



(12) 发明专利

(10) 授权公告号 CN 103472434 B

(45) 授权公告日 2015. 05. 20

(21) 申请号 201310455238. 9

(22) 申请日 2013. 09. 29

(73) 专利权人 哈尔滨工程大学

地址 150001 黑龙江省哈尔滨市南岗区南通大街 145 号哈尔滨工程大学科技处知识产权办公室

(72) 发明人 莫宏伟 孟龙龙 徐立芳 梁作玉 蒋兴洲 雍升

(51) Int. Cl.

G01S 5/18(2006. 01)

审查员 卢浩

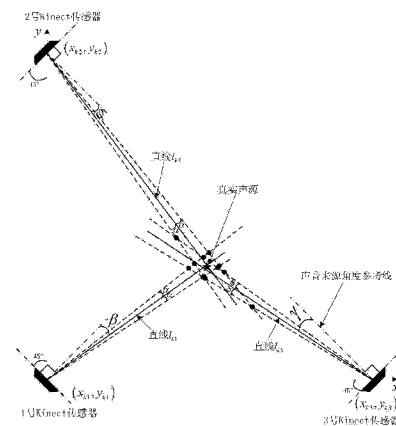
权利要求书1页 说明书6页 附图7页

(54) 发明名称

一种机器人声音定位方法

(57) 摘要

本发明公开一种机器人声音定位方法, 涉及声音定位及机器人导航, 通过至少两个 Kinect 传感器, 获取每个 Kinect 探测到的声音来源方向, 确定每两个 Kinect 传感器所确定的声源所在偏差扇面区域, 共三个区域, 由重心法求得每个区域的重心, 三个重心的均值即为声源的最优位置。通过该方法, 提高了其定位精度, 具有很强的实用性、灵活性, 可用于声音定位、机器人导航运动控制等多领域。



1. 一种机器人声音定位方法,其特征在于,包括:

通过至少两个 Kinect 作为声音传感器,获取来自机器人的声音的来源方向角;

根据所述来源方向角和所述至少两个 Kinect 的位置,确定每两个 Kinect 所确定的声源所在偏差扇面区域, Kinect 传感器探测到的声音来源方向偏差范围为  $[-\alpha, \alpha]$ ,称为偏差扇面,表示两个 Kinect 传感器确定的声源区域,为偏差扇面区域;

确定每两个偏差扇面区域的交叉区域的几何重心;

根据所述确定的几何重心,通过几何重心法计算声源最优位置,所述声源最优位置为定位得到的所述机器人的位置。

2. 根据权利要求 1 所述的机器人声音定位方法,其特征在于,在通过几何重心法计算声源最优位置之后,所述方法还包括:

根据目标位置和所述声源最优位置,确定所述机器人行动的路线;

控制所述机器人沿所述路线运动到所述目标位置。

## 一种机器人声音定位方法

### 技术领域

[0001] 本发明属于机器人领域,涉及一种机器人声音定位方法,可用于机器人运动控制、机器人室内定位与导航等领域。

### 背景技术

[0002] Kinect (啃乃特)是一种三维(3D)体感摄影机,同时它导入了即时动态捕捉、影像辨识、麦克风输入、语音辨识、社群互动等功能。在 Kinect 第一次作为 Xbox360 的外设发布时,骨骼追踪和语音识别是 Kinect SDK 最受开发者欢迎的特性,但是相比骨骼追踪,语音识别中麦克风阵列的强大功能被忽视了。一部分原因归于 Kinect 中令人兴奋的骨骼追踪系统,另一部分原因在于 Xbox 游戏操控面板以及 Kinect 体感游戏没有充分发挥 Kinect 音频处理的优势。

[0003] Kinect 的麦克风阵列在 Kinect 设备的下方。这一阵列由 4 个独立的水平分布在 Kinect 下方的麦克风组成。虽然每一个麦克风都捕获相同的音频信号,但是组成阵列可以探测到声音的来源方向。使得能够用来识别从某一个特定方向传来的声音。麦克风阵列捕获的音频数据流经过复杂的音频增强效果算法处理来移除不相关的背景噪音。所有这些复杂操作在 Kinect 硬件和 Kinect SDK 之间进行处理,这使得在一个较大空间范围内,即使人距 Kinect 有一定的距离也能够进行语音命令的识别以及判断声音的来源方向。

[0004] 机器人室内定位技术是机器人研究领域的一个热点,也是一个难点,研究者们提出了各种各样的方法。比较典型的是 RFID 技术,首先,在室内首先构建一个智能空间或称为传感器网络空间,即事先在室内地面上每隔一定的距离铺设一个 RFID 标签,每个 RFID 标签内存入了其所在位置的绝对坐标,其次,在移动机器人上装有 RFID 标签信息读取装置,当机器人移动到 RFID 标签上时,读取 RFID 标签内的坐标数据即可知道机器人当前的位置。但是该定位方法对环境有一定的要求,而且 RFID 标签的铺设间隔不同,机器人的定位精度也不同。其他定位技术的精度也受多方面因素的影响,如推测航行法对传感器的精度和机器人本身的运动系统都有很大的依赖性;WIFI、蓝牙等技术对环境也有一定的要求;声音定位技术在机器人上也有应用,但限于其复杂的处理以及易受环境噪声的影响,所以定位精度并不高,难以推广;室内地图构建定位精度虽高,但地图构建过程复杂,计算量很大,实时性难以满足要求。

### 发明内容:

[0005] 本发明提供了一种机器人声音定位方法,用于解决现有技术中存在的定位精度不高的问题。

[0006] 一方面,提供了一种机器人声音定位方法,包括:通过至少两个 Kinect 作为声音传感器,获取来自机器人的声音的来源方向角;根据所述来源方向角和所述至少两个 Kinect 的位置,确定每两个 Kinect 所确定的声源所在偏差扇面区域;确定每两个偏差扇面区域的交叉区域的几何重心;根据所述确定的几何重心,通过几何重心法计算声源最优位

置,所述声源最优位置为定位得到的所述机器人的位置。

[0007] 优选的,在通过几何重心法计算声源最优位置之后,根据目标位置和所述声源最优位置,确定所述机器人行动的路线;控制所述机器人沿所述路线运动到所述目标位置。

[0008] 通过上述方案,能够准确而方便地定位机器人的位置。

## 附图说明

[0009] 图 1 是 Kinect 声音定位总体原理图;

[0010] 图 2 是 1 号和 3 号 Kinect 传感器共同确定的声源所在区域原理图;

[0011] 图 3 是不规则四边形重心求法原理图;

[0012] 图 4 是 1 号和 2 号 Kinect 传感器共同确定的声源所在区域原理图;

[0013] 图 5 是 2 号和 3 号 Kinect 传感器共同确定的声源所在区域原理图;

[0014] 图 6 是通过 Kinect 的声音定位实现机器人导航原理图;

[0015] 图 7 是机器人声音定位导航控制原理框图。

## 具体实施方式

[0016] 下面结合附图,对本发明的具体实现过程进行详细说明。

[0017] 本发明实施例提供了一种定位机器人的方法,包括:

[0018] (1) 将 3 个 Kinect 传感器以直角坐标系形式排放,声源在该坐标系内某一位置发出持续一定时间的声音,保存三个 Kinect 传感器获得的声音来源方向角。

[0019] (2) 由于每个 Kinect 传感器接收声音信号有一定偏差,用  $\alpha$  表示偏差角度, Kinect 传感器探测到的声音来源方向偏差范围为  $[-\alpha, \alpha]$ ,称为偏差扇面,表示两个 Kinect 传感器确定的声源区域,本方法称为偏差扇面法。然后利用几何方法求得每两个 Kinect 传感器的偏差扇面区域所形成的交叉区域的重心,称为几何重心法。

[0020] (3) 利用偏差扇面法和几何重心法求得每两个 Kinect 传感器确定的声源所在区域的重心,共三个,计算这三个重心坐标的平均值,即为声源的最优位置。

[0021] (4) 将最优声源位置发送给移动机器人,机器人获得自己的位置信息,调整其运动方向,通过声音定位向目标地点移动,从而实现移动机器人声音定位导航。

[0022] 本发明实施例借助 Kinect 传感器的音频处理能力,探测声音来源方向实现机器人室内定位。

[0023] 与现有的技术相比,本发明实施例具有以下优点:

[0024] 声音定位方法简单。声音定位系统的实时性要求很高,如果采用比较复杂的算法,将不能满足声音定位系统的实时性要求。本发明采用重心法这种比较通用的方法即可快速确定出声源的位置。

[0025] 声音定位精度高。充分利用了 Kinect 传感器的麦克风阵列以及 Kinect 软件驱动程序的背景抑制及回声消除处理,消除虚拟声源及背景噪声的影响,探测真实声源方向,获得声源的最佳位置。

[0026] 定位方法通用。由于是利用 Kinect 探测声音的来源方向,所以即使有障碍物在 Kinect 和声源之间, Kinect 仍能够得到声音的来源方向;另一方面,该定位方法不仅能够用于室内声音定位,同样可应用于室外。

[0027] 可应用于机器人室内的精确定位及导航运动控制。

[0028] 本发明实施例还提供了一种机器人定位和导航方法,包括:

[0029] 1、Kinect 声音区域重心定位方法

[0030] 参照图 1,为了使声源的探测均在 Kinect 传感器的最佳探测角度范围内,三个 Kinect 传感器的布置如图所示。Kinect 传感器获得声音来源方向角是以其中线为参考的,即图中标有声音来源角度参考线的虚线,面对 Kinect 传感器,虚线左侧的角度为负值,虚线右侧的角度为正值。声源在某一位置以 16KHz 的频率发出持续 50ms 的声音。偏差扇面法和区域重心方法结合获取三个 Kinect 传感器确定的声源交叉区域重心过程如下:

[0031] 用  $\beta$  表示 1 号 Kinect 传感器探测到的声音来源方向角,  $l_{k1}$  表示 1 号 Kinect 传感器探测到的声源所在直线,故真实声源在直线  $l_{k1}$  左右偏转  $\alpha$  度的扇形区域内,2 号和 3 号 Kinect 传感器类似。 $\varphi$  表示 2 号 Kinect 传感器探测到的声音来源方向角,  $l_{k2}$  表示 2 号 Kinect 传感器探测到的声源所在直线。 $\gamma$  表示 3 号 Kinect 传感器探测到的声音来源方向角,  $l_{k3}$  表示 3 号 Kinect 传感器探测到的声源所在直线。由上述可知  $\beta$  和  $\varphi$  为负值,  $\gamma$  正值。

[0032]  $(x_{k1}, y_{k1})$  表示 1 号 Kinect 传感器所在位置坐标,  $(x_{k2}, y_{k2})$  表示 2 号 Kinect 传感器所在位置坐标,  $(x_{k3}, y_{k3})$  表示 3 号 Kinect 传感器所在位置坐标,实际中很容易测量,所以均为已知参数。

[0033] 图 1 表示了三个 Kinect 传感器探测到的声音来源方向及两两偏差扇面相互交叉所确定的声源区域,以 1 号和 3 号 Kinect 传感器的偏差扇面相互交叉形成的区域为例,其交叉区域重心求法如下:

[0034] 如图 2,  $l_{k1}^l$  表示 1 号 Kinect 传感器探测到声源所在直线逆时针偏转  $\alpha$  度,  $l_{k1}^r$  表示 1 号 Kinect 传感器探测到声源所在直线顺时针偏转  $\alpha$  度;  $l_{k3}^l$  表示 3 号 Kinect 传感器探测到声源所在直线逆时针偏转  $\alpha$  度,  $l_{k3}^r$  表示 3 号 Kinect 传感器探测到声源所在直线顺时针偏转  $\alpha$  度。 $A(x_A, y_A)$  表示  $l_{k1}^l$  和  $l_{k3}^r$  的交点,  $B(x_B, y_B)$  表示  $l_{k1}^r$  和  $l_{k3}^l$  的交点,  $C(x_C, y_C)$  表示  $l_{k1}^r$  和  $l_{k3}^l$  的交点,  $D(x_D, y_D)$  表示  $l_{k1}^l$  和  $l_{k3}^r$  的交点。四边形 ABCD 即是由 1 号和 3 号 Kinect 传感器共同确定的声源所在区域。为了方便说明求四边形 ABCD 的重心,将其放大。

[0035] 参照图 3,  $N(x_N, y_N)$  表示三角形 DAB 的重心,  $O(x_O, y_O)$  表示三角形 ABC 的重心,  $P(x_P, y_P)$  表示三角形 BCD 的重心,  $Q(x_Q, y_Q)$  表示三角形 CDA 的重心,  $R(x_R, y_R)$  表示四边形 ABCD 的重心。

[0036] 对于四边形 ABCD,连接其一条对角线 AC,这样就将四边形 ABCD 分割成三角形 ABC 和三角形 CDA 的组合体,则四边形 ABCD 的重心在三角形 ABC 重心 O 和三角形 CDA 重心 Q 的连线 OQ 上;同理,连接四边形 ABCD 另一条对角线 BD,这样就将四边形 ABCD 分割成三角形 DAB 和三角形 BCD 的组合体,四边形 ABCD 的重心同样在线段 NP 上,故有四边形 ABCD 的重心在线段 OQ 和 NP 的交点上,即  $R_1(x_{R_1}, y_{R_1})$  点。

[0037] 参照图 4 和图 5,四边形 EFGH 即是由 1 号和 2 号 Kinect 传感器共同确定的声源所在区域,四边形 IJKL 即是由 2 号和 3 号 Kinect 传感器共同确定的声源所在区域,按照上述

同样的方法求得四边形 EFGH 重心为  $R_2(x_{R_2}, y_{R_2})$ , 四边形 IJKL 重心为  $R_3(x_{R_3}, y_{R_3})$ 。

[0038] 最后, 求四边形 ABCD、EFGH、IJKL 的重心  $R_1(x_{R_1}, y_{R_1})$ 、 $R_2(x_{R_2}, y_{R_2})$ 、 $R_3(x_{R_3}, y_{R_3})$  三个坐标的均值即为声源的最优位置  $S(x_s, y_s)$ 。

[0039] 按照上述方法, 同理可求得 1 号和 2 号 Kinect 传感器相互交线所构成的区域的重心以及 2 号和 3 号 Kinect 传感器相互交线所构成的区域的重心。

[0040] 2、最优声源位置算法

[0041] 确定最优声源位置的方法包括:

[0042] 步骤 1: 初始化多个参数

[0043] 三个 Kinect 传感器的位置坐标均可实际测得, 即  $(x_{k1}, y_{k1})$ 、 $(x_{k2}, y_{k2})$  和  $(x_{k3}, y_{k3})$  均为已知参数, 误差角度  $\alpha$  根据 Kinect 的技术指标及实际实验设定为  $5^\circ$ , 声源及三个 Kinect 传感器开始工作。

[0044] 步骤 2: 求直线交点坐标

[0045] 声源发出持续 50ms 的声音后停止发声, 保存 1 号 Kinect 传感器获得的声音来源方向角  $\beta$ , 2 号 Kinect 传感器获得的声音来源方向角  $\varphi$ , 3 号 Kinect 传感器获得的声音来源方向角  $\gamma$ 。

[0046] 根据点斜式可列出如下直线方程:

[0047] 直线  $l_{k1}^l$ :  $y = \tan(45^\circ + \beta + \alpha)x$  (1)

[0048] 直线  $l_{k1}^r$ :  $y = \tan(45^\circ + \beta - \alpha)x$  (2)

[0049] 直线  $l_{k3}^l$ :  $y = y_{k3} + \tan(135^\circ + \gamma + \alpha)(x - x_{k3})$  (3)

[0050] 直线  $l_{k3}^r$ :  $y = y_{k3} + \tan(135^\circ + \gamma - \alpha)(x - x_{k3})$  (4)

[0051] 直线  $l_{k1}^l$  和直线  $l_{k3}^r$  的交点为  $A(x_A, y_A)$ , 解直线方程(1)和(4)组成的方程组得交点  $A(x_A, y_A)$  的坐标。直线  $l_{k1}^r$  和直线  $l_{k3}^l$  的交点为  $B(x_B, y_B)$ , 解直线方程(2)和(4)组成的方程组得交点  $B(x_B, y_B)$  的坐标。直线  $l_{k1}^l$  和直线  $l_{k3}^l$  的交点为  $C(x_C, y_C)$ , 解直线方程(1)和(3)组成的方程组得交点  $C(x_C, y_C)$  的坐标。直线  $l_{k1}^r$  和直线  $l_{k3}^r$  的交点为  $D(x_D, y_D)$ , 解直线方程(2)和(3)组成的方程组得交点  $D(x_D, y_D)$  的坐标。

[0052] 步骤 3: 求不规则四边形重心

[0053] 由三角形重心坐标公式,

[0054] 三角形 DAB 的重心  $N(x_N, y_N)$ ,  $x_N = \frac{x_D + x_A + x_B}{3}$ ,  $y_N = \frac{y_D + y_A + y_B}{3}$ ,

[0055] 三角形 ABC 的重心  $O(x_O, y_O)$ ,  $x_O = \frac{x_A + x_B + x_C}{3}$ ,  $y_O = \frac{y_A + y_B + y_C}{3}$ ,

[0056] 三角形 BCD 的重心  $P(x_P, y_P)$ ,  $x_P = \frac{x_B + x_C + x_D}{3}$ ,  $y_P = \frac{y_B + y_C + y_D}{3}$ ,

[0057] 三角形 CDA 的重心  $Q(x_Q, y_Q)$ ,  $x_Q = \frac{x_C + x_D + x_A}{3}$ ,  $y_Q = \frac{y_C + y_D + y_A}{3}$ 。

[0058] 根据两点式可列出如下直线方程：

[0059] 线段 OQ 所在直线方程： $y = y_O + \frac{y_Q - y_O}{x_Q - x_O}(x - x_O)$  (5)

[0060] 线段 NP 所在直线方程： $y = y_N + \frac{y_P - y_N}{x_P - x_N}(x - x_N)$  (6)

[0061] 线段 OQ 和 NP 的交点为  $R_1(x_{R_1}, y_{R_1})$ ，解直线方程 (5) 和 (6) 组成的方程组得四边形 ABCD 重心为  $R_1(x_{R_1}, y_{R_1})$ 。同理，按照步骤 2 和步骤 3 的方法求得求四边形 EFGH、IJKL 的重心  $R_2(x_{R_2}, y_{R_2})$ 、 $R_3(x_{R_3}, y_{R_3})$ 。

[0062] 步骤 4：求声源最优位置

[0063] 声源最优位置  $S(x_S, y_S)$  坐标  $x_S = \frac{x_{R_1} + x_{R_2} + x_{R_3}}{3}$ ,  $y_S = \frac{y_{R_1} + y_{R_2} + y_{R_3}}{3}$ ，以此代表真实的声源的位置。

[0064] 3、机器人 Kinect 声音定位导航算法

[0065] 参照图 6，机器人上装有声源，发送声音到空间中，某一时刻机器人位于  $W(x_W, y_W)$ ，机器人目标位置为  $V(x_V, y_V)$ 。机器人上装有磁罗盘，可以测出机器人自身坐标轴  $y_1$  和北向的夹角。不失一般性，设  $\delta$  为坐标轴  $y$  和北向的夹角。

[0066]  $\phi$  表示机器人偏离理论行走轨线的夹角，如果机器人整个控制系统及轮子结构均无误差，则给机器人一直行信号，使其行走一定距离即到达目标  $V(x_V, y_V)$ ，即机器人理论上的行走路线如图 WV 实线所示。而实际给机器人一直行信号，由于轮子结构误差，机器人总会偏离原始方向，即实际机器人沿着 WV 虚线行走。

[0067]  $\varepsilon$  表示机器人最终到达以目标点  $V(x_V, y_V)$  为圆心的误差圆半径，因为实际中存在其他各种因素的影响，例如机器人本身的尺寸，机器人最终到达的位置不一定和目标点  $V(x_V, y_V)$  完全重合，所以根据实际精度要求定义一误差圆。只要机器人到达以目标点  $V(x_V, y_V)$  为圆心， $\varepsilon$  为半径的误差圆内，则认为机器人已到达目标点。Kinect 声音定位机器人导航算法步骤如下：

[0068] 步骤 1：

[0069] 根据机器人当前位置  $W(x_W, y_W)$  和目标位置  $V(x_V, y_V)$  可计算出直线 WV 与水平  $x$  轴

正向的夹角  $\theta$  及线段 WV 的距离  $d$ ,  $\theta = \text{atan} \frac{y_V - y_W}{x_V - x_W}$ ,  $d = \sqrt{(x_W - x_V)^2 + (y_W - y_V)^2}$ 。则为了

朝目标  $V(x_V, y_V)$  运动，机器人调整其  $y_1$  轴方向和北向夹角为  $90^\circ - \theta - \delta$ 。机器人从起点  $W(x_W, y_W)$  开始行走。

[0070] 步骤 2：

[0071] 机器人每隔 1s 发出持续 50ms 的声音。设某一时刻机器人位于  $U(x_U, y_U)$ ，计算机处理三个 Kinect 传感器获得的声源方向信息得到机器人当前位置  $U(x_U, y_U)$ ，将该位置通过无线发送给机器人。

[0072] 步骤 3 :

[0073] 机器人计算当前位置  $U(x_u, y_u)$  和起点  $W(x_w, y_w)$  的连线  $WU$  与  $WV$  的偏离角度  $\phi$ , 机器人自动调整其  $y_1$  轴朝着偏离角度  $\phi$  减小的方向运动。

[0074] 步骤 4 :

[0075] 重复步骤 2 到步骤 3, 最终, 当计算机处理三个 Kinect 传感器获得的声源方向信息得到机器人当前位置在以目标点  $V(x_v, y_v)$  为圆心,  $\varepsilon$  为半径的误差圆内, 则认为机器人已到达目标位置, 完成了机器人从起点  $W(x_w, y_w)$  到目标点  $V(x_v, y_v)$  的导航任务。

[0076] 步骤 5 :

[0077] 从点  $V(x_v, y_v)$  到目标点  $Z(x_z, y_z)$  的导航任务重复步骤 1 到 4。

[0078] 按照上述步骤不断地重复, 基于 Kinect 的声音定位就可实现机器人的连续导航任务, 证明了上述 Kinect 的声音定位算法正确且有效。

[0079] 图 7 是机器人整个声音定位导航控制原理框图, PC (个人电脑) 机上连接 3 个 Kinect 传感器和一个无线数传模块, 无线数传模块用于 PC 机和机器人之间的通信。机器人上包括机器人核心处理器、电机驱动、磁罗盘、无线数传模块和声源模块。机器人核心处理器用于机器人的运动控制, 机器人和 PC 机的通信, 采集传感器信息等。电机驱动用于机器人电机功率的放大。磁罗盘用于测量机器人自身相对地理真北的夹角。声源模块用于机器人发出声音信号, 以使用 3 个 Kinect 传感器实现机器人声音定位。

[0080] 上述内容仅是本发明的优选实施例, 在此基础上, 本领域技术人员能够做出一些变形, 在不偏离本发明的思想的情况下, 这些变形也应在本发明的保护范围之内。



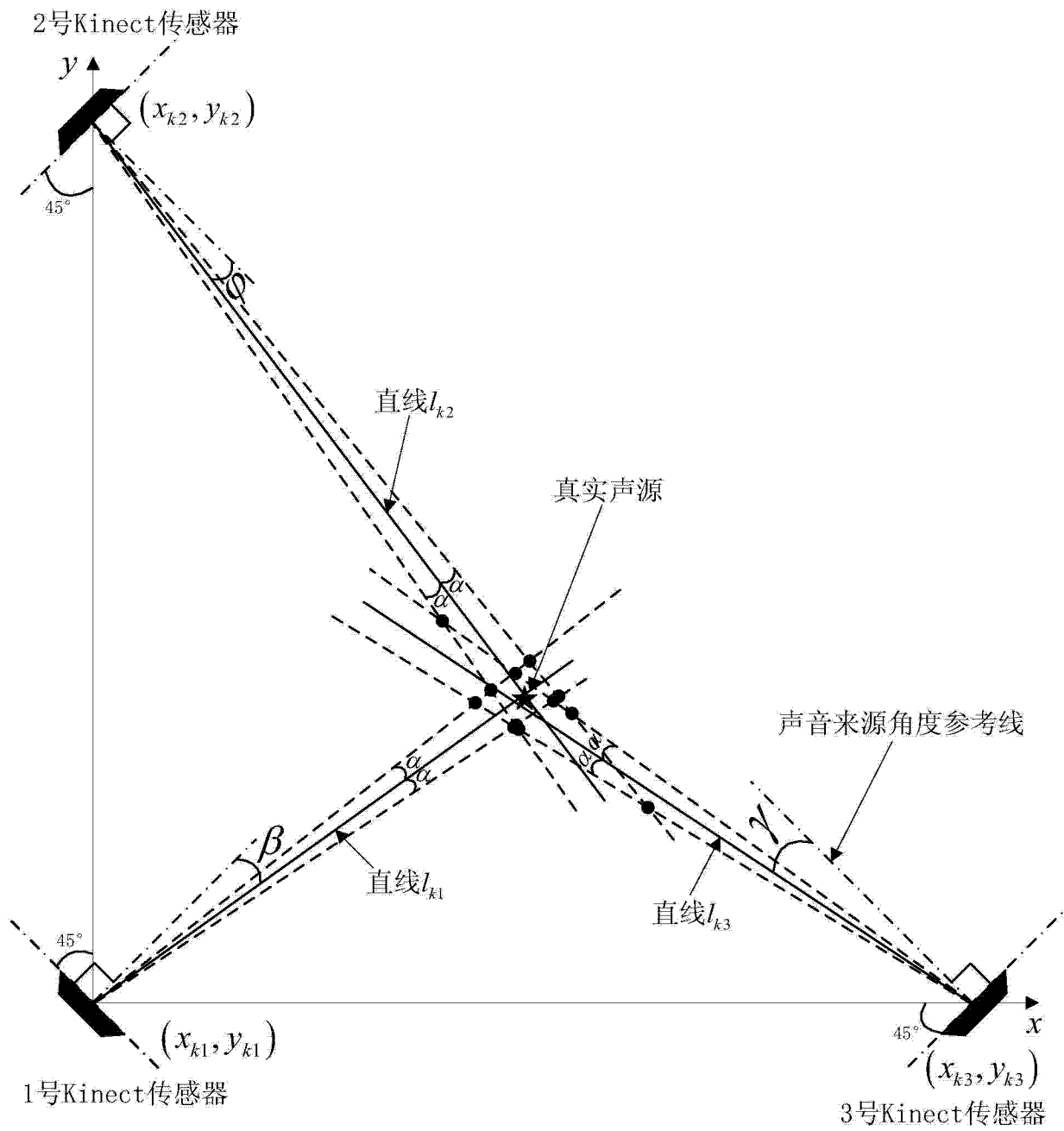


图 1

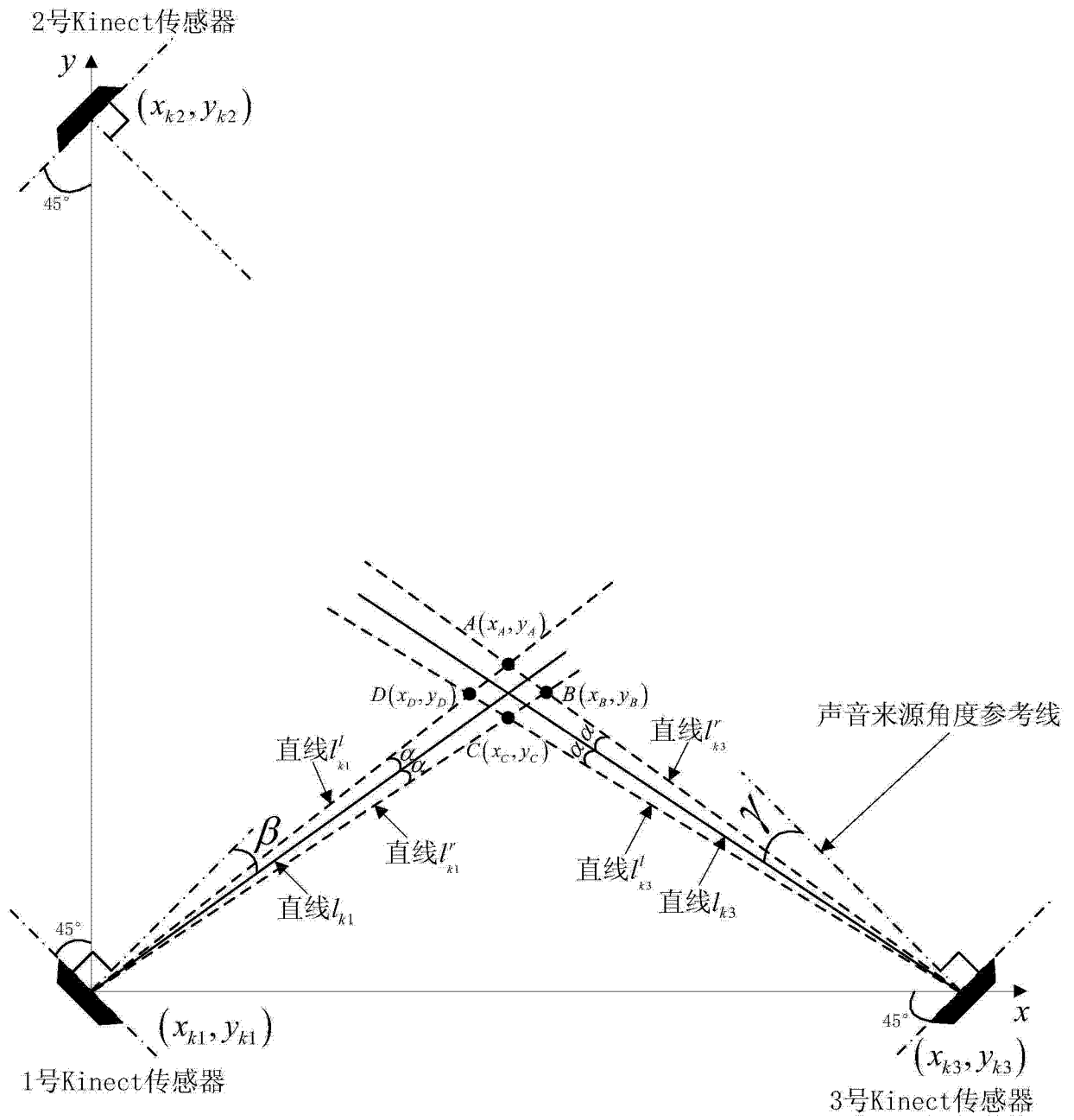


图 2

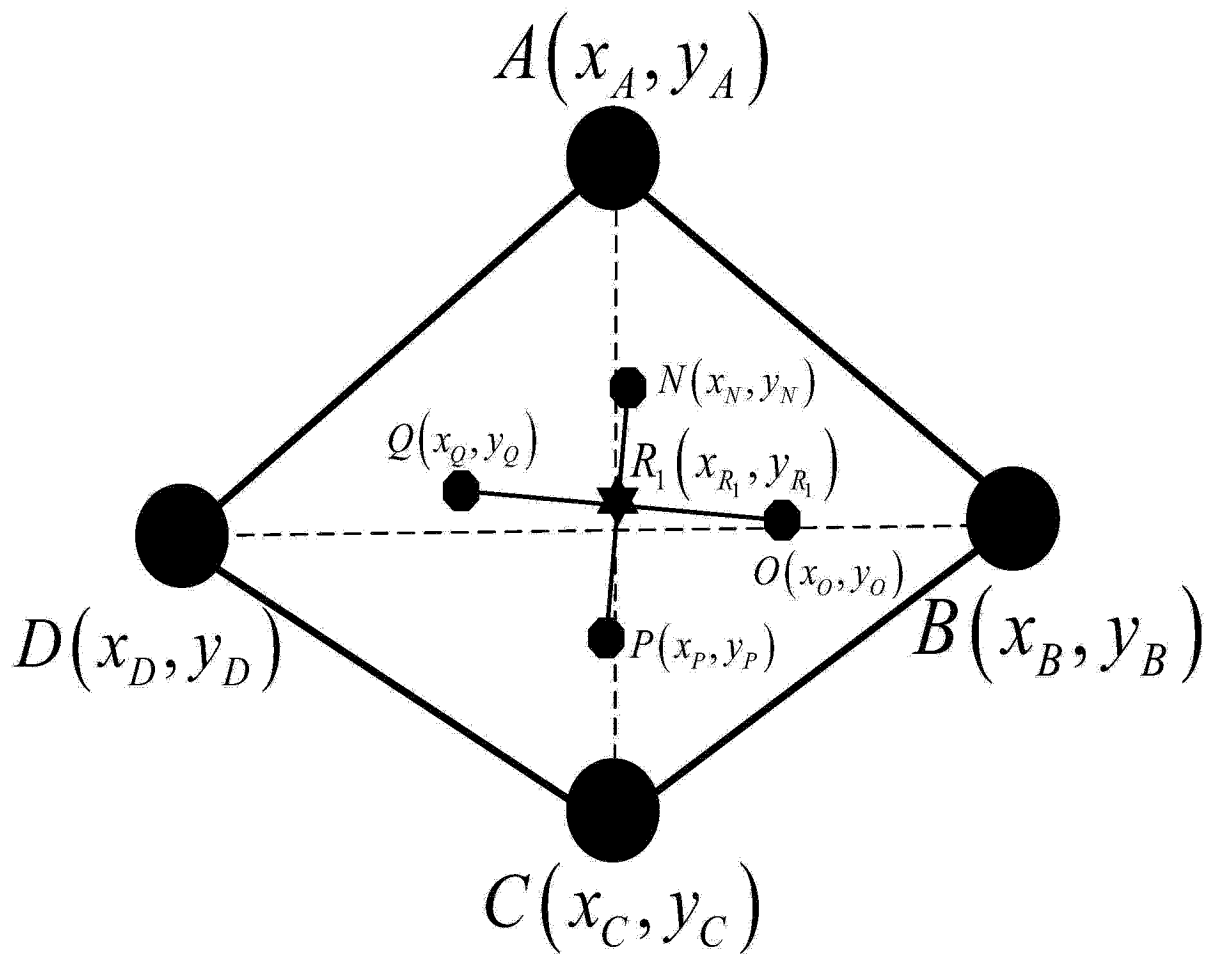


图 3

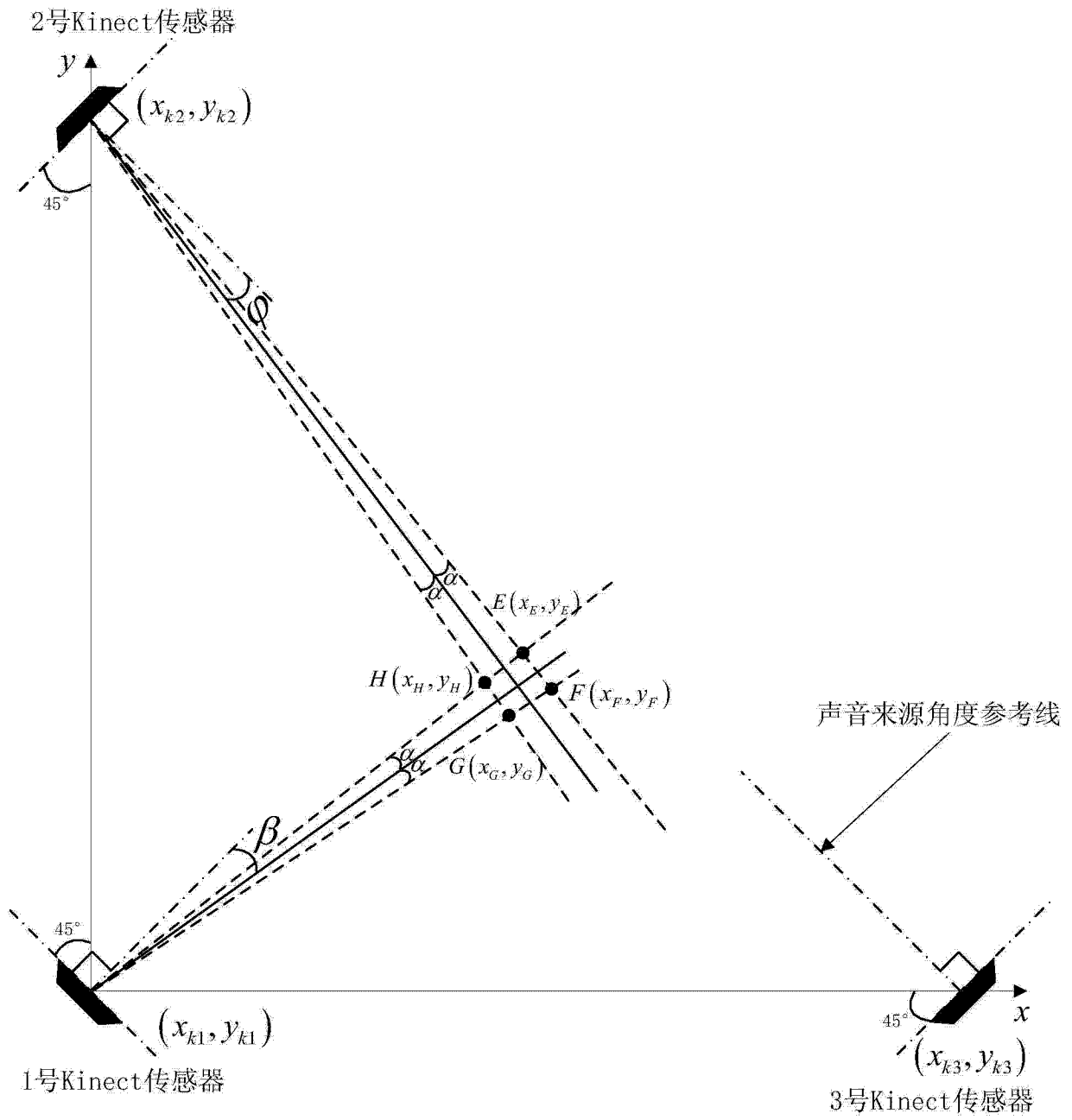


图 4

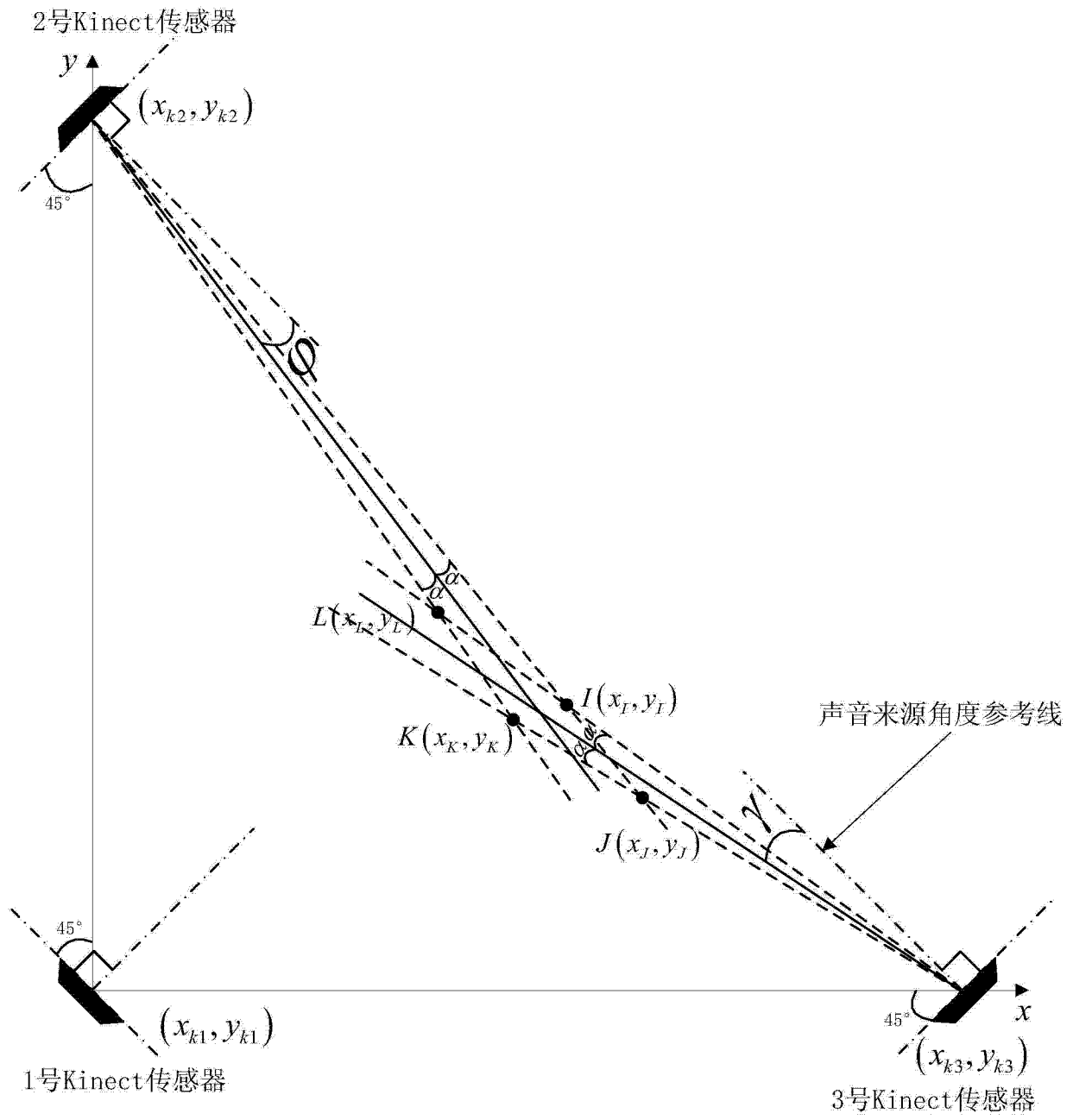


图 5

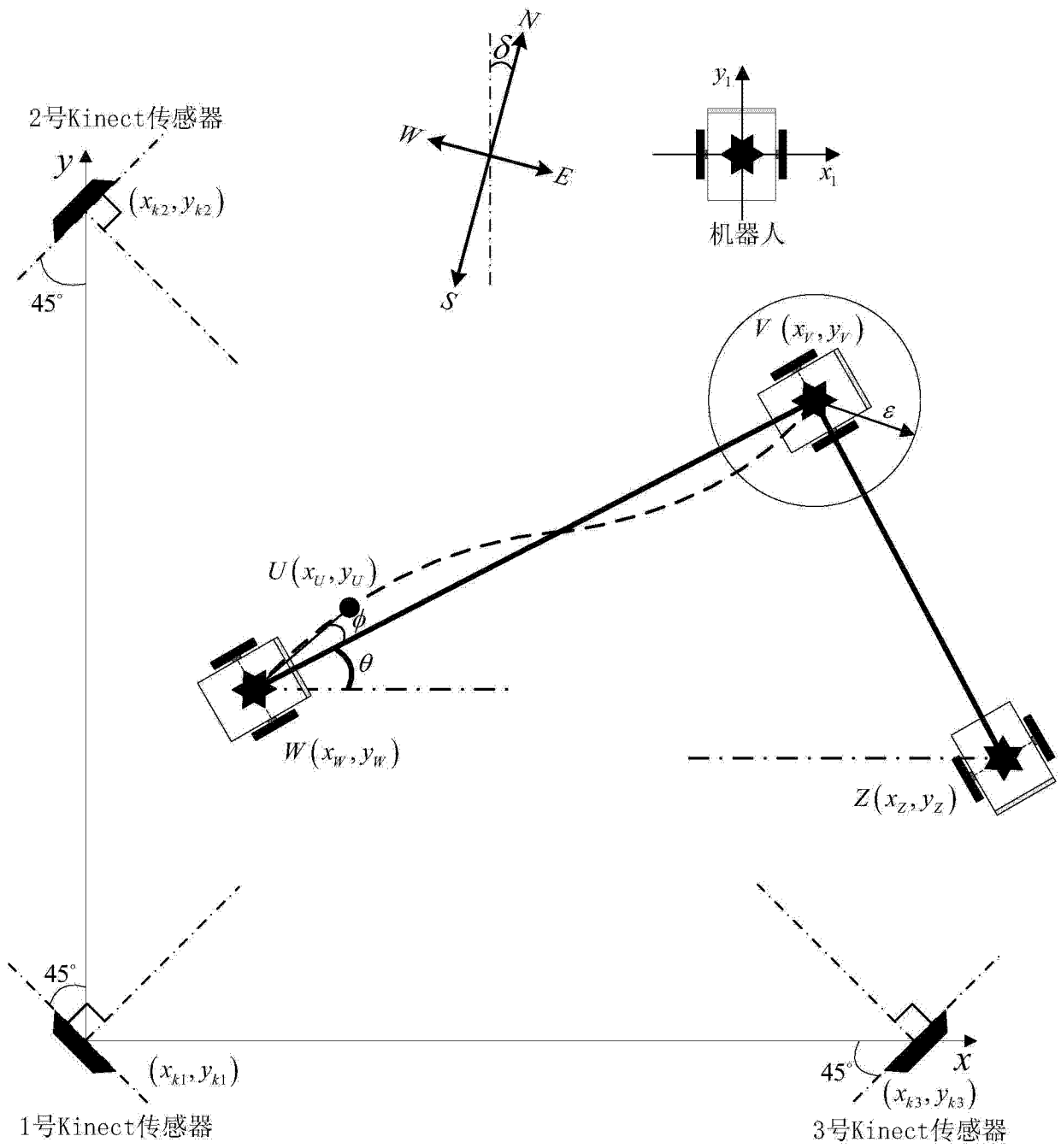


图 6

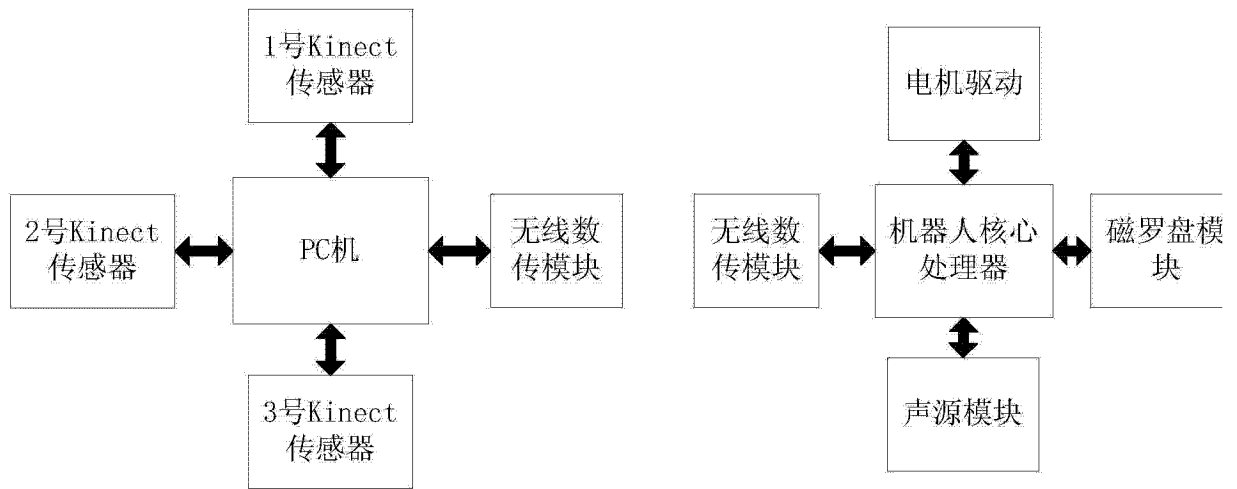


图 7