



(12) 发明专利

(10) 授权公告号 CN 110320998 B

(45) 授权公告日 2025. 01. 21

(21) 申请号 201910257948.8

G06N 3/08 (2023.01)

(22) 申请日 2019.04.01

(56) 对比文件

(65) 同一申请的已公布的文献号

CN 107260506 A, 2017.10.20

申请公布号 CN 110320998 A

US 2016081547 A1, 2016.03.24

(43) 申请公布日 2019.10.11

审查员 林佳丽

(30) 优先权数据

15/942,003 2018.03.30 US

(73) 专利权人 托比股份公司

地址 瑞典丹德吕德

(72) 发明人 埃里克·林登

(74) 专利代理机构 上海脱颖律师事务所 31259

专利代理师 脱颖

(51) Int. Cl.

G06F 3/01 (2006.01)

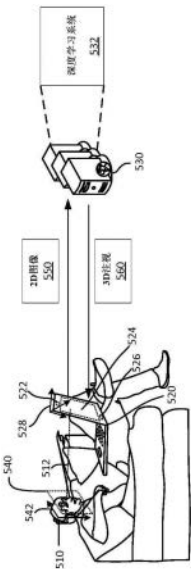
权利要求书3页 说明书16页 附图13页

(54) 发明名称

对用于三维 (3D) 注视预测的神经网络的训练

(57) 摘要

本申请描述基于深度学习系统产生3D注视预测的技术。在一个示例中,深度学习系统包括神经网络。使用训练图像来训练神经网络,训练图像由相机产生,并且示出注视刺激点时的用户眼睛。刺激点中的一些在相机的平面中。其余的刺激点不在相机的平面中。训练包括输入与相机平面中的刺激点相关联的第一训练图像,以及输入与相机平面之外的刺激点相关联的第二训练图像。训练基于刺激点中的至少一个与注视线之间的距离来最小化神经网络的损失函数。



1. 一种非暂时性计算机可读存储介质,其存储指令,一旦在计算机系统上执行,所述指令使得所述计算机系统执行操作,所述操作包括:

访问训练图像,所述训练图像包括第一组训练图像和第二组训练图像,其中,所述第一组训练图像示出与相机的平面中的注视点相关联的用户眼睛,并且其中,所述第二组训练图像示出与所述相机的所述平面之外的注视点相关联的用户眼睛;以及

基于所述训练图像训练神经网络,其中,所述训练包括:

将第一训练图像输入神经网络,其中,所述第一训练图像属于所述第一组训练图像;

将第二训练图像输入所述神经网络,其中,所述第二训练图像属于所述第二组训练图像;

基于注视点和注视线之间的距离而最小化所述神经网络的损失函数,其中,所述注视点与所述第一训练图像或所述第二训练图像中的一者相关联,并且其中,所述注视线是由所述神经网络针对在所述第一训练图像或所述第二训练图像中的所述一者中示出的用户眼睛预测的。

2. 根据权利要求1所述的非暂时性计算机可读存储介质,其中,一旦完成训练,所述神经网络基于以用户眼睛为中心的扭曲图像预测: (i) 距离校正, (ii) 所述用户眼睛的2D注视原点,以及 (iii) 所述用户眼睛的2D注视方向。

3. 根据权利要求1所述的非暂时性计算机可读存储介质,其中,所述第一训练图像示出根据注视角注视所述注视点时的所述用户眼睛,其中,所述第二训练图像示出根据所述注视角注视另一注视点时的所述用户眼睛。

4. 根据权利要求1所述的非暂时性计算机可读存储介质,其中,所述第一训练图像和所述第二训练图像示出以注视角注视所述注视点时的所述用户眼睛,其中,所述第一训练图像对应于所述相机和所述用户眼睛之间的第一距离,并且其中,所述第二训练图像对应于所述相机和所述用户眼睛之间的第二距离。

5. 根据权利要求1所述的非暂时性计算机可读存储介质,其中,所述第一训练图像示出根据注视角注视所述注视点时的所述用户眼睛,其中,所述第二训练图像示出根据不同的注视角注视另一注视点时的所述用户眼睛。

6. 根据权利要求1所述的非暂时性计算机可读存储介质,其中,训练所述神经网络进一步包括:

在训练所述神经网络期间,基于校准参数来更新所述神经网络的参数。

7. 根据权利要求6所述的非暂时性计算机可读存储介质,其中,一旦完成训练,将所述校准参数的更新值输入到所述神经网络。

8. 一种计算机系统,其包括:

处理器;以及

存储器,所述存储器存储计算机可读指令,一旦由所述处理器执行,所述计算机可读指令将所述计算机系统配置为执行操作,所述操作包括:

访问训练图像,所述训练图像包括第一组训练图像和第二组训练图像,其中,所述第一组训练图像示出与相机的平面中的注视点相关联的用户眼睛,并且其中,所述第二组训练图像示出与所述相机的所述平面之外的注视点相关联的用户眼睛;以及

基于所述训练图像训练神经网络,其中,所述训练包括:

将第一训练图像输入神经网络,其中,所述第一训练图像属于所述第一组训练图像;

将第二训练图像输入所述神经网络,其中,所述第二训练图像属于所述第二组训练图像;

基于注视点和注视线之间的距离而最小化所述神经网络的损失函数,其中,所述注视点与所述第一训练图像或所述第二训练图像中的一者相关联,并且其中,所述注视线是由所述神经网络针对在所述第一训练图像或所述第二训练图像中的所述一者中示出的用户眼睛预测的。

9. 根据权利要求8所述的计算机系统,其中,一旦完成训练,所述神经网络基于以用户眼睛为中心的扭曲图像预测: (i) 距离校正, (ii) 所述用户眼睛的2D注视原点,以及 (iii) 所述用户眼睛的2D注视方向。

10. 根据权利要求9所述的计算机系统,其中,所述操作进一步包括,一旦完成训练:

基于确定眼睛跟踪系统的相机和用户眼睛之间的估算距离而将所述扭曲图像输入所述神经网络,所述扭曲图像基于二维(2D)图像而产生,所述2D图像由所述相机产生并且示出所述用户眼睛;

至少通过基于所述距离校正更新所述估算距离来产生所述用户眼睛和所述相机之间的校正距离;以及

基于所述2D注视原点、所述2D注视方向和所述校正距离来产生所述用户眼睛的3D注视信息。

11. 根据权利要求10所述的计算机系统,其中,所述操作进一步包括:

基于所述2D图像确定所述相机和所述用户眼睛之间的所述估算距离;以及

基于所述校正距离和所述相机在3D空间中的位置来估算所述用户眼睛在所述3D空间中的位置,其中,所述3D注视信息包括所述用户眼睛在所述3D空间中的所述位置。

12. 根据权利要求11所述的计算机系统,其中,所述操作进一步包括:

基于所述2D注视原点和所述2D注视方向来估算从所述用户眼睛在所述3D空间中的所述位置起始的3D注视方向,其中,所述3D注视信息包括所述3D注视方向。

13. 根据权利要求11所述的计算机系统,其中,产生所述扭曲图像包括归一化所述2D图像以产生归一化图像,并且围绕所述用户眼睛剪裁所述归一化图像以产生所述扭曲图像。

14. 根据权利要求11所述的计算机系统,其中所述2D图像进一步示出第二用户眼睛和用户脸部,并且其中,所述操作进一步包括:

由所述计算机系统从所述2D图像产生围绕所述第二用户眼睛的第二扭曲图像;

从所述2D图像产生围绕所述用户脸部的第三扭曲图像;以及

将所述第二扭曲图像和所述第三扭曲图像输入到所述神经网络,其中,所述神经网络进一步基于所述第二扭曲图像和所述第三扭曲图像来预测所述距离校正。

15. 根据权利要求14所述的计算机系统,其中,所述操作进一步包括:

基于所述校正距离来产生所述第二用户眼睛在所述3D空间中的第二位置和从所述3D空间中的所述第二位置起始的第二注视方向,

其中,所述3D注视信息包括所述第二位置和所述第二注视方向。

16. 根据权利要求8所述的计算机系统,其中,所述训练进一步包括:

将校准参数输入到所述神经网络;以及

基于所述损失函数来更新所述神经网络的网络参数和所述校准参数。

17. 根据权利要求16所述的计算机系统,其中所述操作进一步包括,一旦完成训练:

将校准图像和校准参数输入到所述神经网络,所述校准图像基于与用户的眼睛跟踪系统相关联的相机所产生的图像数据而示出所述用户的眼睛,其中,所述图像数据在校准用于所述用户的所述眼睛跟踪系统期间产生;以及

更新所述校准参数而不更新所述神经网络的所述网络参数,其中,所述校准参数至少通过基于所述校准图像和所述校准参数最小化所述损失函数而更新。

18. 根据权利要求17所述的计算机系统,其中,所述操作进一步包括:

由所述计算机系统产生所述用户的三维(3D)注视信息,其中,至少通过以下步骤来产生所述三维(3D)注视信息:

将图像和校准参数输入到所述神经网络,所述图像基于所述相机所产生的附加图像数据示出所述用户的所述眼睛,以及

从所述神经网络接收基于所述图像和所述校准参数的预测,其中,所述预测包括距离校正、所述图像中所述用户的所述眼睛的2D注视原点以及所述图像中所述用户的所述眼睛的2D注视方向。

19. 一种计算机实施的方法,其包括:

由计算机系统访问训练图像,所述训练图像包括第一组训练图像和第二组训练图像,其中,所述第一组训练图像示出与相机的平面中的注视点相关联的用户眼睛,并且其中,所述第二组训练图像示出与所述相机的所述平面之外的注视点相关联的用户眼睛;以及

由所述计算机系统基于所述训练图像来训练神经网络,其中,所述训练包括:

将第一训练图像输入神经网络,其中,所述第一训练图像属于所述第一组训练图像;

将第二训练图像输入所述神经网络,其中,所述第二训练图像属于所述第二组训练图像;

基于注视点和注视线之间的距离而最小化所述神经网络的损失函数,其中,所述注视点与所述第一训练图像或所述第二训练图像中的一者相关联,并且其中,所述注视线由所述神经网络针对在所述第一训练图像或所述第二训练图像中的所述一者中示出的用户眼睛而预测。

20. 根据权利要求19所述的计算机实施的方法,其中,一旦完成训练,所述神经网络基于以用户眼睛为中心的扭曲图像而预测:(i) 距离校正,(ii) 所述用户眼睛的2D注视原点,以及(iii) 所述用户眼睛的2D注视方向。

对用于三维 (3D) 注视预测的神经网络的训练

技术领域

[0001] 本申请涉及注视检测系统和方法。在一个实例中,该系统和方法依赖于深度学习系统——例如神经网络——以检测三维 (3D) 注视。

背景技术

[0002] 与计算设备的互动在当今世界中是十分重要的动作。计算设备——诸如个人计算机、平板计算机、智能电话——在日常生活中随处可见。另外,可佩戴的计算设备——例如可佩戴的头戴设备(例如,虚拟现实头戴设备和增强现实头戴设备)——正变得越来越流行。用于与这些设备互动的系统和方法定义了这些设备是如何使用的以及这些设备是用来做什么的。

[0003] 眼睛跟踪技术的发展使得能够使用人的注视信息来与计算设备交互。换句话说,显示器上用户的注视位置。该信息可单独地用于交互,或与基于接触的互动技术(例如,使用用户输入设备,比如键盘、鼠标、触摸屏或其它输入/输出接口)相结合以用于交互。

[0004] 之前提出的使用注视信息的交互技术可以在美国专利6,204,828、美国专利申请公布20130169560、美国专利7,113,170、美国专利申请公布20140247232和美国专利9,619,020中找到。这些专利和申请的全文援引包含于此。

[0005] 总的来说,基于注视的交互技术依赖于检测用户对注视点的注视。已有的系统和方法可精确地检测二维 (2D) 注视。最近,已采用神经网络来检测所述2D注视。

[0006] 已尝试将依赖于神经网络的已有技术扩展到三维 (3D) 注视。然而,预测的准确性不如对2D注视的预测准确性那么好。如果没有准确的3D注视跟踪,则对于立体显示和3D应用的支持受到显著限制。此外,即便在2D领域中,神经网络通常是针对具体相机和屏幕配置(例如,图像分辨率、焦长、与计算设备的屏幕的距离、屏幕尺寸等)而被训练的。因此,每当所述配置改变(例如,不同的图像分辨率、不同的屏幕尺寸等),神经网络不再能够以可接受的准确性来预测2D注视。对于新的配置重新训练神经网络将变得需要。

发明内容

[0007] 本公开的实施例涉及基于深度学习系统的三维 (3D) 注视检测。在一个示例中,计算机系统访问训练图像,所述训练图像包括第一组训练图像和第二组训练图像。第一组训练图像展示与相机平面内的注视点相关联的用户眼睛。第二组训练图像展示与相机平面之外的注视点相关联的用户眼睛。计算机系统基于训练图像对神经网络进行训练。训练包括将第一训练图像输入到神经网络。第一训练图像属于第一组训练图像。训练也包括将第二训练图像输入到神经网络。第二训练图像属于第二组训练图像。训练还包括基于注视点与注视视线之间的距离使神经网络的损失函数最小化。注视点关联于第一训练图像或第二训练图像中的一个。对于第一训练图像或第二训练图像中的该一个中所示的用户眼睛,通过神经网络来预测注视线。

[0008] 提到这些解说性实施例并不旨在对本公开进行限制或定义,而是为了提供帮助理

解本公开的实例。在具体实施例部分阐述了额外的实施例,并且在那里提供了进一步的说明。

附图简述

[0009] 可通过参照下列附图来实现对各种实施例的本质和优点的进一步理解。在附图中,相似的部件或特征可具有相似的附图标记。此外,相同类型的各个部件可通过在该附图标记后面加上折线和在类似部件之间进行区分的第二附图标记来作出区分。只要在说明书中使用第一附图标记,那么这种描述可适用于具有相同的第一附图标记的任何一个类似部件,而不管第二附图标记如何。

[0010] 图1示出根据一个实施例的眼睛跟踪系统。

[0011] 图2示出根据一个实施例的、由图像传感器捕获的眼睛的图像的实例。

[0012] 图3是示出根据一个实施例的专用计算机系统的方框图。

[0013] 图4示出根据一个实施例的、实施眼睛跟踪系统的部件的可佩戴计算设备的实例。

[0014] 图5示出根据一个实施例的、基于深度学习系统来预测3D注视的实例计算环境。

[0015] 图6示出根据一个实施例的、用于预测3D注视的深度学习系统的实例部件。

[0016] 图7示出根据一个实施例的用于神经网络的实例网络架构。

[0017] 图8示出根据一个实施例的实例图像归一化。

[0018] 图9示出根据一个实施例的实例3D注视预测。

[0019] 图10示出根据一个实施例的由神经网络产生的对2D注视向量以及距离校正的预测。

[0020] 图11示出根据一个实施例的基于深度学习系统预测3D注视的实例流程。

[0021] 图12示出根据一个实施例的使用训练图像的策略,该训练图像与注视点位置相对于产生训练图像的相机的2D平面的多样性相关联。

[0022] 图13示出根据一个实施例的用于训练神经网络的实例流程。

[0023] 图14示出根据一个实施例的使用嵌入式校准参数来训练神经网络的实例流程。

具体实施方式

[0024] 本公开的实施例涉及基于深度学习系统的三维(3D)注视检测。在一个实例中,使用神经网络。该神经网络可独立于相机和屏幕配置而使用。换句话说,不管具体的相机、屏幕以及相机和屏幕配置的组合如何,神经网络都能被正确地训练来预测注视信息以支持3D注视检测。

[0025] 更具体地,相机捕捉正在注视着3D空间内的某个点的用户的二维(2D)图像。从该2D图像估算出相机与用户眼睛之间的粗略距离。该2D图像被归一化以按照高分辨率产生以用户眼睛为中心的扭曲图像,并且按低分辨率产生用户脸部周围的扭曲图像。这些扭曲图像被输入到神经网络,该神经网络则预测每个眼睛的2D注视向量以及对粗略距离的距离校正。眼睛在3D空间中的位置是基于粗略距离和距离校正以及相机在3D空间内的位置来估算的。基于2D空间与3D空间之间的映射函数,每个2D注视向量被映射到3D注视方向上。眼睛的3D位置及其3D注视方向指示与该眼睛相关联的3D注视。

[0026] 由于使用归一化2D图像并由于神经网络预测距离校正和2D注视向量,因此神经网络

络的预测不依赖于相机和屏幕配置。因而,在不同的眼睛跟踪系统中可以使用相同的受过训练的神经网络,前述不同的眼睛跟踪系统包括与不同类型的智能电话、平板计算机、笔记本电脑、可佩戴头戴设备(例如,虚拟现实和增强现实头戴设备)整合在一起的眼睛跟踪系统以及独立的眼睛跟踪系统。此外,由于检测3D注视,因此可支持立体显示和3D应用。

[0027] 神经网络的训练总体上依赖于多个训练图像,这些训练图像就注视点相对于用于捕捉训练图像的相机的2D平面(例如,对于每个相机来说,与相机的视线垂直的X、Y平面)的位置而言是各不相同的。具体地说,其中一些训练图像展示向位于相机的2D平面内的注视点注视的用户眼睛,而其它训练图像展示向该2D平面之外的注视点注视的用户眼睛。在训练期间,神经网络寻找用户眼睛-相机和用户眼睛-注视点之间的注视角,以及眼睛至相机的距离。由于使用了多样性,因此神经网络能够正确地找到前述注视角和距离。从该注视角预测出注视线(也被称为注视射线)并且计算出注视点与注视线之间的距离(注视点至注视线的距离)。神经网络的损失函数牵涉到注视点至注视线的距离。在训练期间,神经网络的参数(例如,在不同层的节点之间的连接的权重)被更新,以通过使真实的注视点至注视线的距离与预测的注视点至注视线的距离之间的差最小化来使损失函数最小化。为简单起见,从这里开始,图像指2D图像,除非另有规定。

[0028] 独立于相机和屏幕配置地训练和使用神经网络以进行3D注视预测的这些和其它特征将在下文中进一步被描述。本文接下来描述眼睛跟踪系统的各种部件和配置以提供对3D注视预测技术的更好理解。

[0029] 此外,在训练期间,神经网络对于训练图像中示出的人们学习“n个”校准参数。神经网络的设计者只需要指定数值“n”而不需要实际地指定校准参数。这些参数是对神经网络的各个层的输入并且成为训练的一部分。一旦训练结束,对跟踪系统进行操作的最终用户一般接着执行校准过程。作为该校准过程的一部分,校准图像(例如,其中请求最终用户注视某些特定的点)被输入到神经网络,该神经网络则对于该用户产生“n个”校准参数。这些参数用于针对最终用户的注视预测。

[0030] 图1示出根据一个实施例的眼睛跟踪系统100(也被称为注视跟踪系统)。系统100包括:照明器111、112,用于照射用户的眼睛;以及图像传感器113,用于捕捉用户眼睛的图像。照明器111、112例如可以是发出红外频带或近红外频带的光的发光二极管。图像传感器113可以例如是相机,比如互补金属氧化物半导体(CMOS)相机或电荷耦合器件(CCD)相机。该相机不仅限于红外相机或深度相机或光场相机。图像传感器的快门机构可以是滚动快门或全局快门。

[0031] 第一照明器111与图像传感器113同轴地布置(或靠近图像传感器113),以使图像传感器113能够捕捉用户眼睛的发亮瞳孔图像。由于第一照明器111和图像传感器113的同轴布置,从眼睛的视网膜反射的光穿过瞳孔朝向图像传感器113返回,以使得瞳孔在第一照明器111照射眼睛的图像中看上去比围绕瞳孔的虹膜更亮。第二照明器112与图像传感器113不同轴地设置(或较远离图像传感器113)以捕捉昏暗的瞳孔图像。由于第二照明器112和图像传感器113的非同轴布置,从眼睛的视网膜反射的光不会到达图像传感器113,并且瞳孔在第二照明器112照射眼睛的图像中看上去比瞳孔周围的虹膜更暗。照明器111、112例如可轮流照射眼睛,以使每个第一图像是发亮瞳孔图像而每个第二图像是昏暗瞳孔图像。

[0032] 眼睛跟踪系统100也包括电路120(例如,包括一个或多个处理器),用于处理由图

像传感器113捕捉的图像。电路120可例如经由有线连接或无线连接而连接至图像传感器113和照明器111、112。在另一示例中,以一个或多个处理器形式出现的电路120可以被设置在图像传感器113的光敏表面下面的一个或多个层叠的层内。

[0033] 图2示出由图像传感器113捕捉的眼睛200的图像的一个示例。电路120可以例如采用图像处理(例如,数字图像处理)以提取图像中的特征。电路120可例如采用瞳孔中心角膜反射(PCCR)眼睛跟踪来确定眼睛200看向哪里。在PCCR眼睛跟踪中,处理器120估算瞳孔210的中心的位置以及眼睛200的闪光点220的中心的位置。闪光点220是由来自照明器111、112中的一个的光的反射造成的。处理器120使用闪光点220计算用户在空间内的位置,并使用瞳孔210计算用户眼睛200所朝向的位置。由于眼睛200的光学中心与中央凹(fovea)之间通常有偏移,处理器120执行对中央凹偏移的校准以能够确定用户看向哪里。然后可以将从左眼和从右眼获得的注视方向组合起来以形成组合的估算的注视方向(或观看方向)。如下文将要描述的那样,许多不同的因素可能影响当形成这种组合时左眼的注视方向和右眼的注视方向相对于彼此应当占据怎样权重。

[0034] 在参照图1描述的实施例中,照明器111、112被布置在眼睛跟踪模块110内,该眼睛跟踪模块110被布置在用户观看的显示器下面。这种布置仅作为示例。将理解,可采用任意更多或更少数量的照明器和图像传感器以用于眼睛跟踪,并且这些照明器和图像传感器可相对于用户观看的显示器以许多不同的方式分布。将理解,本公开中描述的眼睛跟踪方案例如可以被采用以用于远程眼睛跟踪(例如,在个人计算机中、智能电话中,或集成在车辆内),或用于可佩戴眼睛跟踪(例如,在虚拟现实眼镜或增强现实眼镜中)。

[0035] 图3是示出其中可实施本公开的实施例的专用计算机系统300的方框图。该示例展示了专用计算机系统300,其例如可完整地、部分地或通过各种修改地使用以提供本文描述的各个部件的功能。

[0036] 专用计算机系统300被图示为包括可经由总线390电耦接的硬件元件。这些硬件元件可包括一个或多个中央处理单元310、一个或多个输入设备320(例如,鼠标、键盘、眼睛跟踪设备等)以及一个或多个输出设备330(例如,显示设备、打印机等)。专用计算机系统300也可包括一个或多个存储设备340。作为示例,存储设备340可以是盘驱动器、光存储设备、固态存储设备(例如,随机存取存储器(RAM)和/或只读存储器(ROM)),这些存储设备是可编程的、可闪速更新的和/或类似的。

[0037] 专用计算机系统300可另外包括计算机可读存储介质读取器350、通信系统360(例如,调制解调器、网卡(无线的或有线的)、红外通信设备、蓝牙™设备、蜂窝通信设备等)以及工作存储器380,所述工作存储器可包括如前所述的RAM和ROM设备。在一些实施例中,专用计算机系统300也可包括处理加速单元370,所述处理加速单元可包括数字信号处理器、专用处理器和/或类似物。

[0038] 图4示出实现结合图1-2描述的眼睛跟踪系统的上述部件的一些或全部的可佩戴计算设备400的示例。可佩戴计算设备400可以是可由用户佩戴的虚拟现实(VR)头戴设备或增强现实(AR)头戴设备。如图所示,可佩戴计算设备400包括一组透镜410(例如,菲涅尔透镜)、一组相机420、一组热反射镜430(例如,如图12-14进一步展示的,这组热反射镜在各个实施例中包括两个热反射镜以用于每个眼睛)以及一组显示器440。相机420可包括图1的图像传感器113。尽管图4中未示出,但可佩戴计算设备400也可包括一组照明器和处理电路。

这些和其它部件可被集成在可佩戴计算设备400的外壳450内。如此,一旦用户将可佩戴计算设备400安装到他/她的头上,这组透镜410将相对靠近用户的眼睛,并且这组显示器将相对远离用户的眼睛,并且其余的部件可位于透镜和显示器之间。这些部件的布置允许在三维虚拟或真实空间内检测用户的注视点460。

[0039] 本文接下来描述使用深度学习系统来进行3D注视预测。为了便于清楚解释,结合相机、屏幕和用户双眼来描述该系统(例如,相机捕捉图像,所捕获的图像的一些或者全部展示用户的双眼)。深度学习系统可与使用可见光、被动红外光、主动亮瞳(BP)红外光等进行眼睛跟踪的任意相机和屏幕配置一起使用。然而,本公开的实施例不局限于此。

[0040] 例如,这些实施例同样适用于对每个用户眼睛使用一个相机的眼睛跟踪系统,例如在虚拟现实头戴设备或增强现实头戴设备的背景下。为了进行每眼单相机跟踪而如何实施深度学习系统的改变在本公开的启发下对于本领域技术人员来说是显而易见的。例如,与其输入两个扭曲图像(每个扭曲图像侧重于用户的其中一只眼睛),倒不如使用与相机关联的用户眼睛的仅一个扭曲图像。此外,用户脸部周围的扭曲图像可以不被输入。在训练期间,神经网络学着从所述扭曲图像预测距离校正,而不是基于用户眼睛和用户脸部周围的扭曲图像来预测这种校正。然后该系统可以输出与相机关联的用户眼睛的2D注视向量以及对粗略的相机至眼睛距离的距离校正。另外,在VR或AR设备的情形下,粗略距离可以是预定义的并且不需要基于由相机产生的图像进行估算。

[0041] 图5示出根据一个实施例的、基于深度学习系统来预测3D注视的示例性计算环境。总的来说,2D注视信息表示在2D平面上的X、Y注视位置。相比而言,3D注视信息不仅表示X、Y注视位置,还表示Z注视位置。在一个示例中,3D注视可由在3D空间内作为原点的眼睛位置以及3D注视从原点开始的方向来表征。

[0042] 如图5所示,用户510操作计算设备520,该计算设备跟踪用户510的3D注视512。为此,计算设备520在一个示例中与宿留有(或托管)深度学习系统532的服务器计算机530通信。计算设备520通过数据网络(未示出)将2D图像550发送至服务器计算机530,该2D图像550示出用户510在注视时的用户眼睛。服务器计算机530将该2D图像550输入至深度学习系统532,该深度学习系统532作为响应来预测3D注视512。服务器计算机530通过数据网络将关于3D注视512的信息560(例如,3D眼睛位置和3D注视方向)发送返回到计算设备520。计算设备520使用该信息560以将基于3D注视的计算服务提供给用户510。

[0043] 尽管图5示出宿留有深度学习系统532的计算机服务器530,但本公开的实施例不局限于此。例如,计算设备520可下载深度学习系统532的代码和宿留有深度学习系统532的例示。如此,计算设备520根据该例示在本地预测3D注视512并且不需要将2D图像550发送至服务器计算机530。在该示例中,服务器计算机530(或通过数据网络与其相连的一些其它计算机系统)可训练深度学习系统532并提供接口(例如,网络接口)以将该深度学习系统530的代码下载至计算设备,由此将深度学习系统530的例示宿留在这些计算设备上。

[0044] 在一个示例中,计算设备520包括相机522、屏幕524和3D注视应用526。相机522产生2D图像550,该2D图像是用户脸部的2D表示540。该2D图像550展示在3D空间内注视时用户的眼睛。可关联于相机522定义3D坐标系528。例如,相机522处于该3D坐标系528的原点处。X和Y平面可以是垂直于相机522的视线的平面。相比而言,2D图像550具有可定义在2D坐标系542周围的2D平面,该2D坐标系542与用户脸部的2D表示540处于相同位置。相机522与2D空

间和3D空间之间(例如,两个坐标系542、528之间)的映射相关联。在一个示例中,该映射包括相机522的背投矩阵并且本地地存储于计算设备520(例如,存储在与3D注视应用526关联的存储位置)。

[0045] 屏幕524可以——但不必——位于相机522的X、Y平面内(若非如此,则基于计算设备520的配置来确定两者之间的相对位置)。3D注视应用526可处理2D图像550以将其输入到深度学习系统530(不管位于计算设备520远端还是在计算设备520本地上),并且可处理关于3D注视的信息560以支持立体显示(如果屏幕524也支持的话)和3D应用(例如,基于信息560对屏幕524上所显示的对象进行3D控制和操纵)。

[0046] 图6示出根据一个实施例的、用于预测3D注视的深度学习系统600的示例性部件。如图所示,深度学习系统包括眼睛检测器620、距离估算器630、图像发生器640、神经网络650和3D注视发生器660。这些部件中的一些或全部可实现为专用硬件和/或软件模块(例如,具体的计算机可读指令),该软件模块被宿留在专用或通用处理硬件上。

[0047] 如图所示,2D图像610被输入到眼睛检测器620。例如,用相机产生该2D图像610。作为响应,眼睛检测器620检测该2D图像610中的用户眼睛,并输出关于眼睛在该图像中的位置622的信息(例如,瞳孔的中心在该2D图像610的2D平面内的位置)。在一个示例中,眼睛检测器620被实现为机器学习算法,该机器学习算法被训练以进行眼睛检测。许多机器学习算法是可能的并且是本领域技术人员所熟知的。

[0048] 眼睛位置622和2D图像610被输入到距离估算器630。作为响应,距离估算器630产生估算的距离632,例如,粗略距离。为此,距离估算器630将2D图像610中检测出的眼睛投影到以相机为中心的3D坐标系内。这种投影使用相机的2D-3D空间映射。距离估算器630搜索3D空间内的眼睛投影,在该眼睛投影处的双眼间距离(ID)大约是人的平均双眼间距离(ID)(例如,63毫米)。相机和这些眼睛投影中的每一个眼睛投影之间距离可被称为投影距离。根据人的平均双眼间距离(ID)和投影距离来设定粗略距离。

[0049] 为了说明,令 d_{rough} 表示粗略距离,并且 K 表示固有相机矩阵,并且 e_{left} 和 e_{right} 表示在2D图像610中检测出的眼睛。 K 是相机的2D-3D空间映射的分量。所投影的左眼 $e'_{\text{left,proj}}$ 和右眼 $e'_{\text{right,proj}}$ 被计算为 $e'_{\text{left,proj}} = K^{-1}e_{\text{left}}$ 和 $e'_{\text{right,proj}} = K^{-1}e_{\text{right}}$,并且表示从2D图像610中检测出的眼睛投影到以相机为中心的3D坐标系中的眼睛向量,如图8所示。这些投影的眼睛被归一化为 $e_{\text{left,proj}} = e'_{\text{left,proj}} / ||e'_{\text{left,proj}}||$ 和 $e_{\text{right,proj}} = e'_{\text{right,proj}} / ||e'_{\text{right,proj}}||$ 。粗略距离被计算为 $d_{\text{rough}} = ID / ||e_{\text{left,proj}} - e_{\text{right,proj}}||$ 。

[0050] 图像发生器640接收2D图像610,并作为响应产生用户眼睛周围的扭曲图像642以及用户脸部周围的扭曲图像644。在一个示例中,图像发生器640从本地存储器访问用户眼睛扭曲图像642的像素中的第一预定义距离 s (a first predefined distance s in pixels)以及用户脸部扭曲图像644的像素中的第二预定义距离 s 。这些预定义距离 s 是像素中的不同双眼间距离 s ,以使用户眼睛扭曲图像642和用户脸部扭曲图像644在3D空间内的不同投影位置并且具有不同的分辨率(通过使用相比第二预定义距离 s 较小的第一预定义距离 s ,用户眼睛扭曲图像642的分辨率高于用户脸部扭曲图像644的分辨率)。图像发生器640使用这些预定义的距离 s 以产生扭曲图像,以使它们相对于相机被归一化。具体地说,每个扭曲图像表示2D图像610在以相机为中心的3D坐标系内的投影、2D图像610围绕X轴的旋转(以使得任何头部倾斜被旋转到水平位置)、2D图像610的缩放(基于预定义的距离 s)以及

扭曲(以使得用户眼睛和用户脸部处于每个扭曲图像的中心,并且不会在几何上歪斜失真)。换句话说,每个扭曲图像代表可被输入到神经网络650的归一化图像,其中所述归一化将该输入对于相机配置(例如,图像分辨率、焦长、至屏幕的距离、相机类型(例如,针孔相机和非针孔相机)等)的依存性解除。

[0051] 为了说明,图像发生器640产生旋转矩阵 R ,该旋转矩阵将点从真实3D空间(例如,以相机为中心的3D坐标系)旋转到归一化的3D空间(例如,同样以相机为中心但相对于真实3D空间旋转以使得2D图像610中的用户双眼之间的向量变得水平的3D坐标系),如图8进一步所示。图像发生器640也基于预定义的距离 s 产生缩放矩阵 M (对于用户眼睛产生矩阵 M ,并对于用户脸部产生另一矩阵 M)。例如, M 被产生为对角矩阵,其中, $M = \text{diag}([1, 1, f])$,其中, f 是被选择为使用户双眼之间的双眼间距离等于预定义距离 s 的焦长。变换矩阵 T 是根据固有矩阵 K 、旋转矩阵 R 和缩放矩阵 M 定义的,并且被用于将2D图像610归一化成归一化的图像(根据 M 矩阵针对用户眼睛和针对用户脸部)。例如,变换矩阵 T 被表示为 $T = MRK^{-1}$,并且被应用于2D图像610以对于用户眼睛在第一预定义距离 s 处产生第一归一化图像并且对于用户脸部在第二预定义距离 s 处产生第二归一化图像。这些归一化图像中的每一个是2D图像610从真实3D空间至归一化3D空间的旋转和投影,其中经旋转和投影的图像在归一化3D空间内在预定义距离 s 处。图像发生器640通过使用双线性插值法(bilinear interpolation)使归一化用户眼睛图像变得扭曲并剪裁出以用户的一只眼睛为中心的 $W \times H$ 区域来产生用户眼睛扭曲图像642。同样,图像发生器640通过使用双线性插值法使归一化用户脸部图像变得扭曲并剪裁出以用户双眼之间的中点为中心的 $W \times H$ 区域来产生用户脸部扭曲图像644。

[0052] 此外,多个眼睛扭曲图像642中的一个被镜像化以使结果得到的镜像化图像与另一(未被镜像化的)眼睛扭曲图像对准。例如,左眼周围的扭曲图像被镜像化。结果,左眼的目内眦与右眼的目内眦对准,并且左眼的目外眦与右眼的目外眦对准,如图10中的镜像化的左眼图像和扭曲的右眼图像所示。通过这种对准,被输入到神经网络的眼睛扭曲图像就取向而言是相同的,由此简化神经网络的架构和训练。

[0053] 眼睛扭曲图像642和脸部扭曲图像644被输入到神经网络650。在一个示例中,2D图像610不被输入到该神经网络650。作为响应,神经网络650输出距离校正652和2D注视向量654(对每个眼睛输出一个)。距离校正652是乘法校正因数,其可以与估算的距离632相乘以校正这种估算的距离并产生经校正的距离。每个2D注视向量654具有在相应的眼睛剪裁图像内的注视原点(例如,瞳孔或闪光点的中心)和注视方向(原点和注视方向在该图像的2D平面内并且可被追溯到真实3D空间)。

[0054] 在一个示例中,神经网络650是包括多个子网络(例如,沿神经网络650的平行分支)的卷积的神经网络。这些子网络(并且等同地,整个卷积神经网络)可以被协同地训练。图7中示出网络架构的示例。进一步结合图12和图13描述了训练的示例。每个扭曲眼睛图像642被输入到子网络。这些图像642可被协同输入(例如,并行地输入两个图像)或者相互独立地输入(例如,一次输入一个图像,其中第一子网络将从该图像预测注视方向)。作为响应,子网络产生每个眼睛的2D注视向量654(例如,对应于输入图像中所示的用户眼睛)。每个2D注视向量可被表示为2D注视原点 o_{2D} (例如,在图像平面内的用户眼睛2D位置)以及2D注视方向 d_{2D} 。脸部扭曲图像644也被输入到子网络,其作为响应给出距离校正 c_{652} 。因此,来自神经网络的输出包括五个部分:每只眼睛的2D注视原点 o_{2D} 和2D注视方向 d_{2D} ,以及距离校

正c。

[0055] 距离校正652、估算距离632和2D注视向量654被输入到3D注视发生器660。作为响应,3D注视发生器660产生并输出3D注视662。在一个示例中,3D注视662包括用户每只眼睛的3D注视方向(它可以被表达在3D真实空间内)以及用户眼睛的3D位置(它也可以被表达在3D真实空间内)。

[0056] 为了说明,经校正的距离 d_{corr} 被产生为 $d_{\text{corr}} = d_{\text{rough}} \times c$ 。参照眼睛之一及其2D注视向量,3D注视发生器660基于校正的距离和2D-3D空间映射(例如,相关矩阵)来计算其3D注视。例如,归一化3D空间内的3D注视原点 $o_{3D,N}$ 被计算为 $o'_{3D,N} = M^{-1} o_{2L}$ 并且 $o_{3D,N} = o'_{3D,N} / ||o'_{3D,N}|| d_{\text{corr}}$ 。为了产生归一化3D空间内的3D注视方向,首先产生归一化基础向量(X,Y,Z),其中 $Z' = M^{-1} o_{2D}$, $Y' = [0, 1, 0]^T \times Z'$ 并且 $X' = Y' \times Z'$,且 $X = X' / ||X'||$, $Y = Y' / ||Y'||$ 且 $Z = Z' / ||Z'||$ 。归一化的3D注视方向 $d_{3D,N}$ 被产生为 $d'_{3D,N} = [X, Y] d_{2D} - Z$ 和 $d_{3D,N} = d'_{3D,N} / ||d'_{3D,N}||$ 。基于旋转矩阵R将3D注视原点和3D注视方向从归一化3D空间映射至真实3D空间。例如,真实3D空间内的3D注视原点 o_{3D} 被计算为 $o_{3D} = R^{-1} o_{3D,N}$ 。同样,真实3D空间内的3D注视方向 d_{3D} 被计算为 $d_{3D} = R^{-1} d_{3D,N}$ 。

[0057] 图7示出根据一个实施例的神经网络的示例网络架构。如图7所示,三个图像710-714被输入到网络:低分辨率下的脸部扭曲图像710、高分辨率下的左眼扭曲图像712以及高分辨率下的右眼扭曲图像714(其中,“低”和“高”是相对术语,例如“低分辨率”表示低于“高分辨率”)。眼睛扭曲图像712和714在眼睛检测上居中 $x - e_{\text{left or right}}$,被缩放至 $s = 320$ 像素的双眼间距离并被剪裁至 224×112 像素。通过修正旋转矩阵R将右眼图像714镜像化716。这为网络提供了一致的外观。脸部扭曲图像710被居中在眼睛检测之间的中点,被缩放至等于84像素并被剪裁至 224×56 像素。

[0058] 网络包括对于眼睛和脸部的不同的卷积神经网络(CNN) 720-726(图示为包括对于左眼的CNN 722和对于右眼的CNN 724,其具有同等的权重,并且包括对于脸部的CNN 720)。两者均是ResNet-18的卷积部分,其类似于K.He, X.Zhang, S.Ren, 和J.Sun在CoRR, abs/1512.03385, 2015发表的“对于图像识别的深度残留学习(Deep residual learning for image recognition)”中所描述的,该文献的内容援引包含于此。来自眼睛和脸部的所有CNN 720-726的输出被级联为级联层730并且被馈送至完全连接的模块740,该完全连接的模块740预测距离校正c。

[0059] 来自每个眼睛CNN 722和CNN 724的输出与一组“n个”个人校准参数和距离校正c级联在一起。左眼和右眼的级联分别是通过级联层732和级联层734实现的。源自每个级联的组合特征向量被馈送至完全连接的模块(其被图示为针对左眼的完全连接的模块742和针对右眼的完全连接的模块744)。

[0060] 完全连接的模块740-744可以被描述为:FC(3072)-BN-ReLU-D0(0.5)-FC(3072)-BN-ReLU-D0(0.5)-FC({4,1})。最终输出或者是2D注视原点和2D注视方向,或者是距离校正c。

[0061] 不需要将“n个”校正参数提供给距离估算部件(例如,级联层),因为从在一个距离上采集的校准数据(这是通常可用的)检测距离误差一般是有难度的。

[0062] 网络被训练以使预测的注视线与基础-真实刺激点之间的平均最小错开距离最小化(如接下来的附图所示)。落在眼睛图像之外的注视原点和在任何方向上超过40%的距离

校正也被惩罚。

[0063] 图8示出根据一个实施例的示例性图像归一化。如图所示,真实3D空间810以产生真实图像830的相机为中心。3D空间810可被定义为3D坐标系。归一化的3D空间820也可以相机为中心并且基于旋转矩阵R从真实3D空间810产生。例如,归一化3D空间820对应于真实3D空间810沿真实3D空间810的其中一根轴线(例如,Y轴)的旋转。用户双眼之间的中心形成一条直线。该直线平行于归一化3D空间820中的一根轴线(例如,X轴)。归一化图像840(被图示为对应于左眼扭曲图像)从真实图像830被投影到归一化3D空间内(被图示为与真实图像830相距一定距离并与之平行)。在该归一化图像840对应于两个眼睛之间的双眼间距离是63mm的人类平均ID的位置的情况下,相机与归一化图像840中的眼睛的中心之间的距离为粗略距离。

[0064] 图9示出根据一个实施例的示例性3D注视预测。如图所示,使用图8的归一化3D空间820,虽然该预测也可通过使用结合图6描述的旋转矩阵R的逆矩阵被映射到真实3D空间810。3D注视原点910是基于缩放矩阵M的逆矩阵和经校正的距离930从2D注视原点920推导出的。2D注视原点920和2D注视方向940表示归一化图像840的平面内的2D注视向量。3D注视方向950是基于结合图6描述的缩放矩阵M的逆矩阵从2D注视原点920和2D注视方向940推导出的。

[0065] 在训练期间,3D注视线(或3D注视射线)沿着3D注视方向950从3D注视原点910投射。注视的刺激点960是已知的(例如,已知的注视点)。刺激点960和3D注视线之间的距离970(例如,最短距离)是损失函数的距离参数。通过使该距离970最小化来使损失函数最小化(例如,如果神经网络正确地预测3D注视,则距离970可以为零并且刺激点960将落在3D注视线上)。

[0066] 图10示出根据一个实施例的、由神经网络产生的对2D注视向量和距离校正的预测。图10示出被输入到神经网络的三个图像1010、1020、1030。图像1010、1020是眼睛扭曲图像,而图像1030是脸部扭曲图像。此外,(例如,左眼的)图像1020被镜像化以使两个图像1010、1020对准(例如,左眼的目内眦与右眼的目内眦对准,并且左眼的目外眦与右眼的目外眦对准)。

[0067] 预测的2D注视向量被覆盖在图像1010、1020之上。如图像1010的覆盖图像所示(并且类似地,如图像1020的覆盖图像所示),2D注视向量具有注视原点1012(例如,右眼上的闪光点)和注视方向1014。经估算和校正的距离1032被覆盖在图像1030中。如图所示,经估算的距离是653毫米并且经校正的距离为475毫米,这代表大约72%的预测距离校正。

[0068] 图11示出根据一个实施例的、基于例如深度学习系统600的深度学习系统预测3D注视的示例性流程。眼睛跟踪系统被描述为执行示例性流程的操作。在一个示例中,眼睛跟踪系统宿留有所述深度学习系统。在另一个示例中,远程计算机系统宿留有深度学习系统,并且眼睛跟踪系统通过数据网络与该远程计算机系统交互以提供2D图像并接收3D注视预测。在又一示例中,深度学习系统被分布在眼睛跟踪系统和远程计算机系统之间(例如,远程计算机系统可宿留有神经网络,而眼睛跟踪系统可宿留有深度学习系统的其余部分)。

[0069] 执行示例流程的操作的指令可作为计算机可读指令被存储在眼睛跟踪系统的非暂时性计算机可读介质上。一旦被存储,指令即代表可编程模块,该可编程模块包括可由眼睛跟踪系统的(一个或多个)处理器执行的代码或数据。这些指令的执行将眼睛跟踪系统配

置成执行附图中所示和本文描述的具体操作。每个可编程模块与处理器结合以表示执行(一个或多个)相应操作的手段。尽管这些操作是按照特定顺序示出的,但应当理解,不需要有特定顺序,并且可省去、跳过和/或重新排列一个或多个操作。

[0070] 示例性流程开始于操作1102,其中,眼睛跟踪系统接收图像(例如,2D图像)。在一个示例中,通过与眼睛跟踪系统关联的相机(例如,与该系统一体形成或与该系统接口连接的相机)产生图像。该图像示出用户眼睛(例如,当相机与用户的两只眼睛关联时,该图像示出两只眼睛;当相机比如在VR头戴设备的情况下与用户的一只眼睛关联时,该图像仅示出用户的那只眼睛)。

[0071] 在操作1104,眼睛跟踪系统从该图像产生在用户眼睛周围的扭曲图像。例如,眼睛跟踪系统在图像中检测用户眼睛并且估算相机与在图像中检测到的用户眼睛之间的粗略距离。眼睛跟踪系统还基于相机的固有矩阵 K 将图像投影到3D空间内,基于旋转矩阵 R 旋转所投影的图像,并基于缩放矩阵 M 将旋转后的图像缩放。缩放矩阵 M 是基于像素中的预定义距离 s 和人的平均双眼间距离而产生的。眼睛跟踪系统然后使用双线性插值法使缩放后的图像扭曲,并基于预定义的区域(例如,具有预定义宽度和高度的区域)在用户眼睛周围对扭曲的图像进行剪裁。该图像具有依赖于预定义距离 s 的第一图像分辨率。

[0072] 在操作1106,眼睛跟踪系统从该图像产生第二用户眼睛周围的第二扭曲图像。如果图像展示第二用户眼睛,则执行该操作,否则就跳过该操作。操作1106类似于操作1104。第二扭曲图像展示第二用户眼睛并且具有第一图像分辨率。

[0073] 在操作1108,眼睛跟踪系统从该图像产生用户周围的第三扭曲图像。如果图像展示第二用户眼睛,则执行该操作,否则就跳过该操作。操作1106类似于操作1104,其中基于第二预定义距离 s (并因此第二缩放矩阵 M)来进行缩放,以取得第二图像分辨率。第三扭曲图像展示用户脸部或至少用户的两只眼睛,并具有低于第一图像分辨率的第二图像分辨率。

[0074] 在操作1110,眼睛跟踪系统将用户眼睛的扭曲图像输入到神经网络。如果产生另外两个扭曲图像(例如,第二用户眼睛的扭曲图像和用户脸部的扭曲图像),它们也被输入到神经网络。神经网络已被训练并预测:(i)距离校正 c ,和(ii)在相关联的扭曲眼睛图像中每只眼睛的2D注视原点和2D注视方向。总的来说,距离校正 c 被用来校正所估算的粗略距离,并基于扭曲图像以及(如果有的话)第二和第三扭曲图像来预测该距离校正 c 。用户眼睛(同样,第二用户眼睛)的2D注视原点和2D注视方向是从与第二和第三扭曲图像不同的扭曲图像产生的(同样,从与该扭曲图像和第三扭曲图像不同的第二扭曲图像产生的)。

[0075] 在一个示例中,神经网络也基于内嵌在神经网络中的“ n 个”校准参数而被训练。在眼睛跟踪系统的校准期间,通过指示用户注视数个已知的注视点,该系统产生多个校准图像。这些校准图像也被归一化和扭曲,并被输入到神经网络中。调整“ n 个”校准参数以使神经网络的损失函数最小化。该最小化使用已知的注视点作为基础真实(ground truth)。一旦图像在操作1102由眼睛跟踪系统接收并在操作1106-1108被处理,则神经网络在操作1110的预测使用“ n 个”校准参数。

[0076] 在操作1112,眼睛跟踪系统从神经网络接收距离校正 c 。眼睛跟踪系统也从神经网络接收用户眼睛的2D注视原点和2D注视方向,并在适合的情况下接收第二用户眼睛的2D注视原点和2D注视方向。

[0077] 在操作1114,眼睛跟踪系统至少通过基于距离校正 c 更新所估算的粗略距离而产生用户眼睛和相机之间的校正的距离。例如,距离校正 c 是校正因数,并且眼睛跟踪系统将估算的粗略距离乘以该因数以产生校正的距离。

[0078] 在操作1116,眼睛跟踪系统从校正的距离及其2D注视原点和2D注视方向来产生用户眼睛的3D注视信息。3D注视信息包括用户眼睛在3D空间内的位置(例如,3D注视原点)和从用户眼睛起始的3D注视方向。在一个示例中,3D注视原点和3D注视方向是基于缩放矩阵 M 和旋转矩阵 R (或它们的逆矩阵)从校正的距离、2D注视原点和2D注视方向推导出的。也可重复这种操作以基于校正的距离及其2D注视原点和2D注视方向来产生第二用户眼睛的3D注视原点和3D注视方向。在这种情况下,3D注视信息包括用户的两只眼睛的3D注视原点和3D注视方向。

[0079] 在操作1120,眼睛跟踪系统向3D注视应用提供为用户眼睛产生的3D注视信息,并在适当的情况下提供为第二用户眼睛产生的3D注视信息。3D注视应用使用3D注视信息来支持立体显示和3D应用(对屏幕上所显示的对象3D控制和操纵)。

[0080] 基于训练图像来训练深度学习系统的神经网络以预测距离校正和每个用户眼睛的2D注视向量(例如,2D注视原点和2D注视方向)。总的来说,对于这些训练图像通过反向传播(例如,使用梯度下降的反向传播)重复训练以使损失函数最小化并因此更新神经网络的参数。由于神经网络应当预测两个输出(距离校正和2D注视向量),因此仅靠展示注视着捕获训练图像的关联相机的2D平面内的注视点时的用户眼睛的训练图像是不够的。相反,为了正确训练,注视点的位置相对于2D平面的多样性是必需的。结合图12进一步描述这种训练的问题和解决方案。

[0081] 图12示出根据一个实施例的使用训练图像的策略,该训练图像与注视点位置相对于产生训练图像的相机的2D平面的多样性相关联。为了便于解释清楚,结合图12讨论单个用户眼睛。然而,展示两个用户眼睛的训练图像是等同适用的。在训练图像中捕捉的用户眼睛在本文中也称为“所展示的用户眼睛”。用来产生训练图像的相机在本文中也称为“训练相机”。如图所示,相机1210产生展示用户眼睛的训练图像。每个训练图像是在用户眼睛注视着注视点时捕捉的2D图像。

[0082] 例如,第一训练图像示出在用户注视着注视点1220时的用户。如果第一训练输入是在神经网络训练期间被送至神经网络的输入,则神经网络可能不正确地找到用户眼睛到注视点1220的注视方向以及用户眼睛到相机1210的距离。如图12所示,对于注视方向和距离可以有多个解(图12示出了这些解中的两个解)。在第一解中,神经网络可预测用户眼睛(在该解中被图示为要素1230)正相对于注视点1220以第一角度“ a_1 ”1232注视,并且与相机1210相隔第一距离“ d_1 ”1234。在第二种解中,神经网络可预测用户眼睛(在该解中被图示为要素1240)正相对于注视点1220以第二角度“ a_2 ”1242注视,并且与相机1210相隔第二距离“ d_2 ”1244。两个解都是可能的,但只有一个是正确的。因此,神经网络可能预测不正确的解,由此训练可能不正确并导致经训练的神经网络产生不准确的预测。

[0083] 为了避免这种可能性,使用注视点位置多样性的策略。在该策略下,额外地使用用户眼睛的第二训练图像。在该第二训练图像中,或是注视角或是用户眼睛至相机距离有改变。因此,当使用两个训练图像时,一组可能的解被缩减至单个解(例如,要么是上例中的第一个解,要么是上例中的第二个解)。

[0084] 许多不同的方法可实现这种多样性。总的来说,训练图像可包括两组训练图像。第一组训练图像是正在注视着相机的2D平面内的注视点的用户眼睛的训练图像。第二组训练图像是正在注视着相机的2D平面外的注视点的用户眼睛的训练图像。如此,当各种训练图像被输入到神经网络时,神经网络学着寻找正确的解。

[0085] 图12示出一种具体方式。如图所示,在用户眼睛沿着与第一训练图像中相同的注视角注视第二注视点1250时,产生用户眼睛的第二训练图像。如此,唯一可能的解是第一个解(例如,第一角度“a1”1232和第一距离“d1”1234)。另一种方法是对第二训练图像使用相同的注视点1220,但改变用户眼睛与相机1210之间的距离(例如,通过在用户眼睛保持注视点1220时移动相机以使其更靠近或更远离用户眼睛)。又一种方法是保持相同的用户眼睛至相机距离,引入第二注视点(不一定在与图12所示的具体方法中相同的注视角上)并在用户眼睛注视着第二注视点时产生第二训练图像。当然,可使用这些不同方法的任意组合来针对不同用户眼睛、不同注视角和不同用户眼睛至相机距离产生多个训练图像。可以——但不一定需要——使用相同相机(或对多个相机的相同相机配置)来产生训练图像,因为训练不依赖于具体的相机或屏幕配置。

[0086] 一旦产生训练图像,它们被输入到神经网络以用于训练。具体地,神经网络从这些训练图像预测注视角以及用户眼睛至相机距离。可基于注视角和用户眼睛至相机距离来定义损失函数。为清楚起见,这里考虑上文所述的第一训练图像(例如,当用户眼睛正注视着注视点1220时产生的第一训练图像)。基于该训练图像(以及第二训练图像),神经网络预测第一解(例如,第一角“a1”1232和第一距离“d1”1234)。让我们假设第一角“a1”1232和第一距离“d1”1234是实际的注视角和距离,这些是训练中已知的(例如,它们是可以被存储在与第一训练图像相关联的训练标签中的基础真实测量值)。让我们也假设预测不完全准确(例如,预测的角与第一角“a1”1232有偏差,并且预测的距离与第一距离“d1”1234有偏差)。损失函数包括距离项和角度项。距离项是预测的距离与基础真实距离(例如,实际的第一距离“d1”1234)之间的差。角度项是预测的注视角与基础真实注视角(例如,实际的第一角“a1”1232)之间的差。训练的目的在于更新神经网络的参数以使其损失函数最小化,其中,使损失函数最小化包括最小化角度项和距离项,以使预测的角度尽可能接近基础真实角并且使预测的距离尽可能接近基础真实距离。

[0087] 在一个示例中,可以用单个距离项来代替前述角度项和距离项。例如,预测的注视线产生在离相机的预测距离处并且具有预测的注视角。测量注视点1220和预测的注视线之间的距离(例如,最短距离)(这在图9中被图示为距离970)。如果该预测是完全准确的,那么注视点至预测的注视线的距离将为零。然而在其他情形下,通过最小化注视点至预测的注视线的距离以使预测的注视线尽可能接近注视点1220,损失函数被最小化(并且神经网络参数被更新)。

[0088] 在该示例中,损失函数也可包括惩罚项。更具体地说,当预测的注视线的原点落在对应的训练图像之外时(回过头来参照前面的示例,如果“d1”1234足够大以至于眼睛1230落在第一训练图像之外),则将惩罚加至损失函数(例如,惩罚具有预定义的值)。同样,如果预测的距离校正任一方向上超过某一阈值(例如,在任一方向上超过40%),则将相同的惩罚或另一惩罚加至损失函数。

[0089] 图13示出根据一个实施例的用于训练神经网络的示例流程。图14还示出训练的额

外示例流程,其中,这种训练根据一个实施例使用内嵌式校准参数。计算机系统被描述为执行示例性流程的操作。在一个示例中,计算机系统执行训练并存储神经网络的代码。一旦训练结束,计算机系统可从眼睛跟踪系统接收图像并使用神经网络来以3D注视预测作出响应。作为附加或替代地,眼睛跟踪系统可从计算机系统下载神经网络的代码。

[0090] 用于执行所示流程的操作的指令可作为计算机可读指令被存储在计算机系统的非暂时性计算机可读介质上。一旦被存储,指令代表可编程模块,该可编程模块包括可由计算机系统的(一个或多个)处理器执行的代码或数据。这些指令的执行配置眼睛计算机系统以使其执行附图所示和本文描述的具体操作。每个可编程模块与处理器结合以表示执行(一个或多个)相应操作的手段。尽管这些操作是按照特定顺序阐述的,但应当理解,不需要该特定顺序,并且可省去、跳过和/或重新排列一个或多个操作。

[0091] 图13的示例性流程开始于操作1302,其中计算机系统访问训练图像,该训练图像包括第一组训练图像和第二组训练图像。第一组训练图像中的一些或全部展示与相机平面内的注视点相关联的用户眼睛。第二组训练图像中的一些或全部展示与相机平面之外的注视点相关联的用户眼睛。如此,取得训练图像的多样性,训练图像示出用户眼睛注视着该平面之内和之外的注视点。

[0092] 在操作1304,计算机系统基于训练图像来训练神经网络。总的来说,训练包括更新神经网络的参数(例如,对神经网络的各个层的节点之间的连接的权重)以使神经网络的损失函数最小化。这种损失函数可使用前面结合图12解释的角度项和/或距离项。在一个示例中,操作1304包括两个额外的操作1306、1308。

[0093] 在操作1306,计算机系统将来自训练图像的第一训练图像和第二训练图像输入至神经网络。第一训练图像属于第一组训练图像。第二训练图像属于第二组训练图像。在一个具体示例中,第一训练图像展示在用户眼睛以一注视角注视着注视点时的用户眼睛,而第二训练图像展示在用户眼睛以该注视角注视着另一注视点时的用户眼睛。在另一具体示例中,第一训练图像和第二训练图像展示在用户眼睛以一注视角注视着该注视点时的用户眼睛。在该示例中,第一训练图像对应于相机与用户眼睛之间的第一距离,而第二训练图像对应于相机与用户眼睛之间的第二距离。在又一具体示例中,第一训练图像展示在用户眼睛以一注视角注视着该注视点时的用户眼睛,而第二训练图像展示在用户眼睛以不同的注视角注视着另一注视点时的用户眼睛。在这些示例中,第一和第二训练图像可成对地输入。

[0094] 如结合图13的流程在这里使用的,训练图像表示一组训练图像,其包括展示人的眼睛的扭曲图像,以及可选择地包括展示人的另一只眼睛的扭曲图像和人脸的扭曲图像。换句话说,训练中使用的图像是在训练结束时将被使用的相同类型的图像(如结合图9所阐述的),并且可以根据投影、旋转、缩放和剪裁等相同操作而产生。

[0095] 在操作1308,计算机系统基于注视点与注视线之间的距离而使神经网络的损失函数最小化。注视点关联于第一训练图像或第二训练图像中的一个。用户眼睛的注视线是由神经网络从第一训练图像或第二训练图像中的该一个预测出的。对许多训练图像(如果不是全部训练图像)重复操作1306-1308以完成神经网络的训练。

[0096] 转向图14,神经网络的训练也可包含校准参数,从而当神经网络预测注视信息时(如前所述),预测基于这些参数。神经网络和/或眼睛跟踪系统的设计者不需要指定校准参数。相反,校准参数是内嵌的并且设计者只需要指定这些参数的数量。在训练过程中,校准

参数被初始化,并随后基于使神经网络的损失函数最小化而将校准参数连同神经网络的参数(例如,网络中的连接之间的权重)一起更新。如此,在训练期间也基于校准参数来更新神经网络的参数。在神经网络的训练期间,图14的流程中的某些操作可以结合图13的操作被实现和使用。一旦训练结束,图14的其余操作可以结合图11的操作被实现和使用。在这种情形下,对于给定的一组校准图像,基于损失函数的最小化来设定用户的校准参数。这些校准参数然后被输入到神经网络以支持3D注视预测。

[0097] 尽管图14描述了在预测注视信息的神经网络的背景下对校准参数进行学习,但实施例不局限于此。总的来说,图14的操作同样适用于与系统相关并为某个任务(例如,预测特定结果)进行训练的神经网络,其中,系统的正确操作包含校准。此外,为了便于解释清楚,图14的示例性流程是结合一个训练图像、一个校准图像和一个图像来描述的。然而,该示例性流程同样适用于多个训练图像、多个校准图像以及多个图像。

[0098] 如图所示,图14的示例性流程开始于操作1402,其中,计算机系统访问与某人关联的训练图像。训练图像展示此人的脸部和/或(一个或两个)眼睛。总的来说,此人具有索引号“i”并且可用索引号“i”对训练图像进行索引(例如,训练图像的标签或元数据包括索引号“i”)。训练图像可从数据存储器获得,并从中获取以作为训练过程的一部分。数值“n”是神经网络的设计者指定的大于零的整数。在一个示例中,数值“n”在2和10之间。在一个具体示例中,数值“n”为3。

[0099] 在操作1404,计算机系统初始化用于此人的“n个”校准参数。例如,这些校准参数的值被设定为零或某个其他的第一值。

[0100] 在操作1406,计算机系统将训练图像和“n个”校准参数输入至神经网络。例如,训练图像被输入至神经网络的相关Res18 CNN,而“n个”校准参数被输入至与神经网络的完全连接的模块相关联的级联层,其中,这些模块负责预测2D注视原点和2D注视方向。

[0101] 在操作1408,计算机系统基于使神经网络的损失函数最小化而更新“n个”校准参数和神经网络的参数(在本文中被称为网络参数,并包括例如神经网络的节点之间的连接的权重)。通过反向传播基于训练图像和“n个”校准参数来最小化损失函数。因此,“n个”校准参数的第一值(例如,零)被更新至第二值。

[0102] 操作1402-1408可通过计算机系统作为对神经网络进行训练的一部分实现。一旦训练结束,神经网络可供眼睛跟踪系统的用户使用。当使用眼睛跟踪系统时,可通过专门针对用户校准神经网络来校准该系统。

[0103] 在操作1410,计算机系统将针对用户的“n个”校准参数初始化。该操作类似于操作1404。在一个示例中,初始化可以是——但不一定需要——初始化至相同的第一值(例如,零)。在另一示例中,初始化是初始化至“n个”校准参数的更新值(例如,第二值),该更新值是在操作1408确定的。

[0104] 在操作1412,计算机系统产生用户的校准图像。例如,计算机系统指示用户注视已知的注视点并因此产生校准图像。基于与眼睛跟踪系统相关联的相机产生的图像数据,该校准图像展示用户的一只眼睛(或在适合的情况下,展示用户的双眼或脸部)。

[0105] 在操作1414,计算机系统将校准图像和校准参数输入至神经网络。此操作类似于操作1406。

[0106] 在操作1416,计算机系统更新针对用户的“n个”校准参数而不更新在训练期间设

定的网络参数。至少通过基于校准图像和“n个”校准参数使损失函数最小化来更新“n个”校准参数,其中,最小化不改变网络参数。这里,校准图像具有已知的校准点。该已知的校准点在损失函数的最小化中作为用户注视的基础真实而使用。计算机系统通过反向传播在保持网络参数不变的同时重复地更新“n个”校准参数,直到损失函数被最小化为止。因此,在操作1410执行的对“n个”校准参数进行初始化的值(例如,第一值)被更新至第三值。一般来说,该第三值与在操作1408确定的“n个”校准参数的第二值不同。

[0107] 操作1410-1416可通过计算机系统作为校准神经网络的一部分来实现。一旦校准结束,则神经网络可供使用以产生用户的3D注视信息。

[0108] 在操作1418,计算机系统将用户的图像和更新的“n个”校准参数输入至神经网络。基于由相机产生的额外图像数据,图像展示用户的一个眼睛(在合适的情况下,展示用户的双眼或脸部)。此操作类似于操作1406。

[0109] 在操作1420,计算机系统从所述神经网络接收基于图像和“n个”校准参数的预测。该预测包括距离校正、图像中的用户眼睛的2D注视原点以及图像中的用户眼睛的2D注视方向。

[0110] 为了清楚和便于理解,已详细描述了本公开。然而将理解,在所附权利要求书的范围内可以作出某些变化和修正。

[0111] 上面的描述仅提供了示例性实施例,并且不旨在对本公开的范围、适用性或配置构成限制。相反,上面对示例性实施例的描述将为本领域技术人员提供用于实现一个或多个示例性实施例的可行性描述。要理解,可在不脱离如本文所述的本公开的精神和范围的情况下对各个要素的功能和配置作出各种改变。

[0112] 例如,针对一个实施例讨论的任何细节可以出现在或者可以不出现在该实施例的所有预想版本中。同样,针对一个实施例讨论的任何细节可以出现在或者可以不出现在本文描述的其他实施例的所有预想版本中。最后,没有讨论的关于本文中的实施例的任何细节应当隐含地表示该细节可以出现在或者可以不出现在本文讨论的任何实施例的任何版本中。

[0113] 上面的描述中给出了具体细节以提供对实施例的透彻理解。然而,本领域技术人员将理解,这些实施例没有这些具体细节也能实施。例如,为了不因不必要的细节而使实施例变得晦涩难懂,本公开中的电路、系统、网络、过程和其他要素可表示成方框图形式的部件。在其他情况下,为了避免使实施例变得晦涩难懂,省略不必要的细节地示出公知的电路、过程、算法、结构和技术。

[0114] 另外要注意,各实施例可被描述为过程,所述过程被表述为流程图、流程框图、数据流程图、结构图或方框图。尽管流程图可能将操作描述为顺序过程,但许多操作可并行地或同时地执行。另外,可以重新安排操作的顺序。当过程的操作结束时,过程可终止,但该过程可以具有未讨论和不包括在附图中的额外步骤。此外,在任意具体描述的过程中的操作并非全部都要出现在所有实施例中。过程可对应于方法、函数、进程、子例程或子程序等。当过程对应于函数时,过程的结束对应于将函数返回到调用函数或主函数。

[0115] 术语“机器可读介质”包括但不限于暂时和非暂时的、便携的或固定的存储设备、光学存储设备、无线信道和能够存储、包含或携带一个或多个指令和/或数据的各种其他介质。代码段或机器可执行指令可代表进程、函数、子程序、程序、例程、子例程、模块、软件包、

类,或指令、数据结构或程序语句的任意组合。代码段可通过传送和/或接收信息、数据、自变量、参数或存储器内容而耦接到另一代码段或硬件电路。信息、自变量、参数、数据等可以经由任何合适的手段传递、转发或发送,所述手段包括存储器共享、消息传递、令牌传递、网络传输等。

[0116] 此外,本公开的实施例可至少部分地手动或自动地实现。可通过使用机器、硬件、软件、固件、中间件、微代码、硬件描述语言或其任意组合来执行手动或自动实施方式,或至少对手动或自动实施方式作出辅助。当以软件、固件、中间件或微代码实施时,用于执行必要任务的程序代码或代码段可被存储在机器可读介质中。一个或多个处理器可执行必要任务。

[0117] 如本文所使用的。短语“基于第二件事做第一件事”等可表示第一件事仅基于第二件事,或者表示第一件事基于第二件事以及一件或多件额外的事。

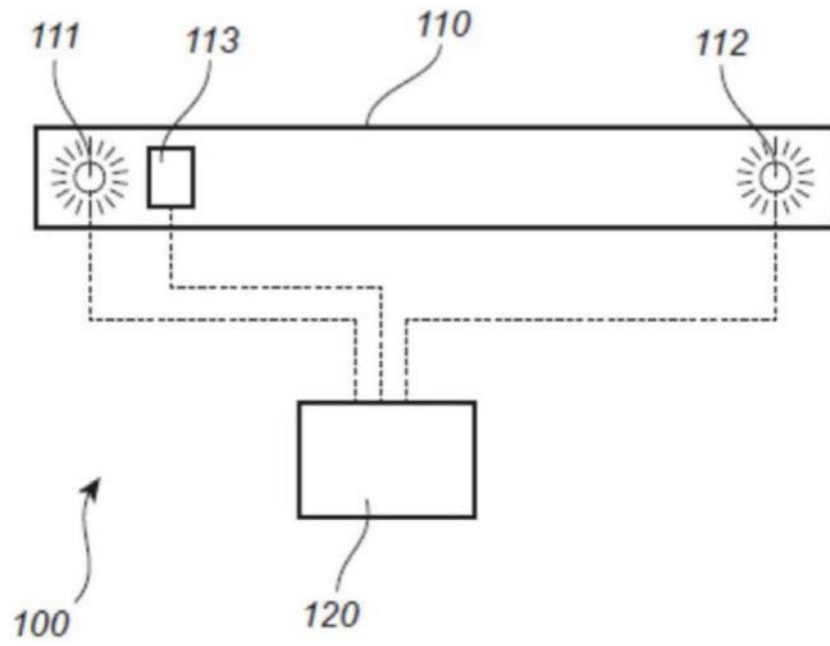


图1

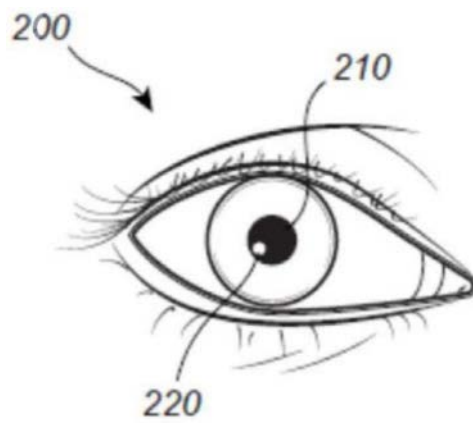


图2

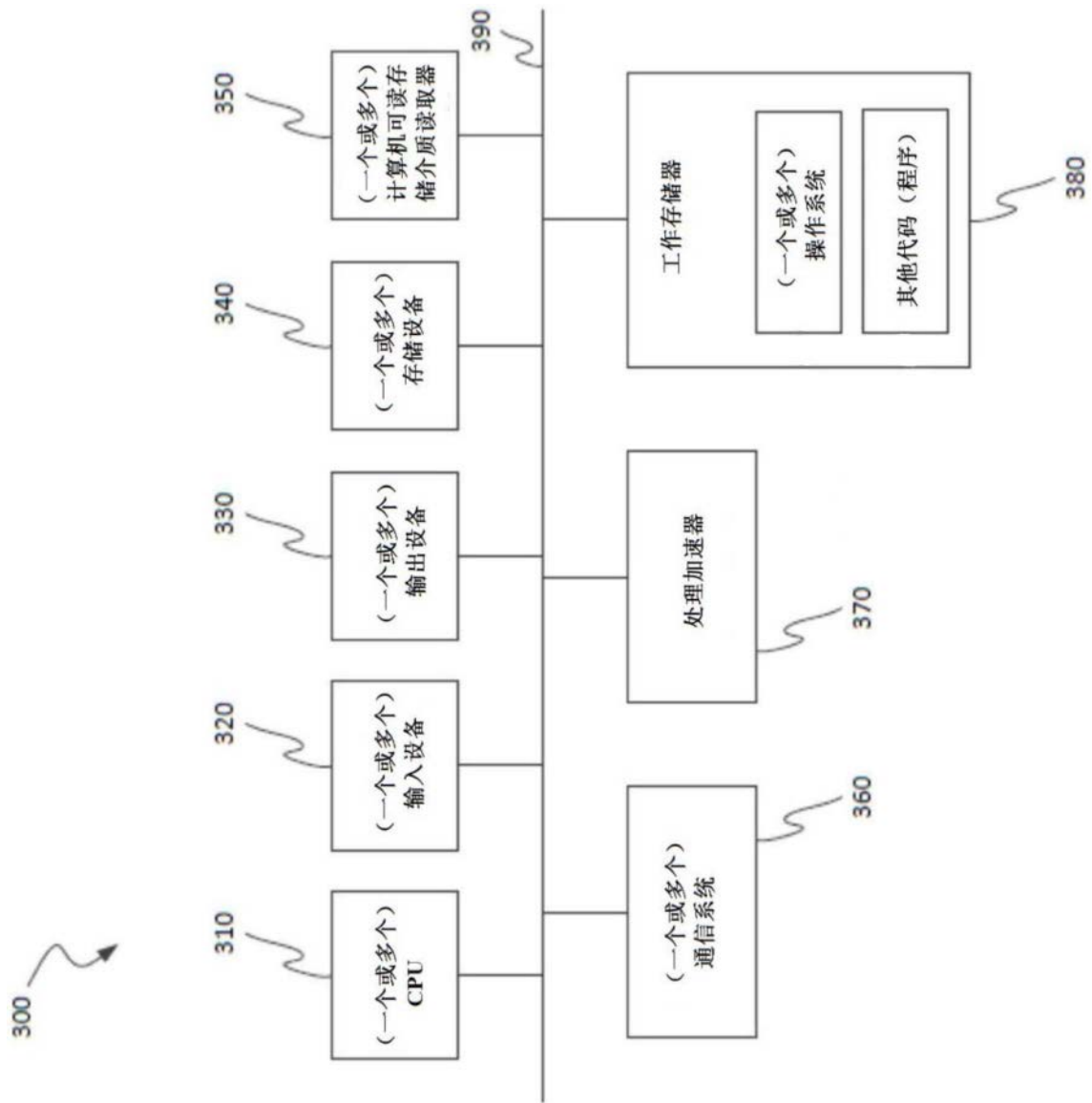


图3

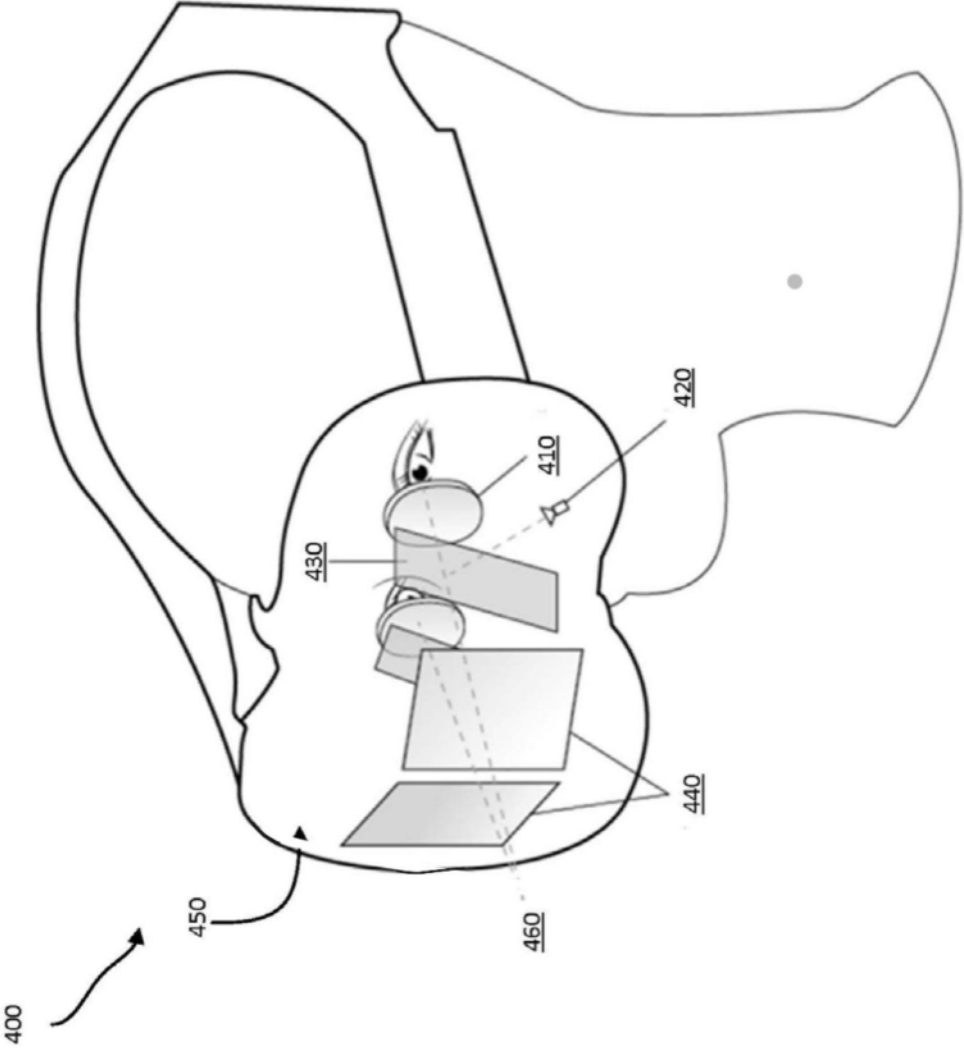


图4

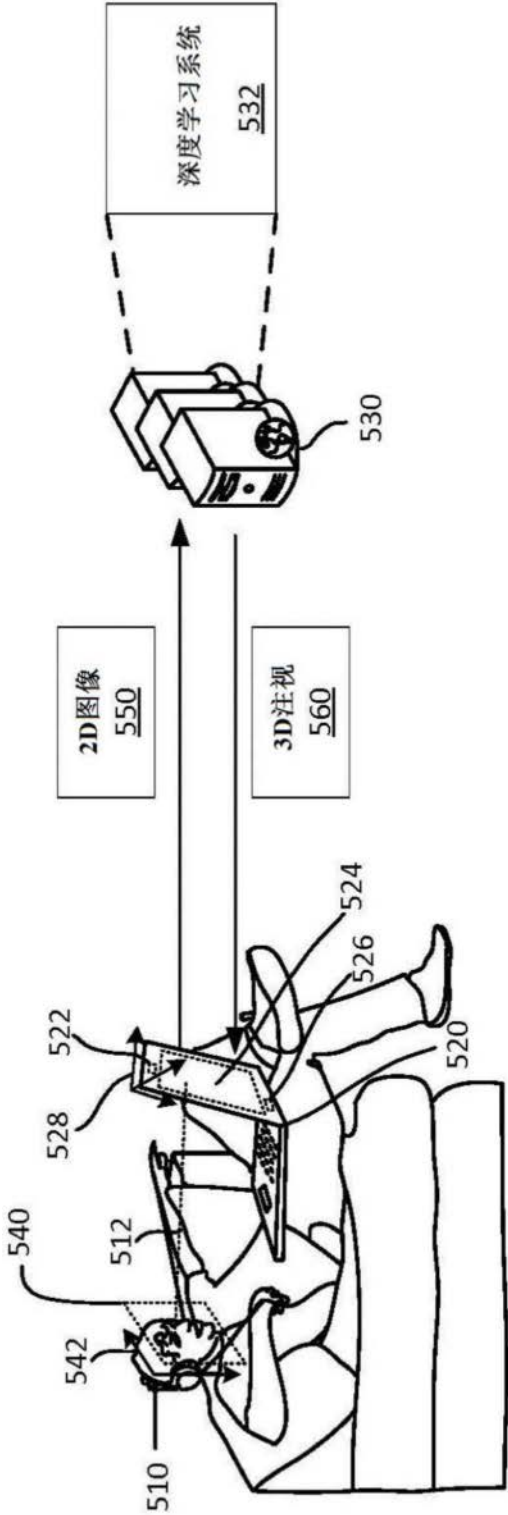


图5

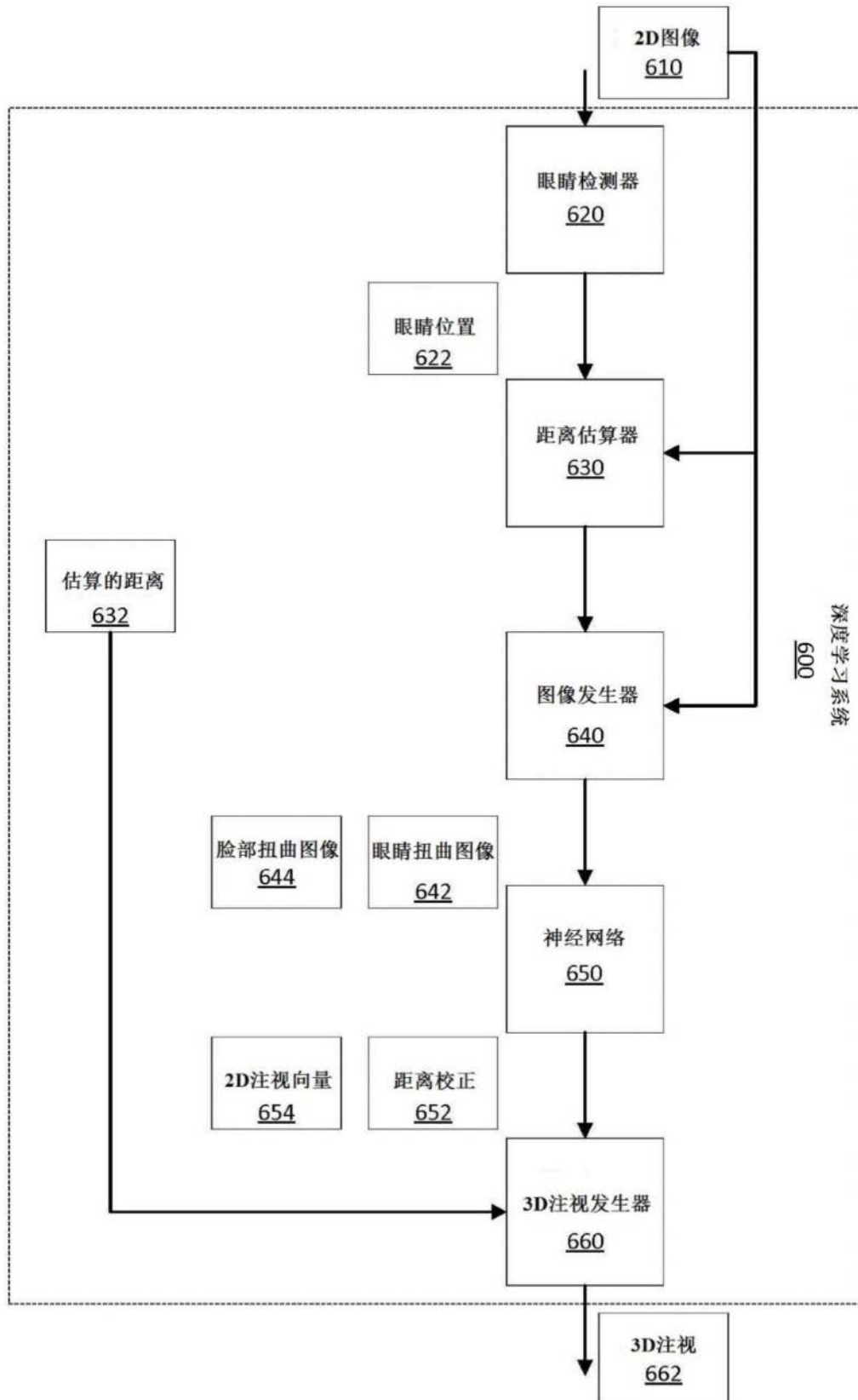


图6

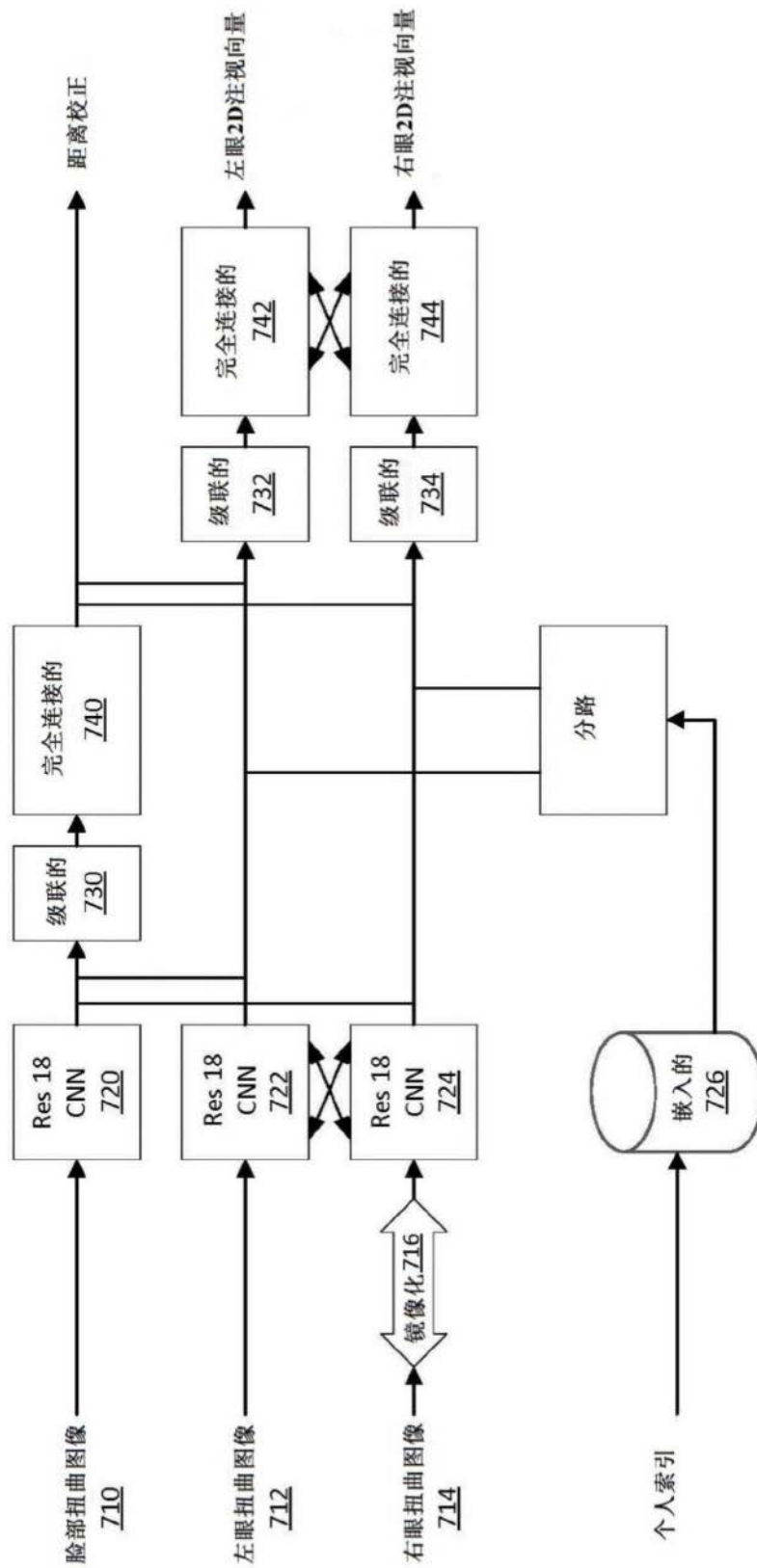


图7

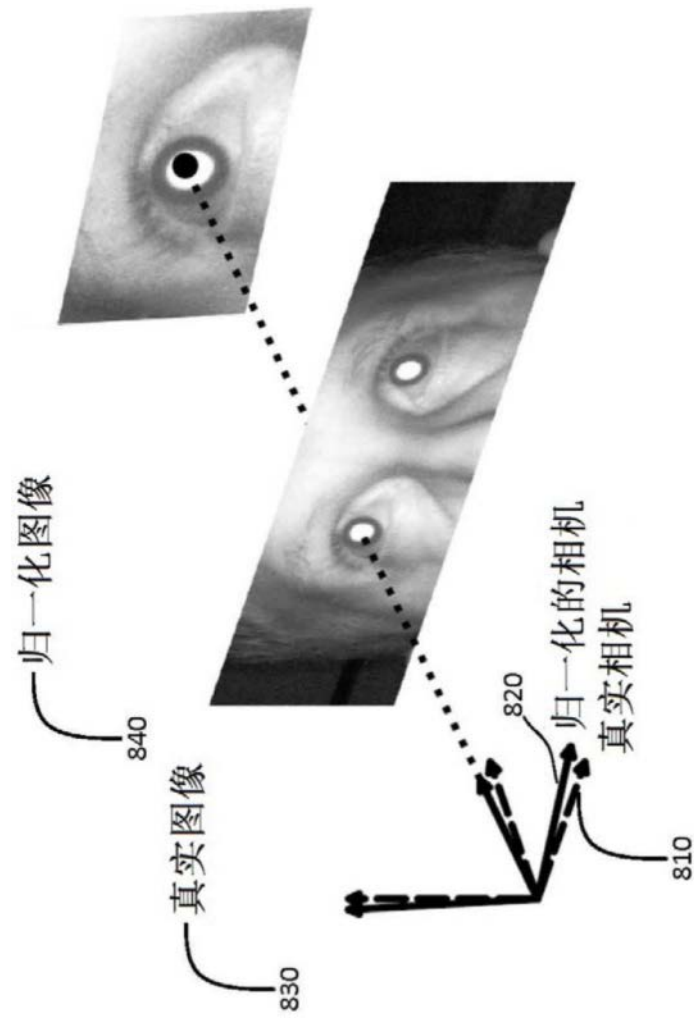


图8

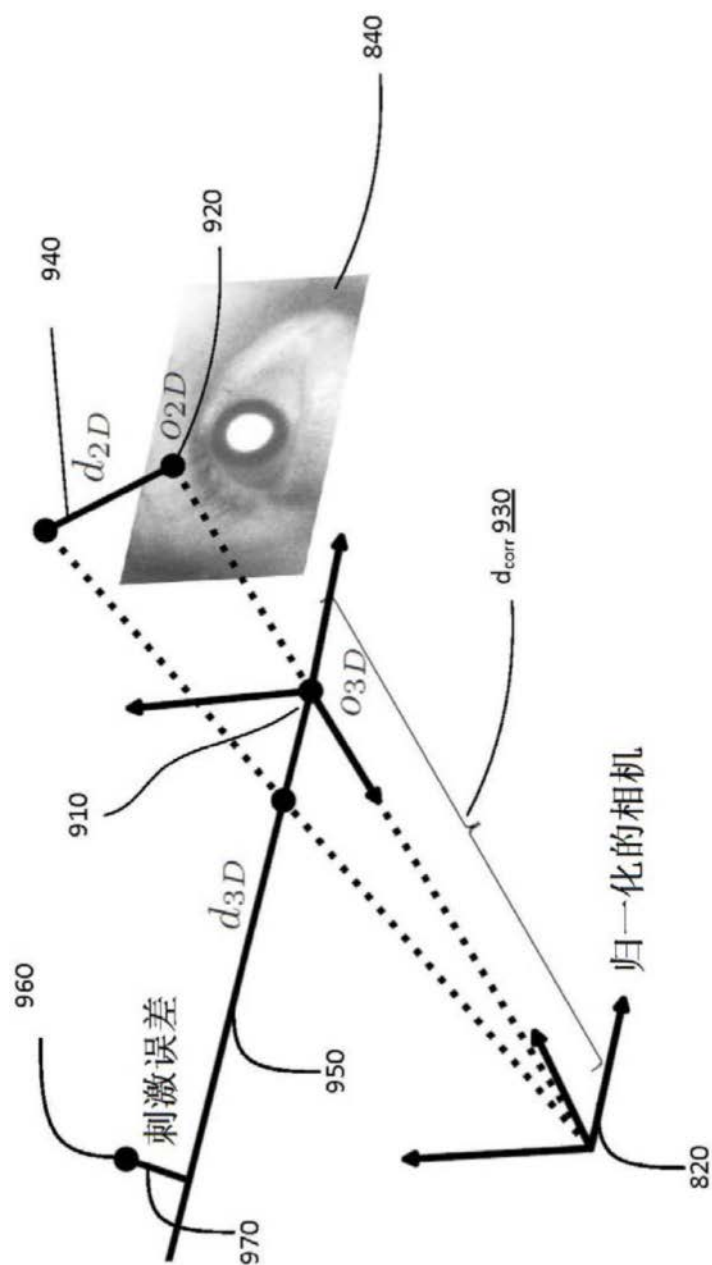


图9

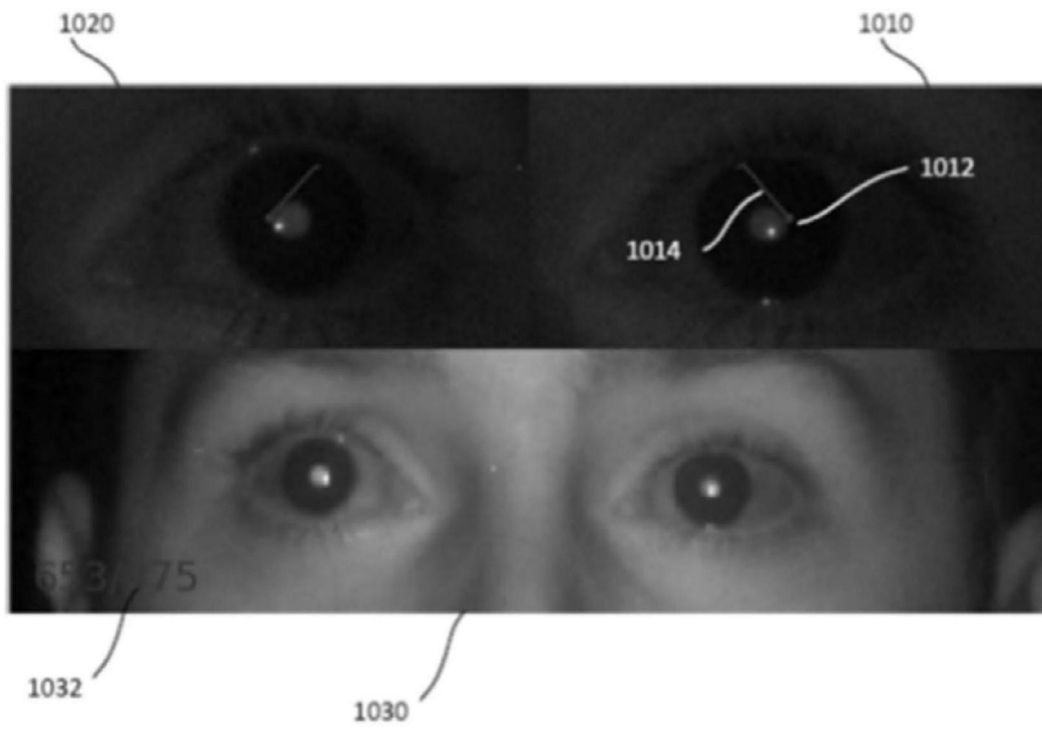


图10

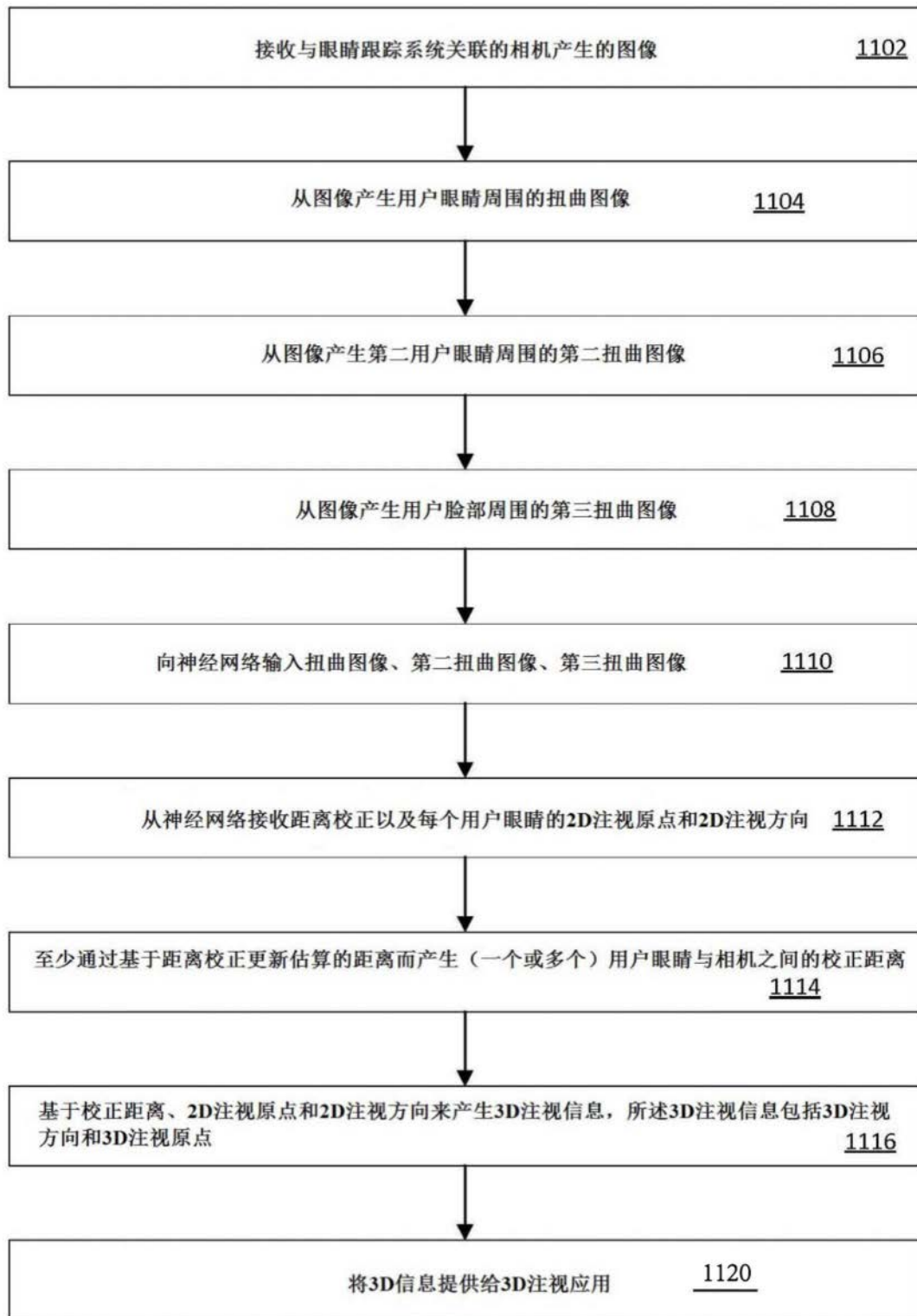


图11

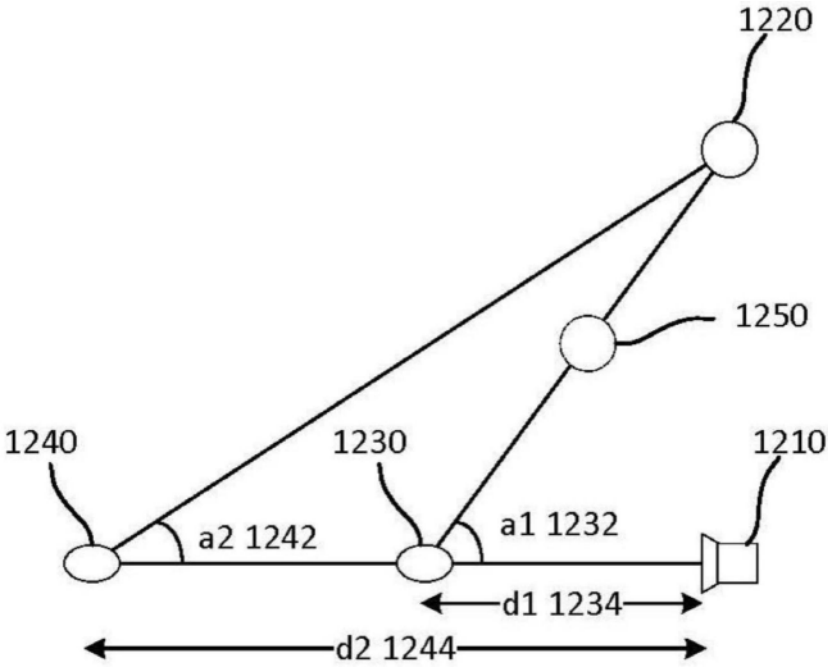


图12

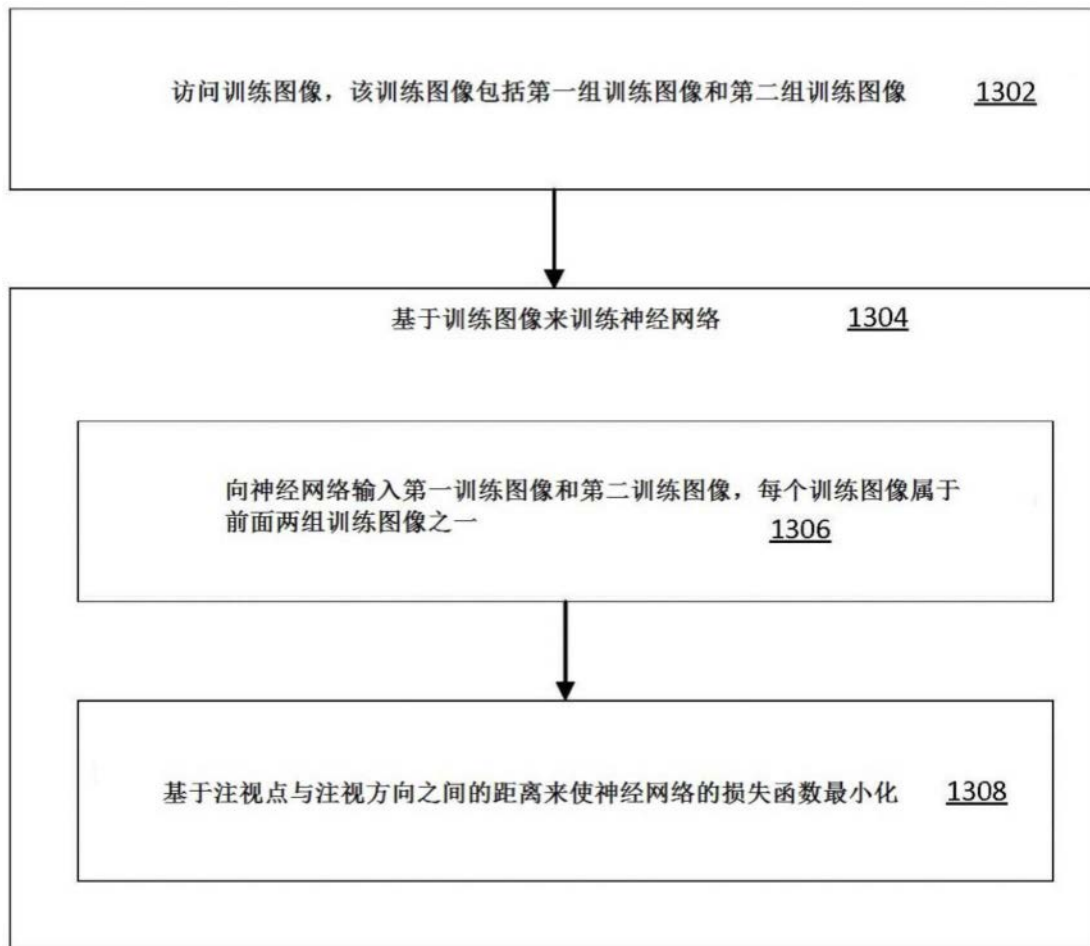


图13

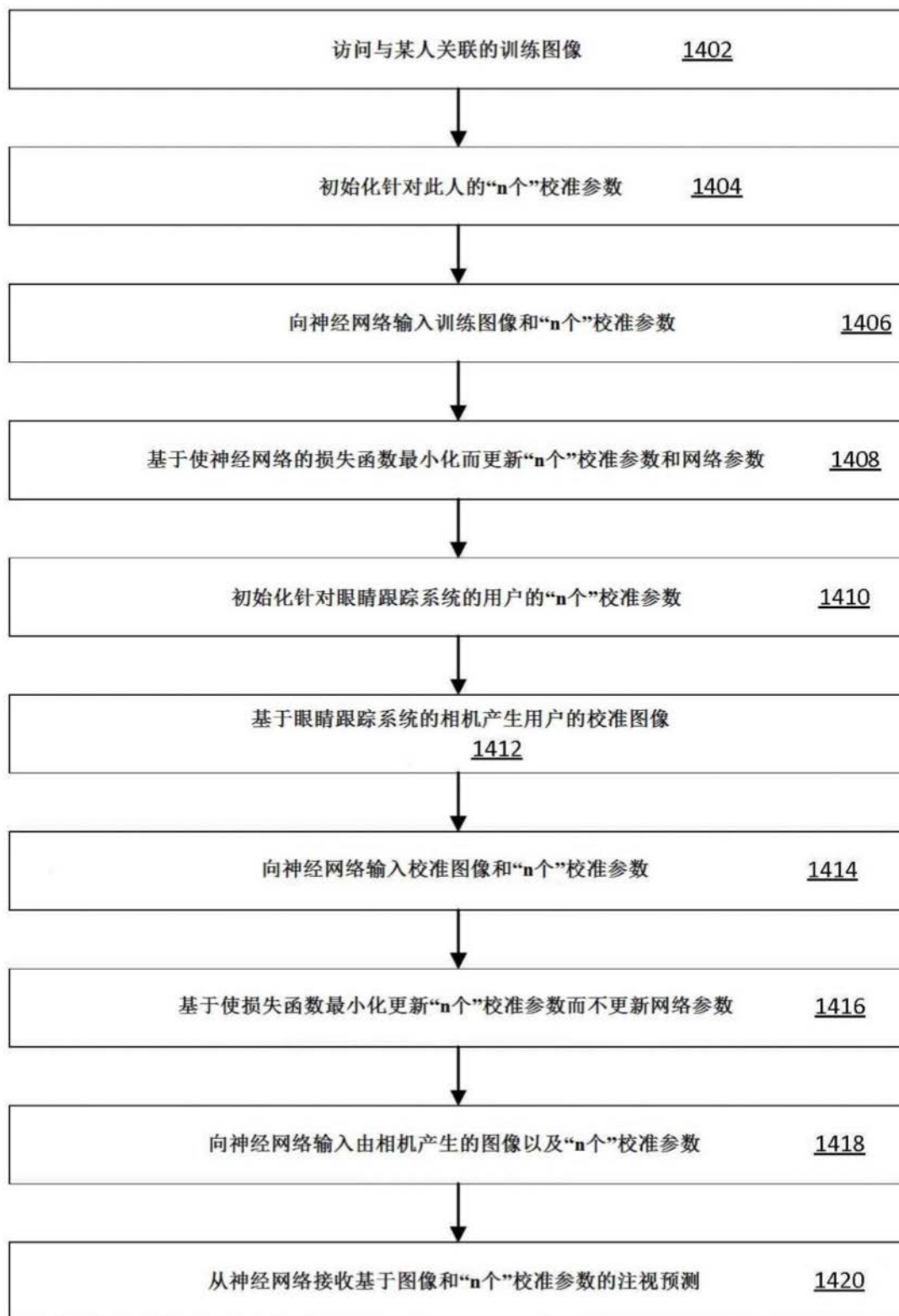


图14