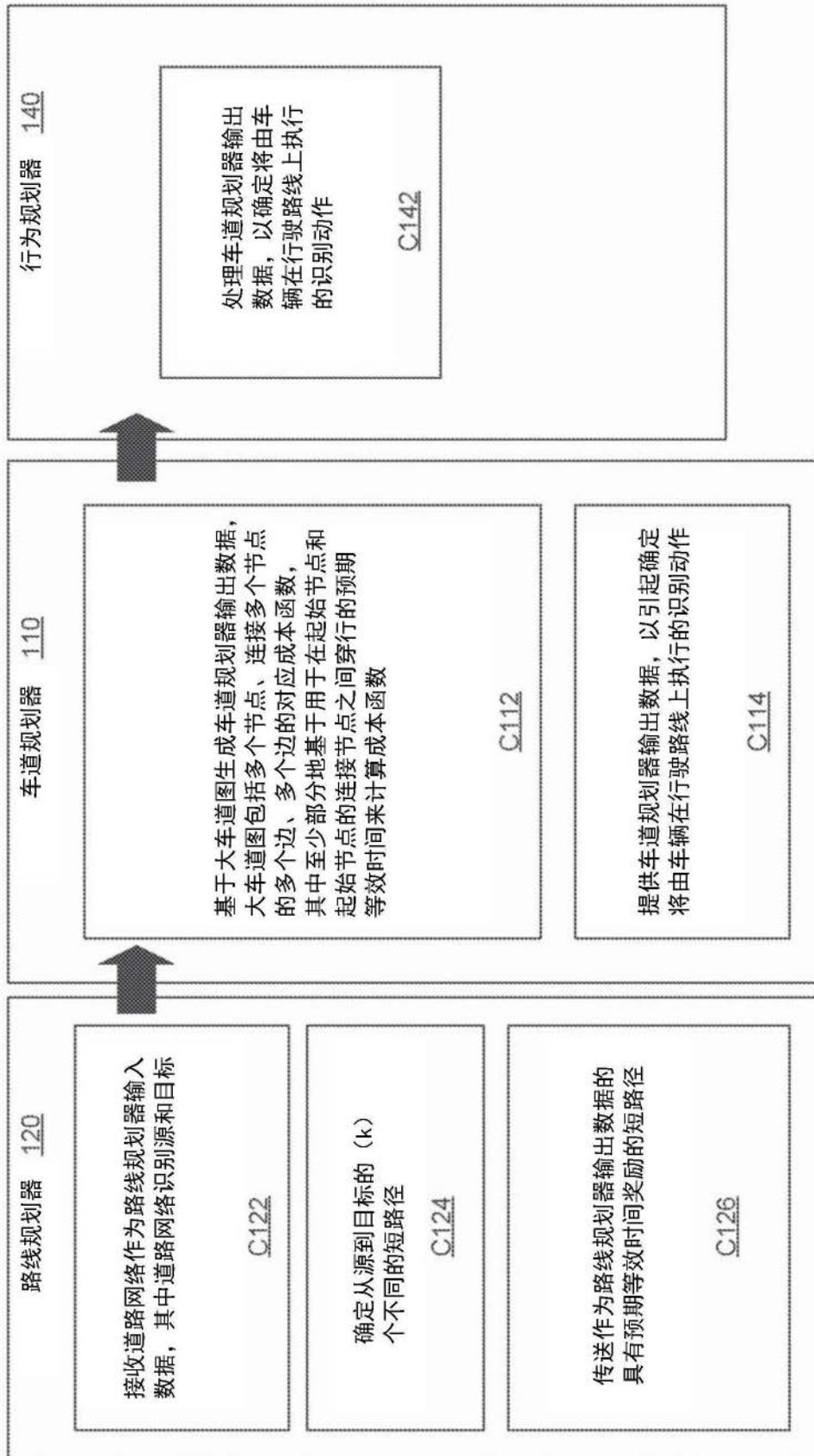


图1C

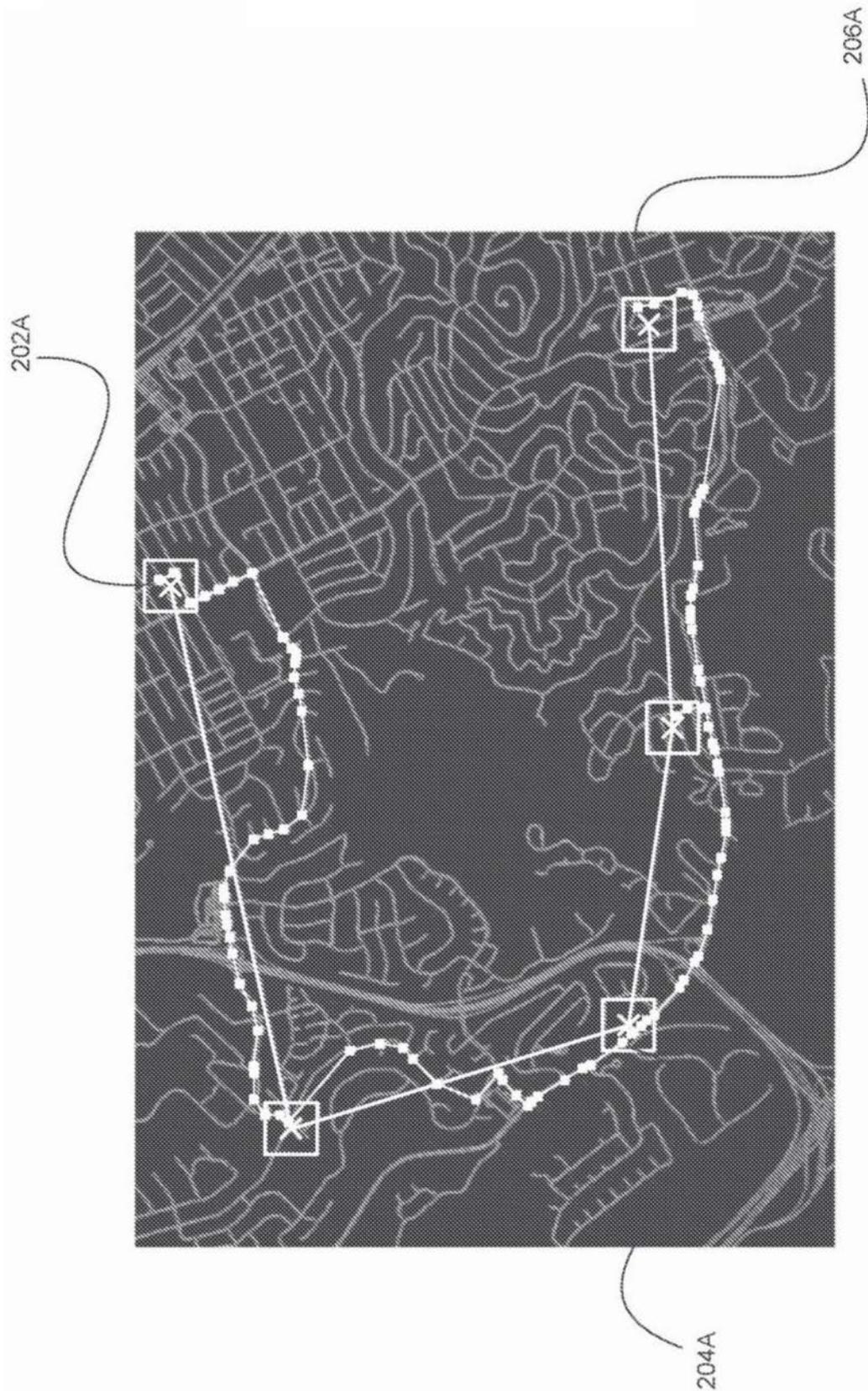


图2A

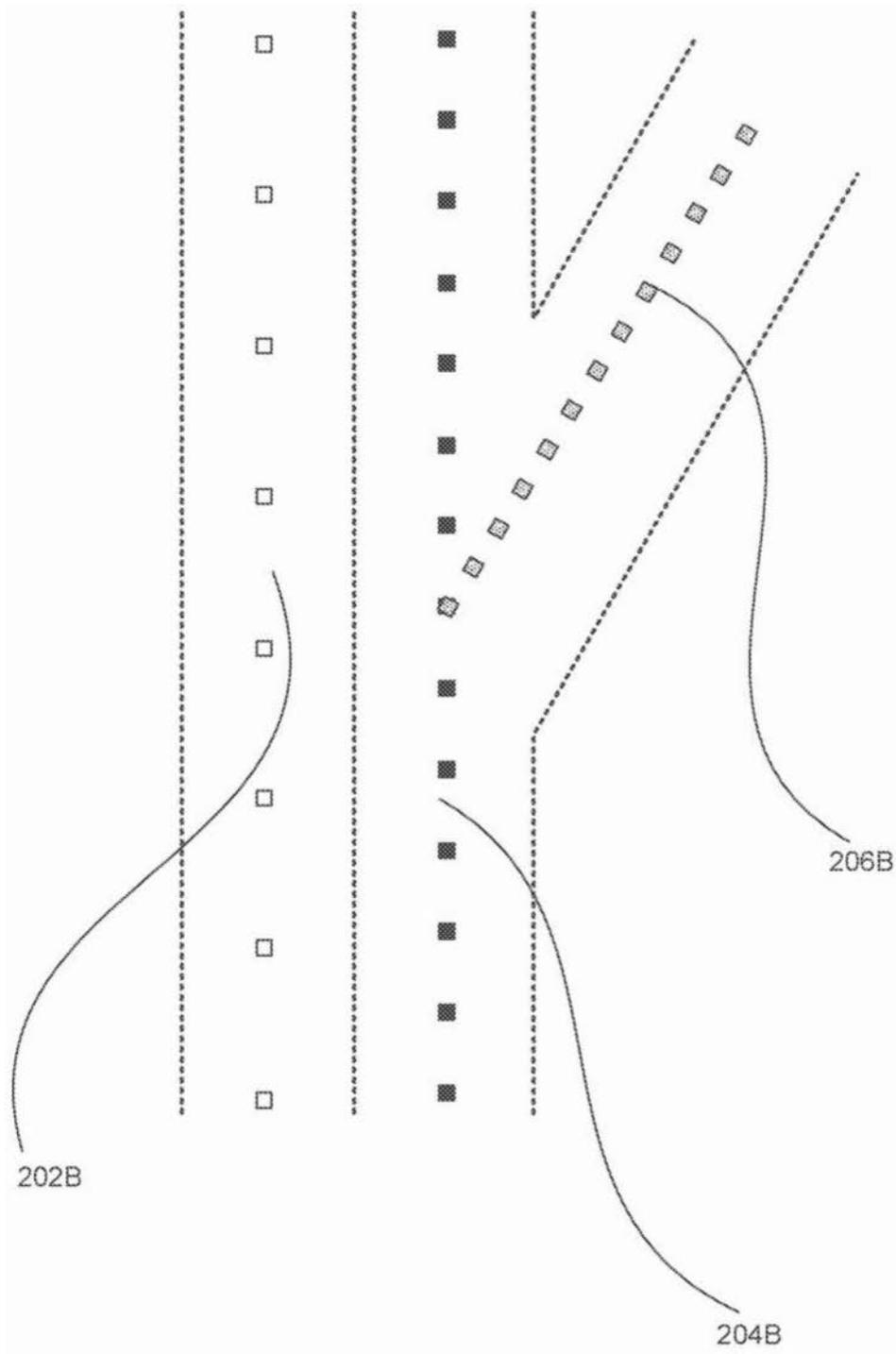


图2B

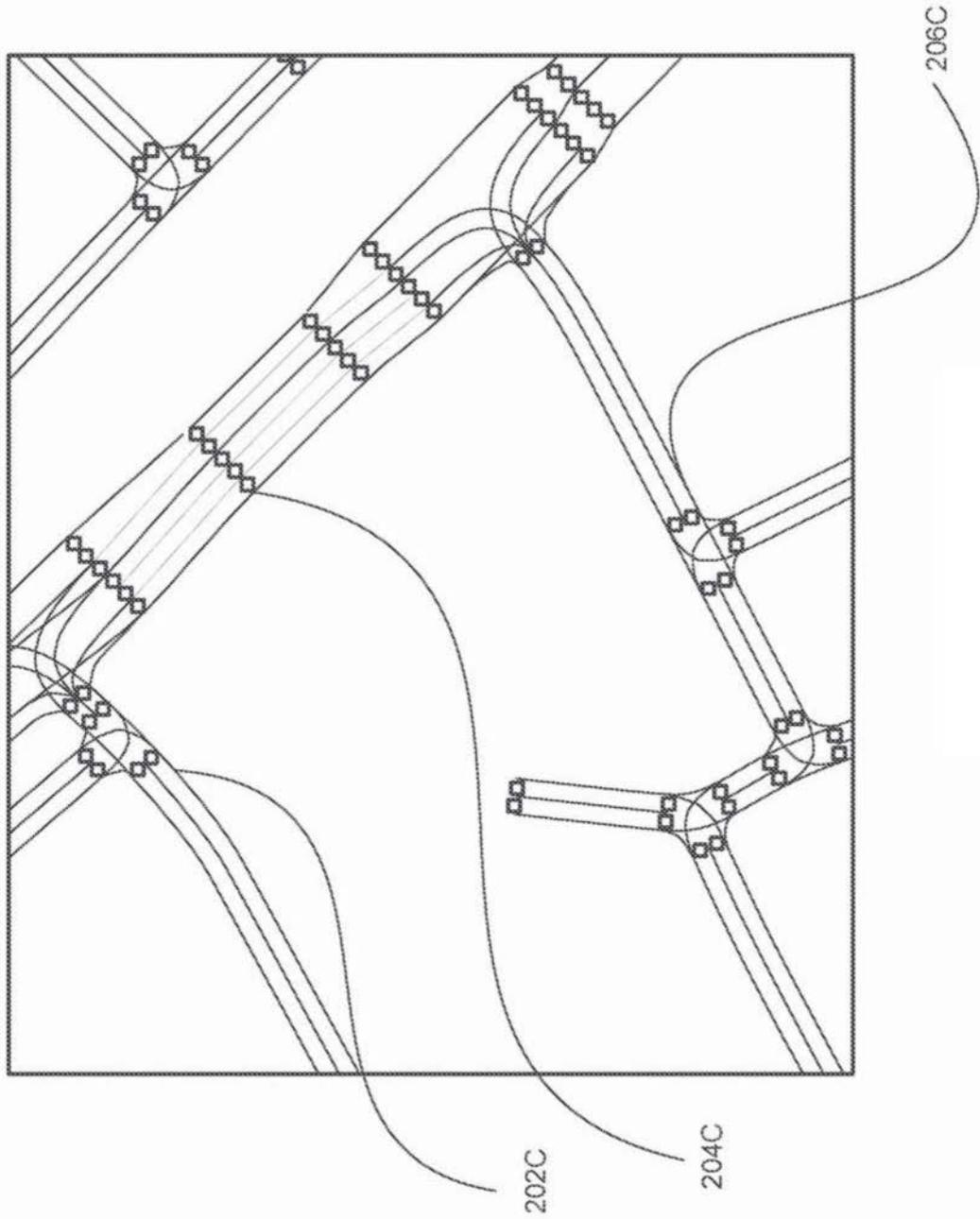


图2C

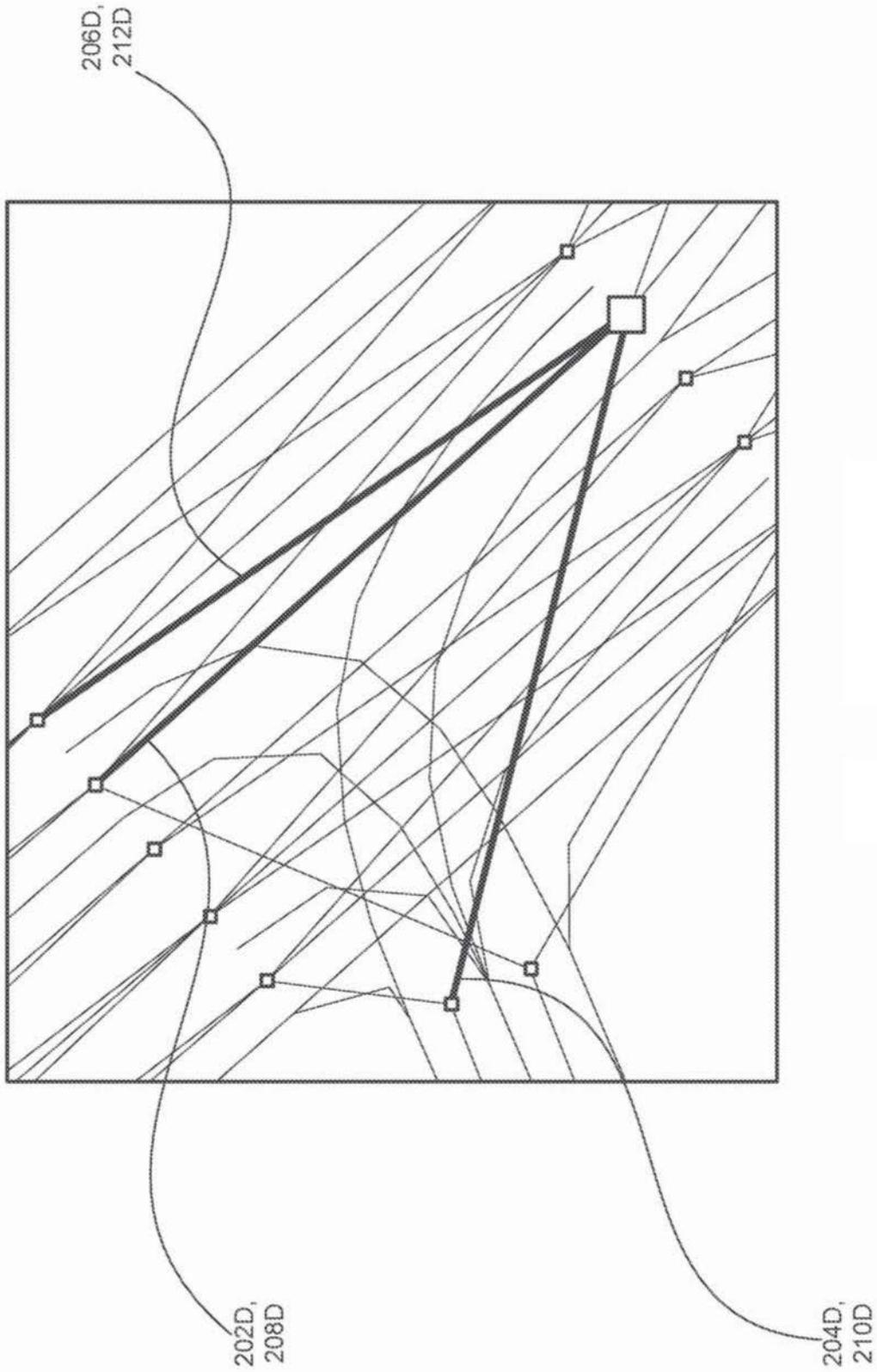


图2D

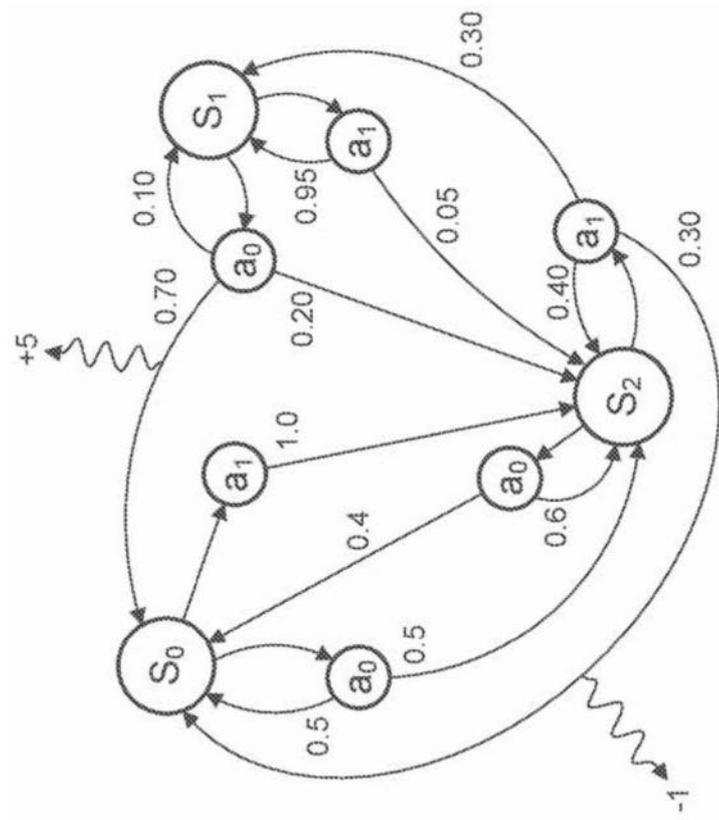


图2E

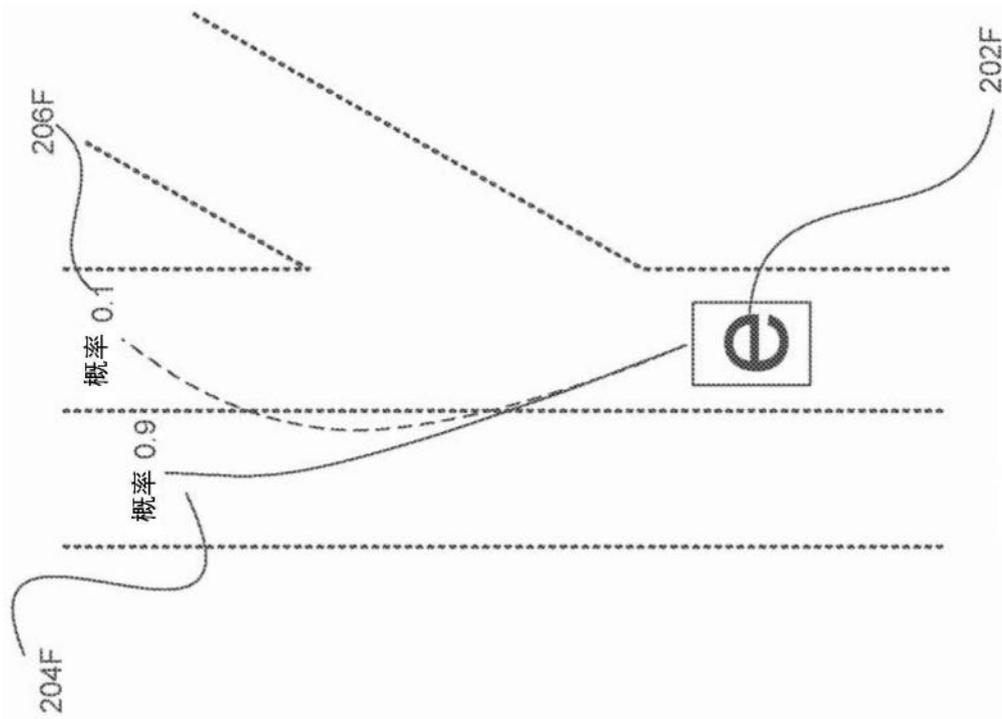


图2F

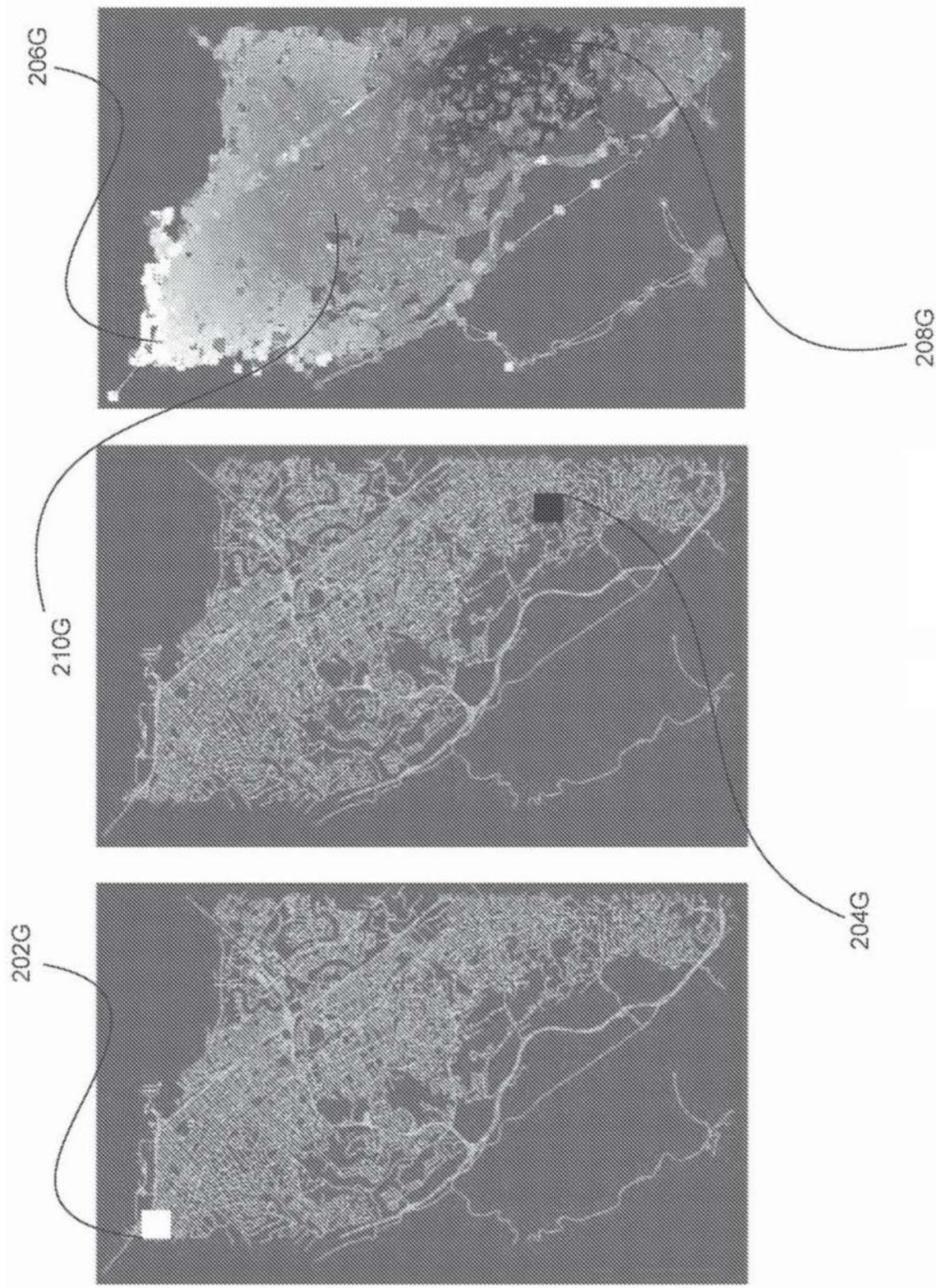


图2G

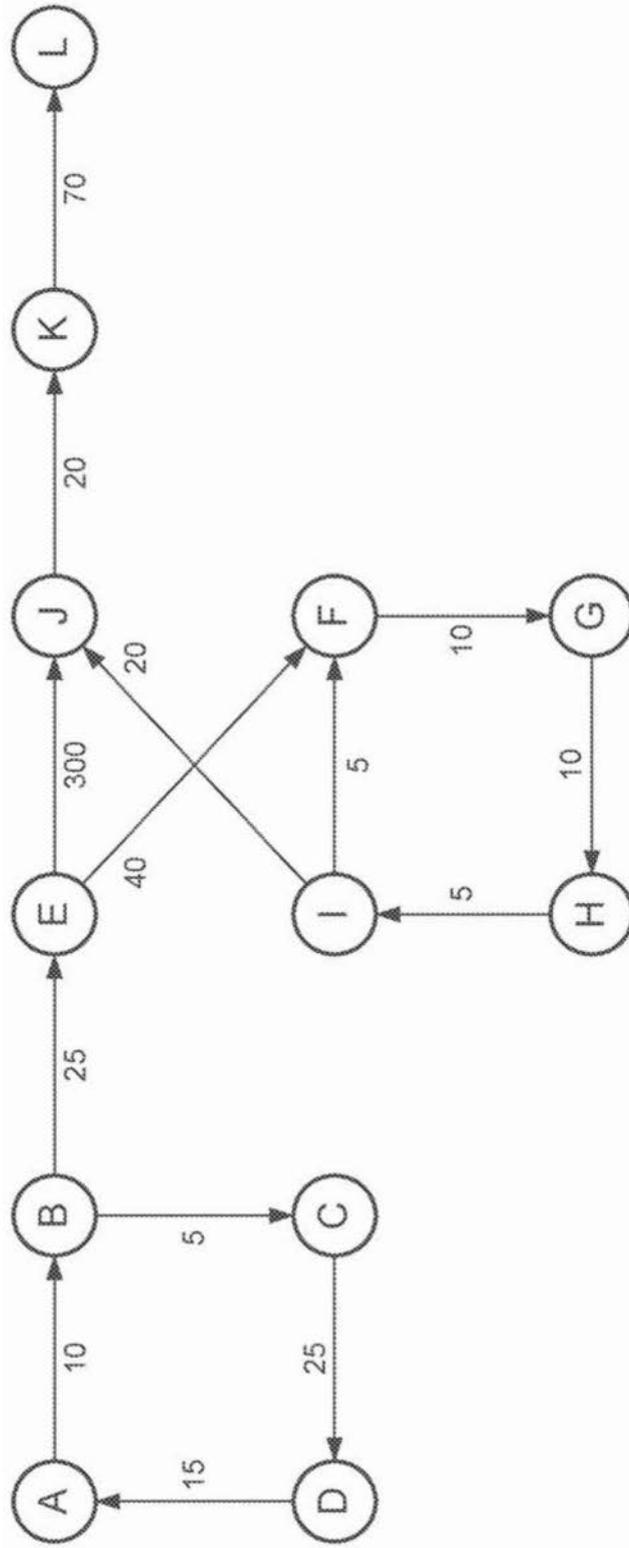


图3A

节点	依赖性计数
A	1
B	2
C	1
D	1
E	2
F	1
G	1
H	1
I	2
J	1
K	1
L	0

图3B

节点	以秒为单位的预期时间奖励
A	245
B	255
C	0
D	230
E	280
F	-20
G	-10
H	0
I	500
J	580
K	600
L	0

图3C

节点	以秒为单位的预期时间奖励
A	410.5
B	420.5
C	205
D	395.5
E	445.5
F	475
G	485
H	495
I	551
J	580
K	600
L	0

图3D

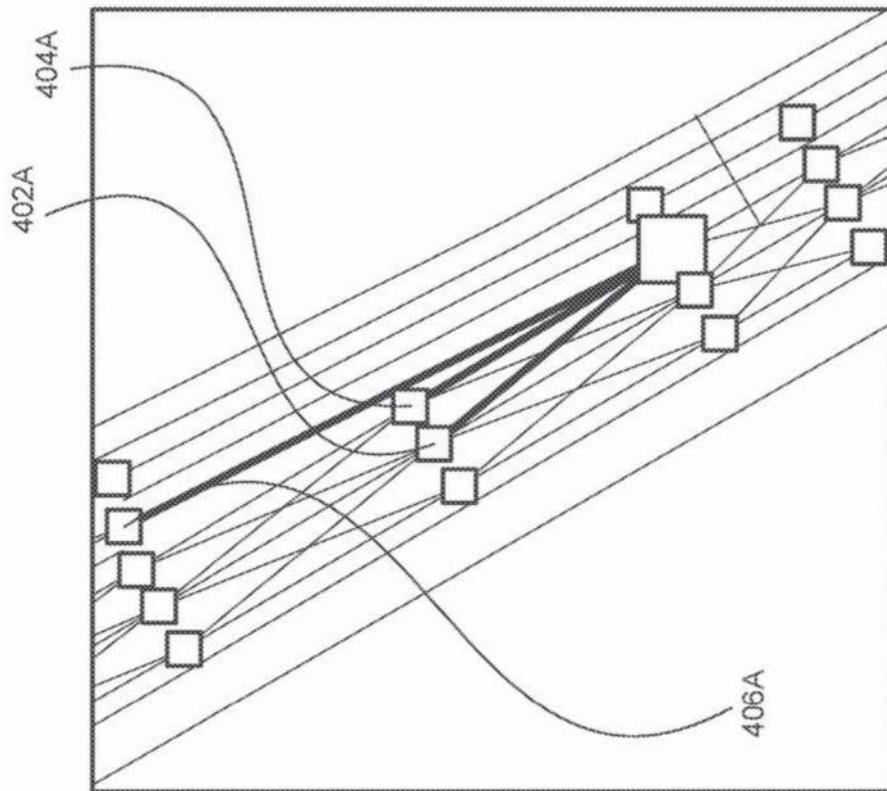


图4A

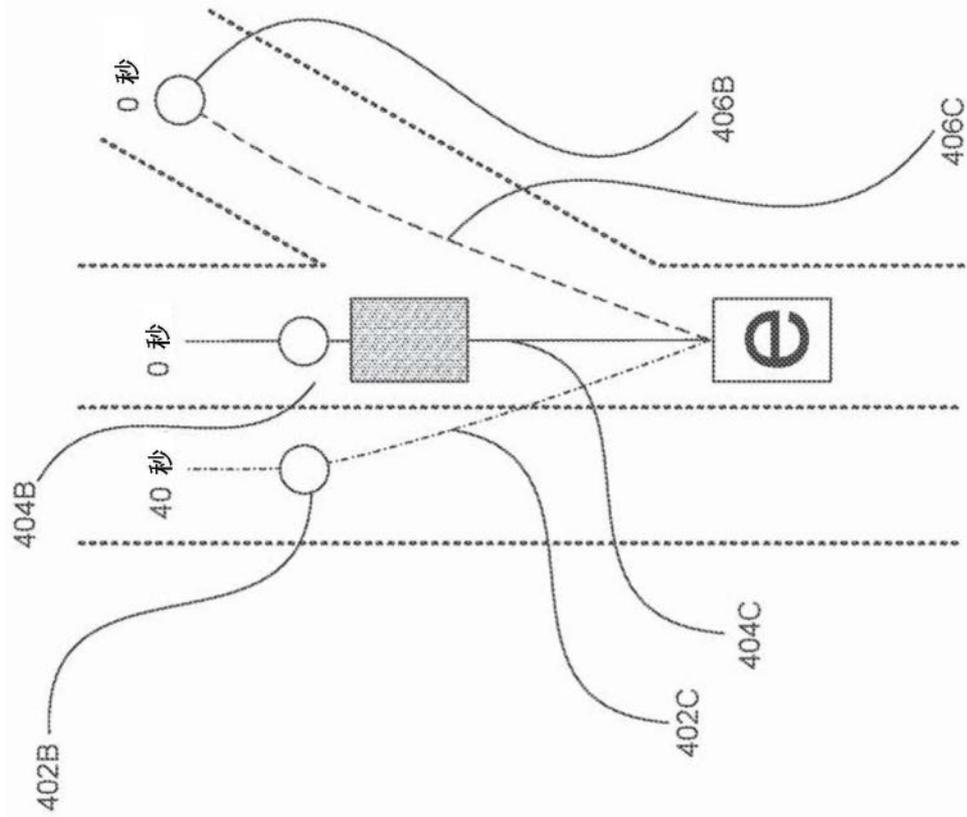


图4B

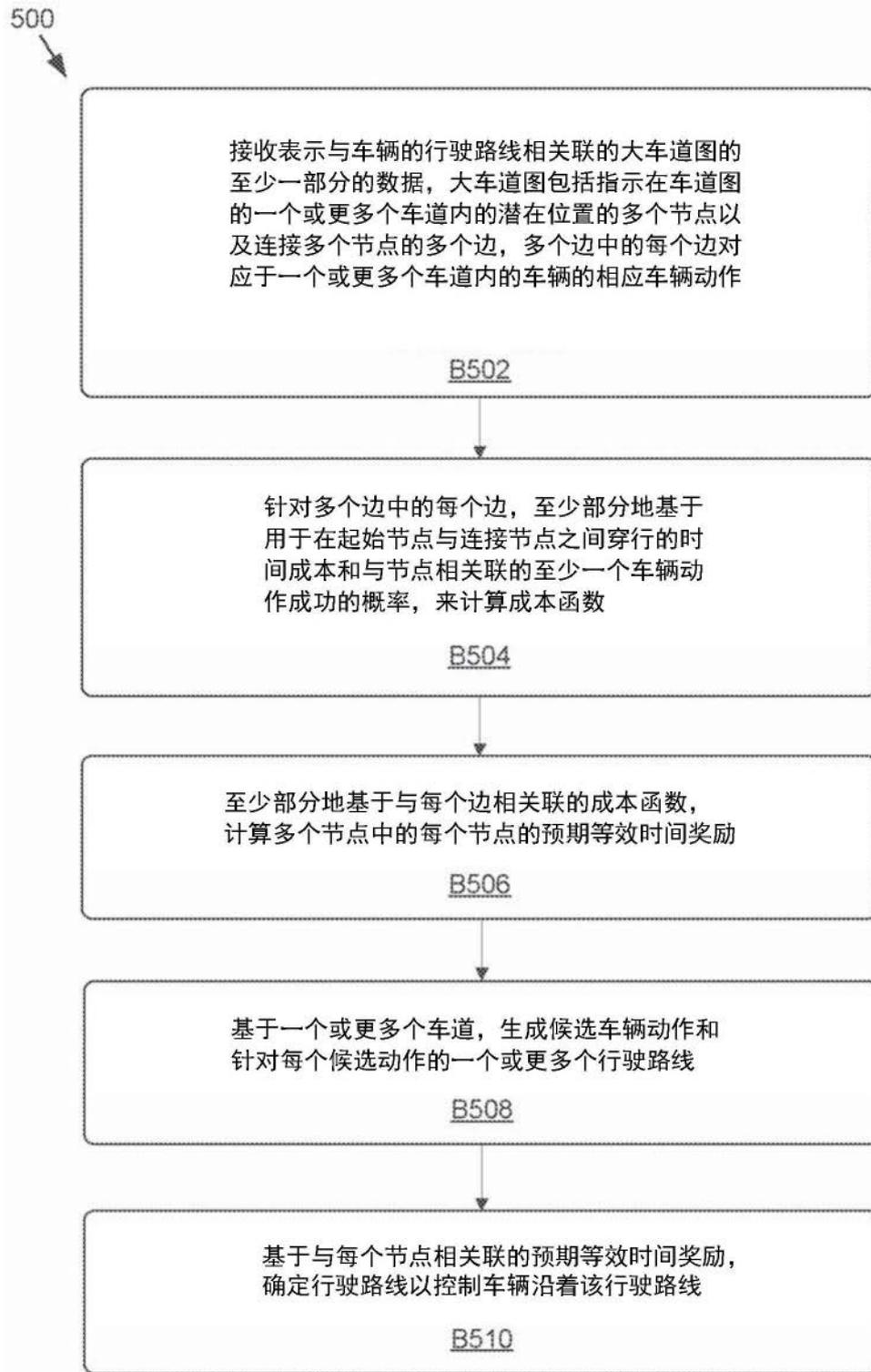


图5

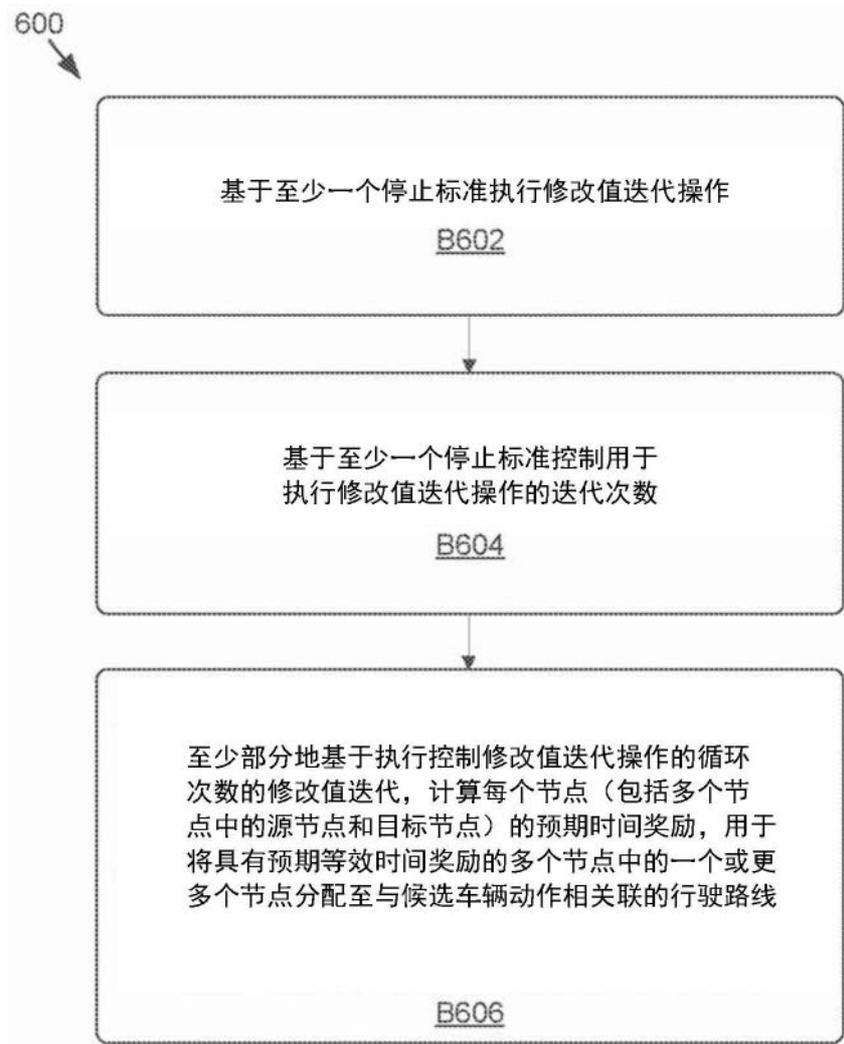


图6

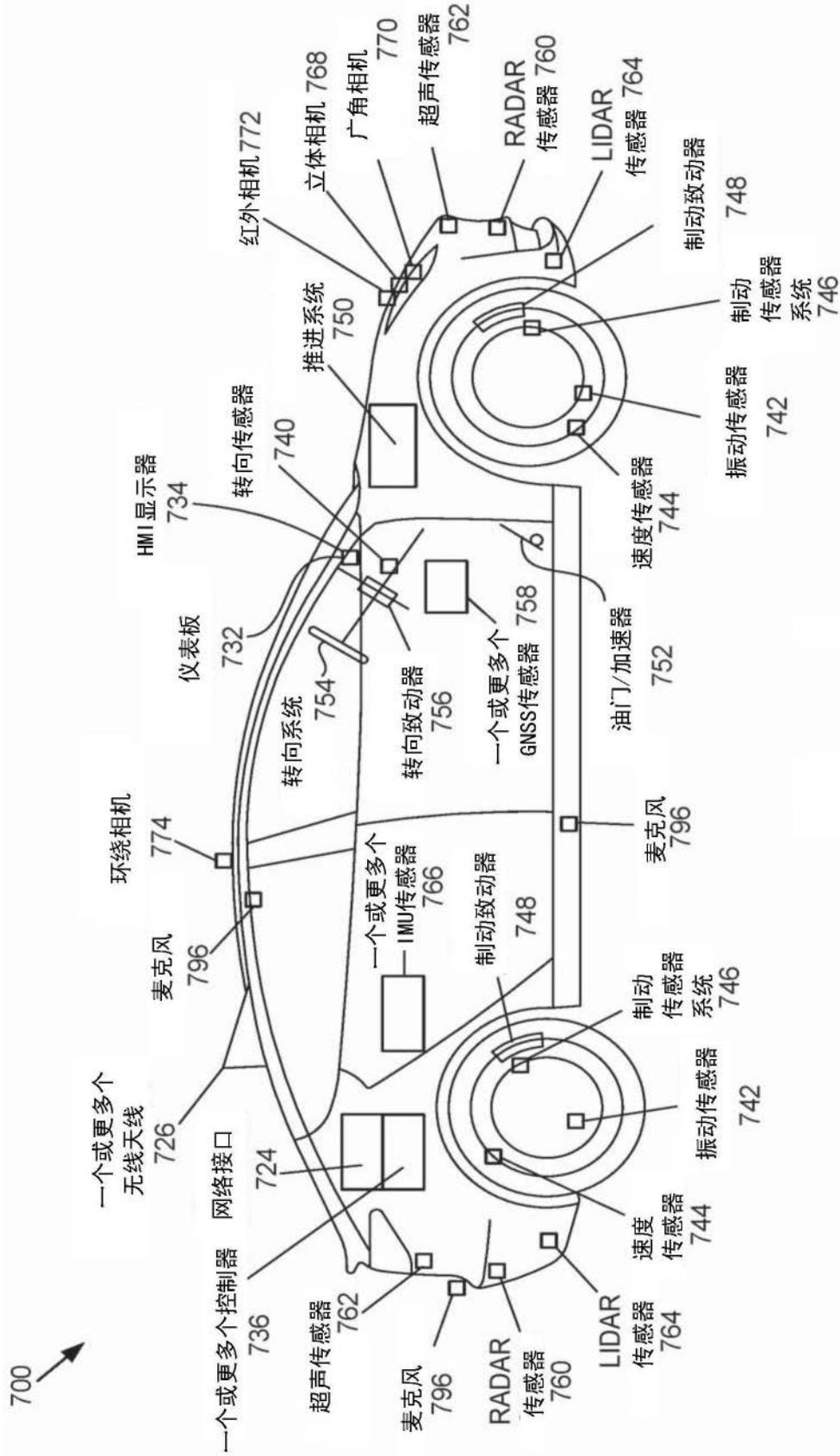


图7A

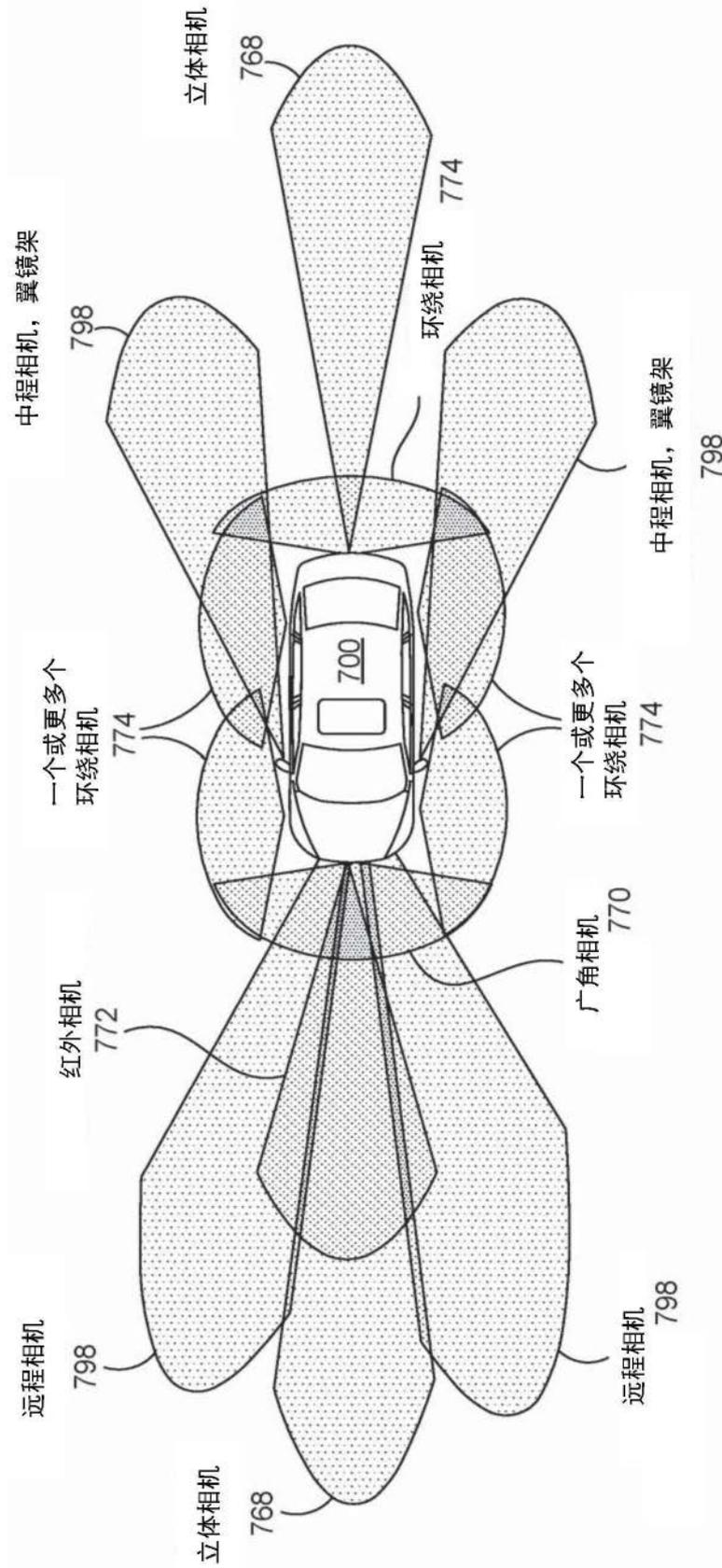


图7B

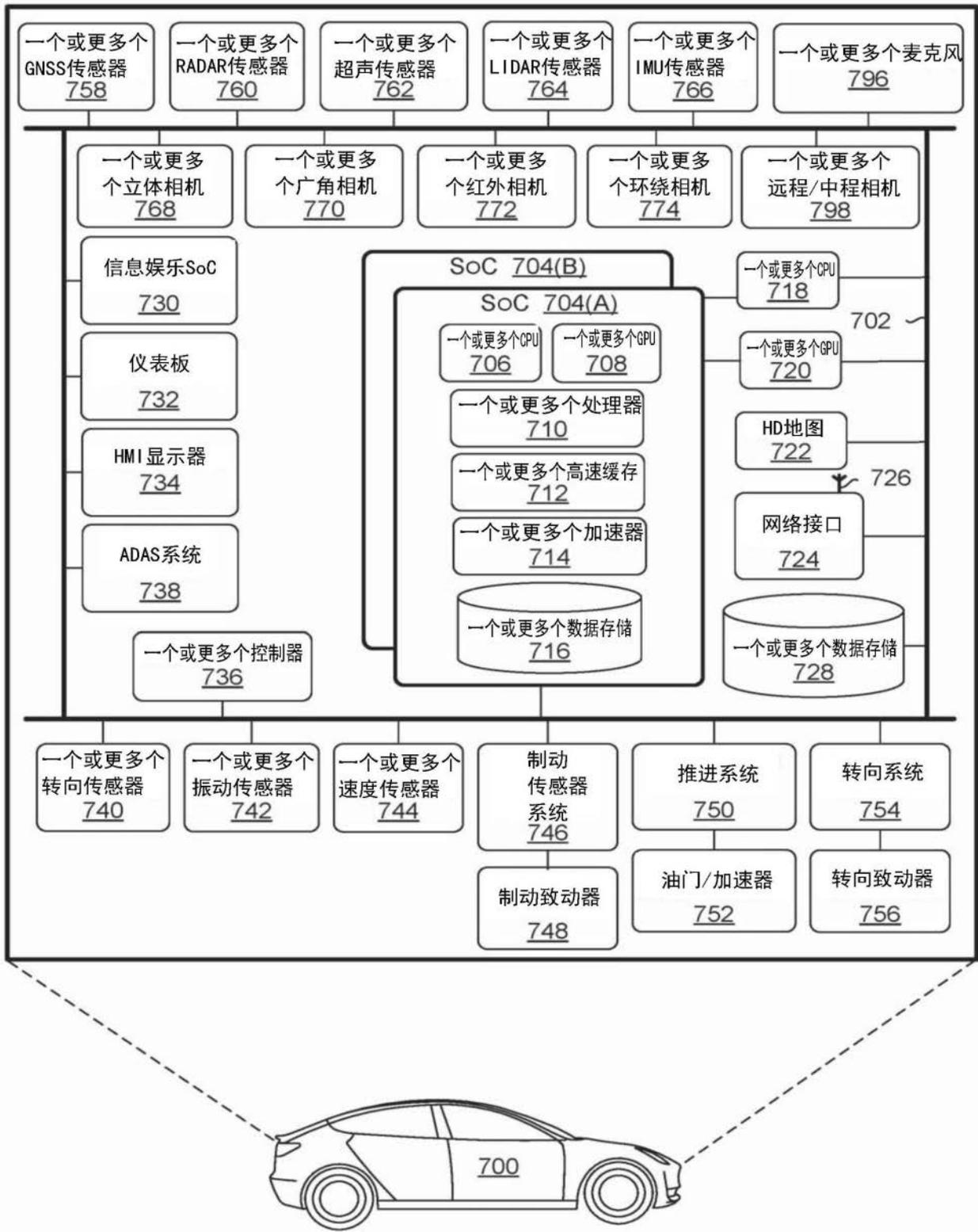


图7C

776 ↗

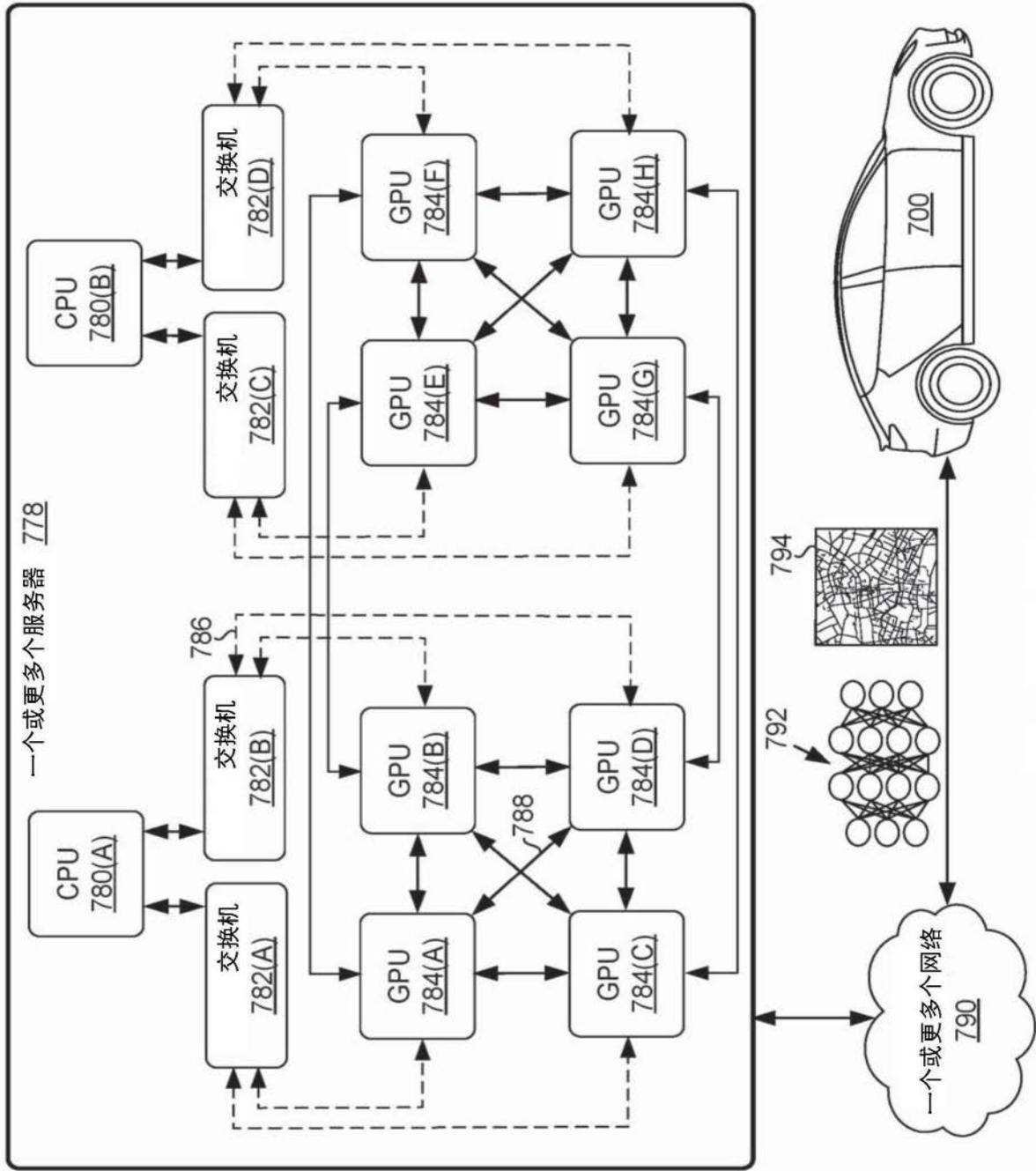


图7D

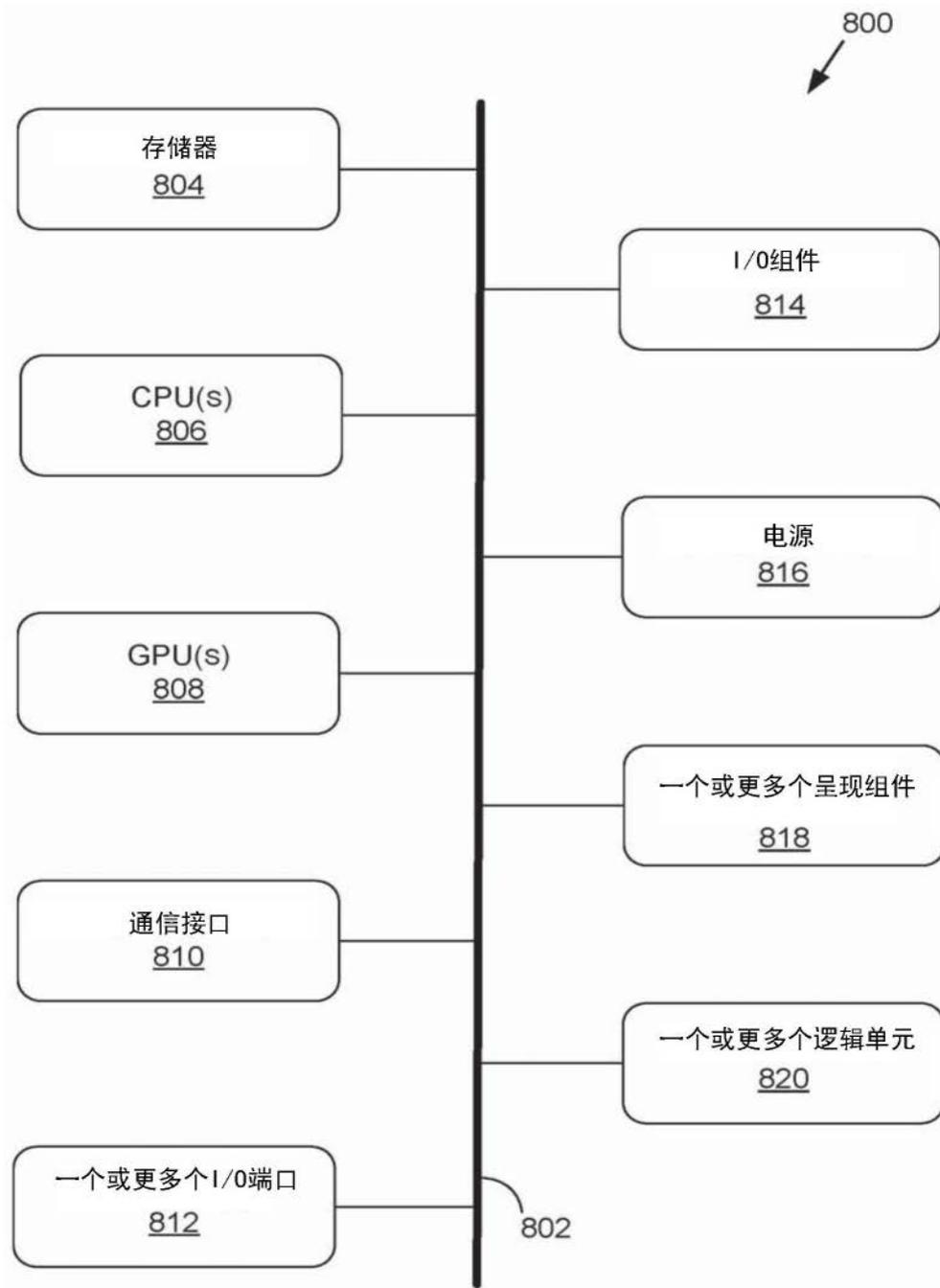


图8



图9