

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2019-83857

(P2019-83857A)

(43) 公開日 令和1年6月6日(2019.6.6)

(51) Int. Cl.	F 1	テーマコード (参考)
<b>A 6 1 B</b> 5/11 (2006.01)	A 6 1 B 5/10 3 1 0 G	3 C 7 0 7
<b>B 6 0 W</b> 40/04 (2006.01)	A 6 1 B 5/10 3 1 0 B	3 D 2 4 1
<b>B 2 5 J</b> 19/02 (2006.01)	B 6 0 W 40/04	4 C 0 3 8
	B 2 5 J 19/02	

審査請求 未請求 請求項の数 6 O L (全 16 頁)

(21) 出願番号	特願2017-212095 (P2017-212095)	(71) 出願人	504133110 国立大学法人電気通信大学 東京都調布市調布ケ丘一丁目5番地1
(22) 出願日	平成29年11月1日(2017.11.1)	(74) 代理人	110000925 特許業務法人信友国際特許事務所
		(72) 発明者	金森 哉吏 東京都調布市調布ケ丘一丁目5番地1 国立大学法人電気通信大学内
		(72) 発明者	岡 和則 東京都調布市調布ケ丘一丁目5番地1 国立大学法人電気通信大学内
		F ターム(参考)	3C707 KS11 KS36 WA03 WA13 WL05 3D241 BA31 BA49 BA50 DC31Z DC32Z 4C038 VA04 VA12 VB14 VB15 VC20

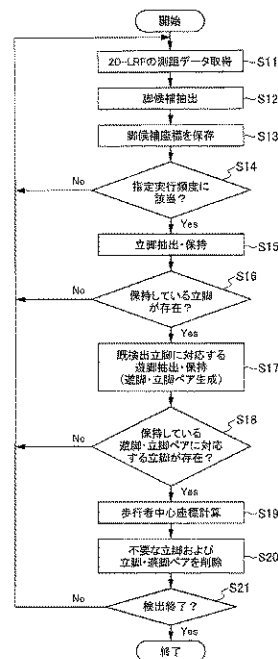
(54) 【発明の名称】 歩行者認識装置、歩行者認識方法及びプログラム

(57) 【要約】

【課題】 歩行速度などの変化に影響されずに的確に歩行動作の評価ができるようにする。

【解決手段】 測距センサから周囲の物体までの距離を示す測距データを取得し、測距データから、想定される脚のサイズを満たす物体を脚候補として抽出する。そして、脚候補の座標の変化に基づいて、立脚期と遊脚期を区別し、立脚期とその立脚期に続いた遊脚期とを示す立脚・遊脚ペアを取得する。その後、立脚・遊脚ペアとその近傍の立脚とから歩行者中心を算出して、算出した歩行者中心の変化から歩行軌跡を得る。

【選択図】 図4



## 【特許請求の範囲】

## 【請求項 1】

測距センサと、

前記測距センサから周囲の物体までの距離を示す測距データを取得する測定部と、

前記測定部で取得された測距データから、想定される脚のサイズを満たす物体を脚候補として抽出する脚候補抽出部と、

前記脚候補抽出部により得られた脚候補の座標の変化に基づいて、脚候補から立脚期と遊脚期を判別する脚候補状態判定部と、

前記脚候補状態判定部で判定された立脚期と遊脚期のデータから、立脚期とその立脚期に続いた遊脚期とを示す立脚・遊脚ペアを取得する歩行者抽出部と、

前記歩行者抽出部により得られた立脚・遊脚ペアと、その立脚・遊脚ペアの近傍の立脚とから歩行者中心を算出して、算出した歩行者中心の変化から歩行軌跡を得る歩行者中心算出部と、を備える

歩行者認識装置。

## 【請求項 2】

前記脚候補抽出部は、前記測距データで得られない隠れ脚を補間して、脚候補の中心又は重心を求める

請求項 1 に記載の歩行者認識装置。

## 【請求項 3】

前記脚候補状態判定部は、脚候補の中心又は重心の座標の移動状態から、立脚期と遊脚期を分類する

請求項 1 又は 2 に記載の歩行者認識装置。

## 【請求項 4】

前記測定部で取得される測距データは、歩行者の足首近傍の高さで距離を計測したデータである

請求項 1 ~ 3 のいずれか 1 項に記載の歩行者認識装置。

## 【請求項 5】

測距センサから周囲の物体までの距離を示す測距データを前記測距センサにより取得する測定処理と、

前記測定処理により取得された測距データから、想定される脚のサイズを満たす物体を脚候補として抽出する脚候補抽出処理と、

前記脚候補抽出処理により得られた脚候補の座標の変化に基づいて、脚候補から立脚期と遊脚期を判別する脚候補状態判定処理と、

前記脚候補状態判定処理で判定された立脚期と遊脚期のデータから、立脚期とその立脚期に続いた遊脚期とを示す立脚・遊脚ペアを取得する歩行者抽出処理と、

前記歩行者抽出処理で得られた立脚・遊脚ペアと、その立脚・遊脚ペアの近傍の立脚とから歩行者中心を算出して、算出した歩行者中心の変化から歩行軌跡を得る歩行者中心算出処理と、を含む

歩行者認識方法。

## 【請求項 6】

測距センサから周囲の物体までの距離を示す測距データを前記測距センサにより取得する測定手順と、

前記測定手順により取得された測距データから、想定される脚のサイズを満たす物体を脚候補として抽出する脚候補抽出手順と、

前記脚候補抽出手順により得られた脚候補の座標の変化に基づいて、脚候補から立脚期と遊脚期を判別する脚候補状態判定手順と、

前記脚候補状態判定手順で判定された立脚期と遊脚期のデータから、立脚期とその立脚期に続いた遊脚期とを示す立脚・遊脚ペアを取得する歩行者抽出手順と、

前記歩行者抽出手順で得られた立脚・遊脚ペアと、その立脚・遊脚ペアの近傍の立脚とから歩行者中心を算出して、算出した歩行者中心の変化から歩行軌跡を得る歩行者中心算

10

20

30

40

50

出手順と、  
をコンピュータに実行させるプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、測距センサの検出信号に基づいて歩行者を認識する歩行者認識装置及び歩行者認識方法、ならびに歩行者認識方法を実行するプログラムに関する。

【背景技術】

【0002】

自律移動ロボットなどの移動体を開発する上では、その移動体の周囲の歩行者を認識して、その歩行者を避けた自律移動を行うことが重要である。従来、自律移動ロボットには、レーザ測定器などの測距センサを搭載して、自律移動ロボットの周囲の物体や人物を検出することが行われている。測距センサが検出した対象物が静止した物体である場合には、自律移動ロボットがその物体を避けた移動を行えばよく、比較的容易に対処が可能である。一方、歩行者の場合には、歩行者の移動状態を正しく認識して、その歩行者の移動経路を考慮して、自律移動ロボットの移動経路を決める必要があり、歩行者の歩行状態を正確に認識することが重要である。

【0003】

特許文献1には、レーザ測定器を使って歩行者の脚部位置を検出して、歩行状態を計測するシステムの例が記載されている。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特開2014-176461号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

特許文献1に記載されるように、歩行状態を計測するシステムは従来から各種提案されているが、測距センサを使用するシステムでは、歩行者と測距センサの位置関係によっては、歩行者の検出が困難になる場合がある。具体的には、歩行者が測距センサの前方を横切るように歩行した場合には、測距センサが2本の脚をほぼ連続して検出することができるので、歩行者を比較的精度よく補足することが可能である。これに対して、測距センサに近づいて来るように歩行している歩行者や、測距センサから遠ざかるように歩行している歩行者の場合には、歩行状態によって一方の脚の一部又は全てが他方の脚の後方に隠れることがあり、そのような状態が発生すると、歩行者を見失ってしまう場合がある。特に、測距センサを搭載した自律移動ロボットの周囲に多数の歩行者が存在している環境では、周囲の全ての歩行者の動きを常時正確に検出することは困難であった。

【0006】

本発明は、かかる点に鑑み、一方の脚が他方の脚の背後に隠れるような状況でも歩行者を的確に補足することができる歩行者認識装置及び歩行者認識方法を提供することを目的とする。

【課題を解決するための手段】

【0007】

本発明の歩行者認識装置は、測距センサと、測距センサから周囲の物体までの距離を示す測距データを取得する測定部と、測定部で取得された測距データから、想定される脚のサイズを満たす物体を脚候補として抽出する脚候補抽出部と、脚候補抽出部が得た脚候補の座標の変化に基づいて、脚候補から立脚期と遊脚期を判別する脚候補状態判定部と、脚候補状態判定部で判定された立脚期と遊脚期のデータから、立脚期とその立脚期に続いた遊脚期とを示す立脚・遊脚ペアを取得する歩行者抽出部と、歩行者抽出部で得られた立脚

10

20

30

40

50

・遊脚ペアと、その立脚・遊脚ペアの近傍の立脚とから歩行者中心を算出して、算出した歩行者中心の変化から歩行軌跡を得る歩行者中心算出部と、を備える。

【0008】

また、本発明の歩行者認識方法は、測距センサから周囲の物体までの距離を示す測距データを測距センサで取得する測定処理と、測定処理により取得した測距データから、想定される脚のサイズを満たす物体を脚候補として抽出する脚候補抽出処理と、脚候補抽出処理により得られた脚候補の座標の変化に基づいて、脚候補から立脚期と遊脚期を判別する脚候補状態判定処理と、脚候補状態判定処理で判定された立脚期と遊脚期のデータから、立脚期とその立脚期に続いた遊脚期とを示す立脚・遊脚ペアを取得する歩行者抽出処理と、歩行者抽出処理で得られた立脚・遊脚ペアと、その立脚・遊脚ペアの近傍の立脚とから歩行者中心を算出して、算出した歩行者中心の変化から歩行軌跡を得る歩行者中心算出処理と、を含む。

10

【0009】

また、本発明のプログラムは、上記歩行者認識方法の測定処理、脚候補抽出処理、脚候補中心座標分類処理、歩行者抽出処理、及び歩行者中心算出処理のそれぞれを実行する手順を、コンピュータに実行させるものである。

【発明の効果】

【0010】

本発明によれば、測距センサの周囲の歩行者の歩行軌跡を的確に取得できるようになる。また、複数の歩行者が存在する環境であっても、それぞれの歩行者の歩行軌跡を的確に取得してトラッキングすることができる。

20

【図面の簡単な説明】

【0011】

【図1】本発明の一実施の形態例による歩行者認識装置の構成例を示すブロック図である。

【図2】人間の歩行時の立脚期と遊脚期を示す図である。

【図3】人間の歩行時の左右の脚と中心軌跡の例を示す図である。

【図4】本発明の一実施の形態例による歩行者認識処理の流れを示すフローチャートである。

【図5】本発明の一実施の形態例による脚候補抽出処理の詳細を示すフローチャートである。

30

【図6】本発明の一実施の形態例による測定環境の例を示す図である。

【図7】本発明の一実施の形態例による測距データ配列の例を示す図である。

【図8】本発明の一実施の形態例による測距データ配列の例を示す図である。

【図9】本発明の一実施の形態例による差分の配列diffの例を示す図である。

【図10】本発明の一実施の形態例による条件を満たす要素番号の例を示す図である。

【図11】本発明の一実施の形態例による正負差分エッジを示す図である。

【図12】本発明の一実施の形態例によるクラスタリング例を示す図である。

【図13】本発明の一実施の形態例による立脚期と遊脚期のペアの生成処理例を示す図である。

40

【図14】本発明の一実施の形態例の処理で実際に生成された立脚期と遊脚期のペアの例を示す図である。

【図15】本発明の一実施の形態例による歩行者中心検出の具体例（歩行状態）を示す図である。

【図16】本発明の一実施の形態例による歩行者中心検出の具体例（検出状態）を示す図である。

【図17】本発明の一実施の形態例による歩行者中心検出の具体例（中心検出状態）を示す図である。

【図18】本発明の一実施の形態例の処理で実際に歩行者中心を計算した例を示す図である。

50

**【発明を実施するための形態】****【0012】**

以下、本発明の一実施の形態例（以下、「本例」と称する）について、添付図面を参照して説明する。

**【0013】****[1. 歩行者認識装置の構成]**

図1は、本例の歩行者認識装置の構成を示す機能ブロック図である。本例の歩行者認識装置は、例えばコンピュータ装置に歩行者認識処理を実行するプログラムを実装することで構成される。

**【0014】**

図1に示す歩行者認識装置は、中央制御ユニット（以下、「CPU」と称する）11、プログラムメモリ12、外部入出力インタフェース13、脚候補抽出部14、脚候補中心座標分類部15、脚候補状態判定部16、歩行者抽出部17、歩行者中心算出部18、及び歩行者特徴量算出部19を備える。

また、歩行者認識装置は、それぞれの処理部で得たデータを記憶するメモリとして、脚候補中心メモリ21、脚候補クラスメモリ22、歩行者クラスメモリ23、歩行者中心メモリ24、及び歩行者特徴量メモリ25を備える。

**【0015】**

また、歩行者認識装置は、測定部31と2次元領域センサ32を備える。測定部31は、歩行者認識を行う上で必要な測定データ（測距データ）を取得する。すなわち、測定部31には、2次元領域センサ32から測距データが供給され、測定部31は、この測距データに基づいて測定処理を行う。なお、測距データは、2次元領域センサ32から周囲の物体までの距離を示すデータである。本例の場合には、2次元領域センサ32として、2Dレーザレンジファインダ（laser range finder：2D-LRF）が使用される。2次元領域センサ32は、地面から約20cmの高さの物体までの距離を検出する。測定部31で得られた測距データは、2次元距離データメモリ33に記憶される。ここでは、2次元距離データメモリ33には、2次元領域センサ32からの距離Lと、2次元領域センサ32から見た方位角が記憶される。

**【0016】**

CPU11は、プログラムメモリ12に記憶されたプログラムを実行することで、歩行者認識装置による歩行者認識処理動作を実行する。プログラムメモリ12は、ROMとRAMで構成される。

外部入出力インタフェース13は、歩行者認識装置で得られた歩行者認識結果を外部に出力する。また、外部入出力インタフェース13は、外部から歩行者認識装置の動作モードなどを設定するデータを受信する。

**【0017】**

脚候補抽出部14は、測定部31で取得される測距データに基づいて、人間の脚候補を抽出する脚候補抽出処理を行う。また、脚候補抽出部14は、測距データでは見えていない、いわゆる隠れ脚候補を補間する脚候補補間部としての機能も持つ。脚候補抽出部14で抽出される脚候補の中心座標は、脚候補中心メモリ21に記憶される。そして、脚候補抽出部14で得られた脚候補のデータ（中心座標のデータ）は、脚候補中心座標分類部15に供給される。

**【0018】**

脚候補中心座標分類部15は、脚候補の中心座標の変化に基づいて、脚候補をクラスタリングする脚候補中心座標分類処理を行う。脚候補中心座標分類部15で得られたクラスタリング結果は、脚候補クラスメモリ22に記憶される。脚候補中心座標分類部15で得られた脚候補のクラスタリング結果は、脚候補状態判定部16に供給される。

**【0019】**

脚候補状態判定部16は、脚候補中心座標分類部15のクラスタリング結果から、脚候補を立脚期と遊脚期に分ける脚候補状態判定処理を行う。立脚期と遊脚期については後述

10

20

30

40

50

する。脚候補状態判定部 16 で分類された立脚と遊脚のそれぞれを示す番号（クラス番号）は、歩行者クラスメモリ 23 に記憶される。脚候補状態判定部 16 で得られた脚候補の立脚期と遊脚期のデータは、歩行者抽出部 17 に供給される。

歩行者抽出部 17 は、脚候補の立脚期と遊脚期のデータに基づいて、立脚期と、その立脚期に続いた遊脚期で構成される立脚・遊脚ペアを生成する歩行者抽出処理を行う。

#### 【0020】

歩行者中心算出部 18 は、立脚・遊脚ペアと、その立脚・遊脚ペアの近傍の立脚とを使って、歩行者中心座標を求める歩行者中心算出処理を行う。すなわち、歩行者中心算出部 18 は、脚候補中心座標分類部 15 が分類する上で必要な歩行者中心の算出処理を、脚候補の座標から行う。歩行者中心算出部 18 で算出された歩行者中心は、歩行者中心メモリ 24 に記憶される。

歩行者特徴量算出部 19 は、歩行者中心算出部 18 で算出された歩行者中心に基づいて、歩行者特徴量を算出する。ここで算出する歩行者特徴量は、例えばタグ番号、歩容、移動方向、移動速度、一時停止状態、停止時間、滞留時間などのデータである。歩行者特徴量算出部 19 で算出された歩行者特徴量は、歩行者特徴量メモリ 25 に記憶される。

#### 【0021】

##### [ 2 . 歩行者の歩行状態の説明 ]

図 2 は、歩行者の左右の脚の立脚期と遊脚期を示す図である。

立脚期は、それぞれの脚が地面と接触した状態を示す。この立脚期では、地面に脚が静止している。遊脚期は、それぞれの脚が地面から離れた状態を示す。この遊脚期では、脚の位置が地面上を移動する。

図 2 に示すように、左脚の立脚期と遊脚期の期間と、右脚の立脚期と遊脚期の期間とが交互にずれている。ここで、左右の脚の片方が立脚期で一方は遊脚期となる区間は、歩行動作に特有のものであり、歩行 1 サイクルの約 80 % を占める。すなわち、図 2 に示すように右遊脚期が約 40 % であり、図 2 には示さないが左遊脚期についても同様に約 40 % であり、合計すると歩行 1 サイクルの約 80 % を占める。

#### 【0022】

ここで、立脚期と遊脚期における足首周りの動きに注目すると、立脚期はほとんど移動せず、遊脚期は歩幅分大きく移動することから、足首周りの動きから脚の状態を判別することができる。

そして、同一人物の 2 つの脚の内の片脚が立脚で一方が遊脚となる場合に、それを同一人物の歩行動作であると推定して、これを判定することができれば、各脚の重心位置から両脚間の中心（重心）位置を推定することができる。その結果、歩行の中心（重心）の軌跡を得ることができる。

#### 【0023】

図 3 は、左右の脚の立脚位置（実線で示す位置）及び遊脚位置（破線で示す位置）を示す。図 3 では、検出した立脚位置及び遊脚位置の中心をプロットした位置を白丸で示し、そのプロットした位置を接続した中心軌跡 m1 を示す。この中心軌跡 m1 が、歩行者の歩行軌跡になる。本例の歩行者認識装置では、このような歩行者の歩行軌跡を検出して、歩行者を認識するものである。

左右の脚は、2次元領域センサ 32 の検出信号から検出される。本例の場合、2次元領域センサ 32 は、地面から約 20 cm の高さの面で、物体を検出して、歩行者の足首を検出する。

#### 【0024】

ところで、図 3 に示すように歩行重心の予測を行う上で、脚の中心（重心）位置を正確に推定することは重要である。しかしながら、実際には、2次元領域センサ 32 から常に左右の脚が見えているとは限らず、2次元領域センサ 32 の検出データで見えていない脚（以下、これを「隠れ脚」と称する）に対しては、測定点を十分に得ることができない。隠れ脚の形状を予測することで隠れ脚の中心を予測することは不可能ではないが、正確な隠れ脚の中心予測を行うのは難しい。そこで本例の場合には、隠れ脚の中心座標を推定す

10

20

30

40

50

ることなく、2次元領域センサ32の検出信号で見えている脚の中心座標のみを参照して、歩行者中心抽出を行っている。

#### 【0025】

例えば2次元領域センサ32から遠ざかる歩行者の場合、遊脚となった左脚は、右立脚の横を通過して、右立脚よりも前の位置で再び立脚となるが、右立脚の横を通過した後は、2次元領域センサ32との位置関係から隠れ脚となる場合が多く、中心座標の検出が安定しない。そこで、本例の場合には、左脚が右立脚近傍へ到達した時点で脚候補軌跡の追従を打ち切り、脚候補中心検出の正確性を確保する。

また、2次元領域センサ32から遠ざかる歩行者の場合、あるタイミングで2次元領域センサ32が左立脚を検出したとき、そのタイミングから後の検出は、左立脚 左遊脚 右立脚となる可能性が高い。

そこで、立脚と遊脚のペアが検出された時点で、そのペアで示される中心座標を保持し、その後、さらに次の立脚と遊脚のペア（左右が逆）が検出されたときに、これらの中心座標を保持し、歩行軌跡を検出する処理を行う。

#### 【0026】

これらの本例の歩行者認識装置が行う歩行者認識処理の詳細を、以下、順に説明する。なお、以下の説明で中心位置や中心軌跡などの「中心」と述べた場合には、距離の二等分などで中心となる位置を求める場合の他に、全体のバランスなどを考慮して重心を求める場合も含まれる。また、「中心」を求めると述べた場合には、「重心」を求める処理も含むと考えてよい。

#### 【0027】

##### [3. 歩行者認識処理の詳細]

図4は、本例の歩行者認識装置が行う歩行者認識処理全体の流れを示すフローチャートである。歩行者認識処理は、歩行者認識装置のCPU11の制御下で実行される。

まず、測定部31が2次元領域センサ(2D-LRF)32から距離データを取得する(ステップS11)。そして、脚候補抽出部14が、距離データから脚候補となる物体を抽出し(ステップS12)、脚候補の座標位置(重心座標位置又は中心座標位置)を算出して保持する(ステップS13)。

#### 【0028】

その後、CPU11は、予め決められた指定実行頻度に該当するか否かを判断する(ステップS14)。ここで、指定実行頻度に該当しない場合には(ステップS14のNO)、ステップS11での測定部31の距離データ取得処理に戻る。そして、ステップS14で指定実行頻度に該当する場合(ステップS14のYES)、脚候補抽出部14が抽出した脚候補から脚候補状態判定部16が立脚の脚候補を抽出し、ここで抽出された脚候補が脚候補クラスメモリ22に保持される(ステップS15)。

#### 【0029】

その後、CPU11は、脚候補クラスメモリ22に保持された立脚が存在するか否かを判断し(ステップS16)、保持された立脚が存在する場合(ステップS16のYES)、脚候補抽出部14は、既検出済みの立脚に対応する遊脚を抽出する。そして、歩行者抽出部17は、脚候補抽出部14で抽出された立脚と遊脚に基づいて、立脚と遊脚とのペアの生成処理を行う(ステップS17)。なお、ステップS16で保持された立脚が存在しない場合(ステップS16のNO)には、ステップS11での測定部31の距離データ取得処理に戻る。

#### 【0030】

そして、CPU11は、立脚と遊脚とのペアに対応する立脚が存在するか否かを判断し(ステップS18)、対応する立脚が存在する場合(ステップS18のYES)、歩行者中心算出部18は、立脚と遊脚との間の中心座標を算出し、中心座標を歩行者中心メモリ24に保存する(ステップS19)。また、ステップS18で対応する立脚が存在しない場合(ステップS18のNO)には、ステップS11での測定部31の距離データ取得処理に戻る。

10

20

30

40

50

## 【 0 0 3 1 】

そして、ステップ S 1 9 で中心座標を計算して保存した後、CPU 1 1 は、各メモリに保存された不要な立脚及び立脚・遊脚ペアのデータを削除する（ステップ S 2 0）。その後、CPU 1 1 は、歩行者認識動作を終了するか否かを判断し（ステップ S 2 1）、終了しない場合（ステップ S 2 1 の NO）、ステップ S 1 1 での測定部 3 1 の距離データ取得処理に戻る。また、ステップ S 2 1 で歩行者認識動作を終了する場合（ステップ S 2 1 の YES）、図 4 のフローチャートの処理を終了する。

## 【 0 0 3 2 】

次に、図 4 のフローチャートのステップ S 1 2 で行われる脚候補抽出処理について説明する。

本例の場合、ステップ S 1 2 の脚候補抽出処理では、2次元領域センサ 3 2 の検出データに基づいて、地面から 20 cm の高さに存在する脚の形状を取得している。地面から 20 cm の高さに存在する脚の形状を取得することは、おおよそ人間の足首周りの形状を取得することになる。したがって、脚候補抽出部 1 4 で行われる脚候補抽出処理では、2次元領域センサ 3 2 の検出データに基づいて、一定の幅より小さい物体を脚候補として抽出し、その中心座標を取得するようにしている。

## 【 0 0 3 3 】

図 5 は、脚候補抽出部 1 4 で行われる脚候補の抽出処理の詳細の一例を示すフローチャートである。

まず、脚候補抽出部 1 4 は、1 フレーム分の測距データ配列の各要素に対して、次の(1)式を適用することにより、配列中の各要素の前後差分を計算し配列 diff として格納する（ステップ S 2 1）。ここで、i は配列の要素番号を表す。なお、ここでは 2次元領域センサ 3 2 から得られる測距配列の要素数が 1081 であり、要素番号 i は 0 から 1080 までの値になる。

$$\text{diff}[i] = \text{測距データ配列}[i+1] - \text{測距データ配列}[i] \cdots (1)$$

但し、 $i = 0, 1, 2, \dots, 1080$

## 【 0 0 3 4 】

その後、脚候補抽出部 1 4 は、前後差分 diff[1080] のうち、 $\text{diff}[i] > \text{EDGE TH}$  ( $i = 0, 1, \dots, 1080$ ) を満たすものを差分エッジとして抽出し、diff edge[x] 配列に格納する（ステップ S 2 2）。ここでは、物体の端点で測距データの値は大きく変化することを利用して、前後差分配列 diff の各要素のうち、絶対値が一定値以上（EDGE TH という変数で定義）となる場所の要素番号を、差分エッジとして取得する処理を行う。

## 【 0 0 3 5 】

次に、脚候補抽出部 1 4 は、ループ処理に移る。すなわち、最初にループ変数 j に 0 を代入し（ステップ S 2 3）、diff edge[j] 及び diff edge[j+1] の値を取得する（ステップ S 2 4）。

その後、脚候補抽出部 1 4 は、条件式  $\text{diff}[a] > 0$  かつ  $\text{diff}[b] < 0$  となる条件を満たすか否かを判断して、正負の差分エッジに挟まれた部分を抽出する（ステップ S 2 5）。

ステップ S 2 5 の条件を満たすとき（正負の差分エッジに挟まれた部分であるとき：ステップ S 2 5 の YES）、脚候補抽出部 1 4 は、端点間の距離が、予め決められた脚のサイズより決まる閾値 TH FOOT より小さいか否かを判断する（ステップ S 2 6）。

ここで、閾値 TH FOOT より小さいと判断したとき（ステップ S 2 6 の YES）、脚候補抽出部 1 4 は、脚候補の中心を算出して、その算出した中心を記憶させる（ステップ S 2 7）。

## 【 0 0 3 6 】

ステップ S 2 7 の処理を行った後と、ステップ S 2 5 で条件を満たさない場合（ステップ S 2 5 の NO）及びステップ S 2 6 で条件を満たさない場合（ステップ S 2 6 の NO）には、脚候補抽出部 1 4 は、ループ変数を 1 つ増加させ（ステップ S 2 8）、次のループに備える。

その後、脚候補抽出部 1 4 は、 $[j == x - 1]$  の終了条件を満たすか否かを判断し（ス

10

20

30

40

50



ステップ S 2 9)、この終了条件を満たす場合(ステップ S 2 9の YES)に処理を終了する。また、ステップ S 2 9で終了条件を満たさない場合(ステップ S 2 9の NO)に、脚候補抽出部 1 4 は、ステップ S 2 3 のループ処理の最初に戻る。

【 0 0 3 7 】

次に、図 5 のフローチャートに示す脚候補抽出処理を、図 6 ~ 図 1 8 を参照して説明する。

図 6 は、2次元領域センサ(2D-LRF)32の検出データから、脚候補を検出する例を示す。図 6 の例では、2次元領域センサ32から離れた箇所に壁が存在し、その壁の手前に脚と物体(脚よりも大きな物体)が配置された状態を示す。図 6 に示す黒丸は2次元領域センサ32の検出データで得られる測距点を示す。図 6 の例では、20個の測距点を示す。この場合、上述した(1)式の  $i$  が 0 ~ 19 になる。

10

【 0 0 3 8 】

図 7 は、図 6 の例の 20 個の測距点 ( $i = 0, 1, 2, \dots, 19$ ) の距離データの例を示す。図 7 の下側に示すデータ列の上段は、測距点  $i$  の要素番号を示し、下段は距離値を示す。ここでは、物体に対応した要素番号  $i = 2 \sim 8$  で、距離 2 が検出され、脚に対応した要素番号  $i = 13 \sim 15$  でも、距離 2 が検出される。また、物体や脚が存在しない箇所では、壁までの距離に相当する距離 4 が検出される。

図 8 は、図 7 とは要素番号の配列を逆(左から右)にしたものである。

【 0 0 3 9 】

この図 8 に示すデータ列に、上述した(1)式を適用することで(但し  $i = 0 \sim 19$ )、図 9 の差分を示す配列  $\text{diff}[19]$  が得られる。

20

ここで、ステップ S 2 2 では、前後差分  $\text{diff}[19]$  のうち、 $\text{diff}[i] > \text{EDGE TH}$  ( $i = 0, 1, \dots, 19$ ) を満たすものを差分エッジとして抽出し、 $\text{diff\_edge}[x]$  配列に格納する。図 9 の例の場合、条件を満たす要素は、要素番号 1, 8, 12, 15 の 4 つであるため、図 10 に示す  $\text{diff\_edge}[4]$  配列が得られる。

【 0 0 4 0 】

次に、図 5 のフローチャートのステップ S 2 3 からステップ S 2 9 までのループ処理について説明する。

ステップ S 2 3 でループ変数  $j$  に 0 を代入し、ステップ S 2 4 で  $\text{diff\_edge}[j]$  及び  $\text{diff\_edge}[j+1]$  の値を取得する処理を、図 10 の  $\text{diff\_edge}[4]$  配列に対して適用する。これにより、脚候補抽出部 1 4 は、1 回目のループで  $\text{diff\_edge}[0] = 1$  と  $\text{diff\_edge}[1] = 8$  を取得し、ステップ S 2 5 の  $\text{diff}[a]$ 、 $\text{diff}[b]$  における変数  $a$ 、 $b$  に代入する。

30

【 0 0 4 1 】

この  $\text{diff\_edge}$  配列は、測距データ配列の前後差分である  $\text{diff}$  配列の変化の大きい部分を抽出したものである。したがって、ステップ S 2 5 で  $\text{diff}[a]$ 、 $\text{diff}[b]$  を参照すると、測距データ中の差分エッジを得ることができる。また、 $a=1$ 、 $b=8$  とすれば  $\text{diff}[1] = 2$ 、 $\text{diff}[8] = 2$  を得る。続いて、ステップ S 2 5 の条件式 ( $\text{diff}[a] > 0$  かつ  $\text{diff}[b] < 0$ ) を適用することで、正負の差分エッジに挟まれた部分を抽出する。

【 0 0 4 2 】

ここでは、2次元領域センサ32から見て遮蔽するものがない物体(脚)の場合には、その物体の左端と右端から正と負の差分エッジがそれぞれ検出されるのに対し、例えば図 11 に示すように、一部が遮蔽される物体(隠れ脚)では、両端ともに負の差分エッジが検出される。

40

すなわち、図 11 の例は、左脚 1 0 0 L と、それより 2次元領域センサ32から見て遠い位置の右脚 1 0 0 R がある例である。この例の場合、2次元領域センサ32から見て 2 つの脚 1 0 0 L、1 0 0 R の一部が重なっている場合には、左脚 1 0 0 L の右端の検出データと右脚 1 0 0 R の左端の検出データとの間に、負の差分エッジが発生する。

したがって、ステップ S 2 5 の条件式 ( $\text{diff}[a] > 0$  かつ  $\text{diff}[b] < 0$ ) を満たす物体のみを抽出すれば、負の差分エッジが発生する状況が排除され、隠れ脚は抽出されなくなる。

50

脚候補抽出部 1 4 は、このようにして抽出された脚候補のそれぞれの中心（重心）座標を求め、その中心座標を順に保持する。

【 0 0 4 3 】

なお、脚候補抽出部 1 4 は、隠れ脚を補間で算出する脚候補補間部としての処理も行う。すなわち、抽出された脚候補の中で、次に説明する遊脚として検出された脚候補がある程度離れて検出された場合には、脚候補抽出部 1 4 は、その遊脚の候補位置を補間して、遊脚の補間位置を求める処理を行う。

【 0 0 4 4 】

次に、脚候補状態判定部 1 6 が、脚候補から立脚を抽出して保持する処理について説明する。

立脚を検出する保持処理では、脚候補の検出処理で一定フレームをスタックして得られた脚候補の中心に対して、クラスタリングアルゴリズム（例えば DBSCAN : Density Based Spatial Clustering）を適用して、密度ベースクラスタリングを行う。そして、脚候補の中心が集まっている箇所を立脚クラスタとして抽出する。すなわち、立脚期には、該当する脚はある程度の期間ほぼ静止しており、その静止状態から立脚期を判別する。

【 0 0 4 5 】

図 1 2 は、脚候補をクラスタリングして、立脚候補を抽出する例を示す。この図 1 2 は、脚候補の中心座標の変化を示す。すなわち、白丸が立脚としてクラスタリングした脚候補を示し、黒丸が立脚でない（つまり遊脚）としてクラスタリングした脚候補を示している。

【 0 0 4 6 】

図 1 2 に示すように、立脚が検出された場合には、立脚クラスタに固有の識別記号（ID）を割り当て、立脚クラスタの中心座標と向き（進行方向）を取得し、これを保持する。また、次に行われる立脚・遊脚ペアの生成の精度を向上させるため、立脚クラスタを構成する点群のうち、時系列的に最も新しいものを、立脚座標として扱う。そして、クラスタリングアルゴリズムでクラスが割り振られなかった脚候補の中心座標は、遊脚の中心座標として扱い、その遊脚の中心座標を保持する。

【 0 0 4 7 】

次に、歩行者抽出部 1 7 が、立脚・遊脚ペアを生成する処理について説明する。

立脚・遊脚ペアを生成する処理では、脚候補状態判定部 1 6 での判定によって保持されている立脚情報をもとに、それぞれの立脚から伸びる遊脚を検出し、立脚と遊脚を結びつける処理を行う。一つの脚が立脚から遊脚に遷移する場合、脚中心座標には連続性があるので、立脚クラスタの終点座標の近傍にある足候補座標を遊脚の始点として結びつけていく。

【 0 0 4 8 】

具体的には、図 1 3 に示すように、立脚クラスタの方向ベクトルと立脚中心から遊脚候補へのベクトルがなす角度（図 1 3（B））を計算し、その角度が閾値以内で、かつ立脚終点座標からのユークリッド距離が最小となる脚候補を、図 1 3（A）に示すように遊脚の始点として結合する。その後は、最近傍点を結合していき、立脚と遊脚のペアである立脚・遊脚ペアを作成し、一定フレーム間保持する。なお、この立脚・遊脚ペアが生成された時点で、立脚を管理する配列から、該当する立脚クラスを削除する。

【 0 0 4 9 】

図 1 4 は、2次元領域センサ 3 2 の検出データ（座標データ）から立脚・遊脚ペアを生成した具体例を示す。この例では、歩行者の進行方向の左右に壁があり、その壁の間の通路を、2次元領域センサ 3 2 から離れる歩行者を検出したものである。

図 1 4 において、大きな丸は立脚期の座標を示し、小さな丸は遊脚期の座標を示す。図 1 4 は、立脚とそれに続いた遊脚が立脚・遊脚ペアとして生成されることを示している。

【 0 0 5 0 】

次に、歩行者中心算出部 1 8 が行う歩行者中心座標の算出処理について説明する。

左右の脚の内の一方向の脚の遊脚・立脚ペアが生成された後、もう一方の脚の立脚（対応

10

20

30

40

50

立脚)が検出されれば、歩行者中心算出部18は、立脚・遊脚ペアの全座標と対応立脚の中点を求め、歩行者中心を算出することができる。具体的には、立脚・遊脚ペアの終点の近傍に別の立脚クラスが生成された場合、歩行者中心算出部18は、これら立脚・遊脚ペアと立脚を同一歩行者のものであると確定して、左右脚の中点座標を計算し、これを歩行者中心とする。なお、この歩行者重心計算が行われた時点で、立脚・遊脚ペア管理配列から、計算に用いたクラスを削除する。

#### 【0051】

図15(A)~(E)は、歩行者が2次元領域センサ(LRF)32から遠ざかる方向に歩行する様子を時系列で表したものである。図15において、「L」は左脚の位置を示し、「R」は右脚の位置を示す。以下、図15(A),(B),(C),・・・と時間が進むごとに、左脚と右脚の位置が2次元領域センサ32から遠ざかっている。

10

#### 【0052】

この図15の状態を2次元領域センサ32で検出したとき、図16に示すように、2つの脚の中で、2次元領域センサ32からより離れている脚が、隠れ脚になって、検出されない可能性が高い。

具体的には、図15(A)の脚位置を検出した図16(A)の検出データでは、左脚「L」が隠れ脚となる。また、図15(B)の脚位置を検出した図16(B)の検出データでは、右脚「R」が隠れ脚となる。以下、一方の脚が隠れ脚となった状態が図16(A)~(E)に示すように継続していく。

#### 【0053】

ここで、本例においては、立脚・遊脚ペアを使って歩行者中心算出部18が歩行者中心座標検出処理を行うことで、隠れ脚を使うことなく、正確な歩行者の中心座標の検出処理を行うことができる。

図17(A)~(E)は、図15(A)~(E)の歩行状態で、この歩行者の中心座標を検出する状態を示す。この図17に従って検出処理状態を説明すると、まず図17(A)に示すように、右足を立脚として検出(「立」として示す位置)し、立脚管理配列に保持する処理が行われる。

20

#### 【0054】

その後、図17(B)に示すように、検出した右足立脚の近傍に存在する右足遊脚を結合して、立脚・遊脚ペアを生成させる。このとき、遊脚座標の結合個数が一定数を超えれば、遊脚・立脚ペアが完成したものとみなし、立脚・遊脚ペア管理配列に保持する。

30

さらに、図17(C)に示すように、右足立脚・遊脚ペアに対応する左足立脚を検出する。この右足立脚・遊脚ペアに対応する左足立脚は、右足立脚・遊脚ペアの近傍に存在する立脚の候補から検出される。

#### 【0055】

右足立脚・遊脚ペアに対応する左足立脚が検出されると、図17(C)に示すように、右足立脚・遊脚ペアを構成する各脚の位置と、左足立脚とを結ぶラインの中点に、歩行者中心を設定し、歩行者中心座標の列を得る。この歩行者中心座標は、歩行者管理配列によって、歩行者ID毎に管理される。この歩行者中心の計算が完了した時点で、脚候補状態判定部16で判定された右足立脚・遊脚ペアのデータは不要となり、立脚管理配列及び立脚・遊脚ペア管理配列から当該クラスを削除する。

40

このようにして、図17(D)に示すように、歩行者中心座標の列が得られる。

そして、図17(E)に示すように、次の立脚・遊脚ペアと、その立脚・遊脚ペアの近傍の立脚との間で、歩行者中心座標を検出する処理が行われる。

#### 【0056】

図18は、実際に歩行者中心の変化を示す歩行経路の検出例を示す。この図18に示す例は、図14に示す例と同様に、2次元領域センサ32から離れる方向に移動する歩行者を補足したものである。図18は、歩行者の移動軌跡が、立脚や遊脚の検出座標に基づいて検出された状態を示している。なお、図18に示す検出座標列w1,w2は、歩行者が歩くときの両側の壁を示している。

50

## 【 0 0 5 7 】

このように歩行者中心座標を時系列で示すことで、歩行者の歩行経路を取得することができる。また、歩行者認識装置の歩行者特徴量算出部 19 は、歩行者中心座標の変化に基づいて、該当する歩行者の歩容、移動方向、移動速度、一時停止状態、停止時間、滞留時間などを算出することができる。歩行者中心算出部 18 で算出された中心座標で示される歩行経路や、歩行者特徴量算出部 19 で算出された特徴量は、例えば CPU 11 の制御で外部入出力インタフェース 13 から外部に出力して表示することができる。

また、本例の歩行者認識装置が、自律移動型ロボットに搭載されている場合には、自律移動型ロボットが移動経路を算出する場合に、周囲の歩行者の歩行経路を避けた移動ルート  
10  
を算出することができる。これにより、自律移動型ロボットは、歩行者の邪魔にならない移動経路で自律移動することができる。

特に本例の場合には、測距データに基づいて一人一人の歩行者を確実に補足して歩行者中心座標の変化を検出できるため、歩行者認識装置の周囲に多数の歩行者が存在する環境でも、その多数の歩行者を同時に補足することが容易にできるようになる。

## 【 0 0 5 8 】

なお、上述した実施の形態例では、図 1 に示す歩行者認識装置として構成した例を示したが、図 4 や図 5 のフローチャートに示す歩行者認識処理を実行するプログラムをコンピュータ装置に実装して、コンピュータ装置が歩行者認識処理を行うようにしてもよい。

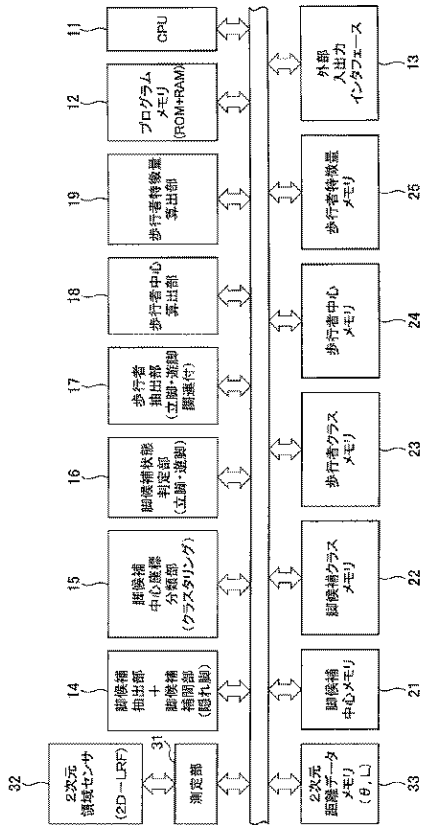
また、上述した実施の形態例では、測距データを取得するセンサとして、2D レーザレンジファインダよりなる 2 次元領域センサを使用した  
20  
が、その他の測距データが得られる測距センサを使用してもよい。

## 【 符号の説明 】

## 【 0 0 5 9 】

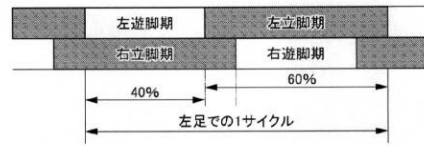
11 中央制御ユニット (CPU)、12 プログラムメモリ、13 外部入出力インタフェース、14 脚候補抽出部 (脚候補補間部)、15 脚候補中心座標分類部、16 脚候補状態判定部、17 歩行者抽出部、18 歩行者中心算出部、19 歩行者特徴量算出部、21 脚候補中心メモリ、22 脚候補クラスメモリ、23 歩行者クラスメモリ、24 歩行者中心メモリ、25 歩行者特徴量メモリ、31 測定部、32 2次元領域センサ、33 2次元距離データメモリ

【図1】



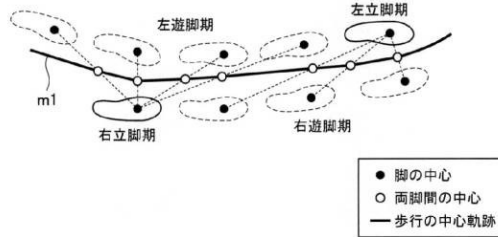
歩行者認識装置の構成

【図2】

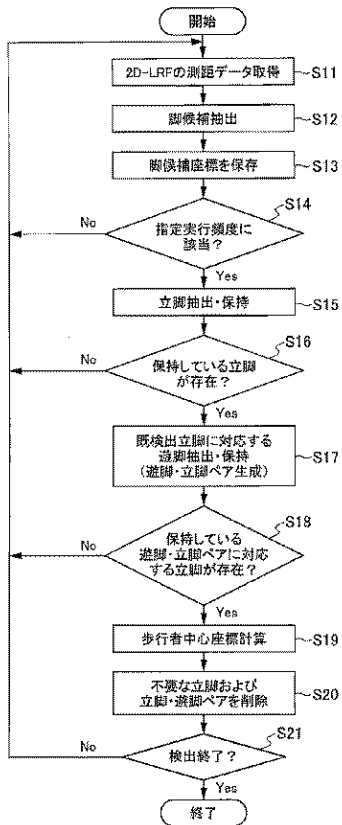


人間の歩行における立脚・遊脚期

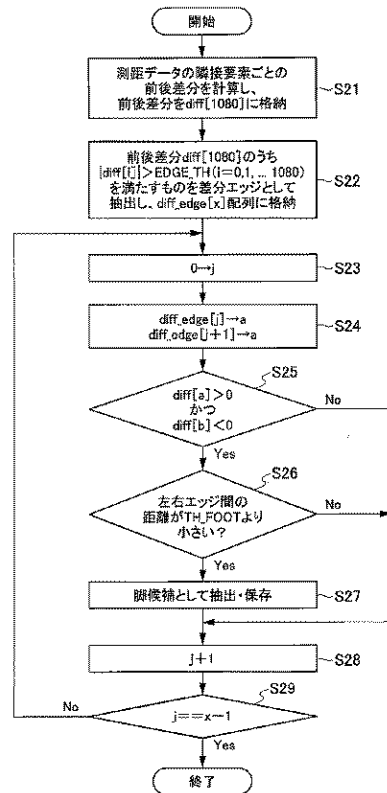
【図3】



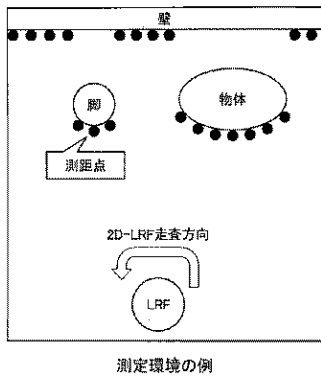
【図4】



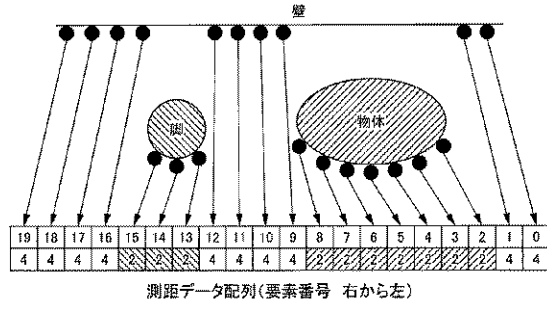
【図5】



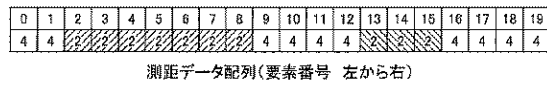
【図6】



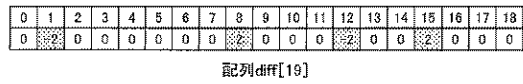
【図7】



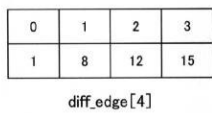
【図8】



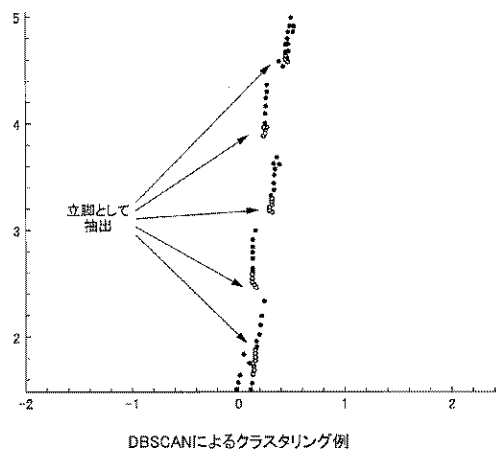
【図9】



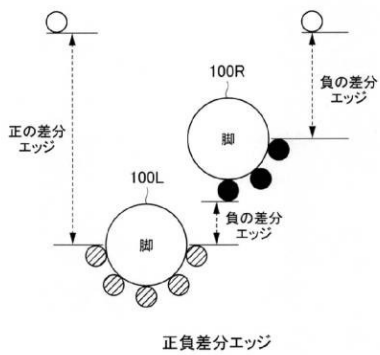
【図10】



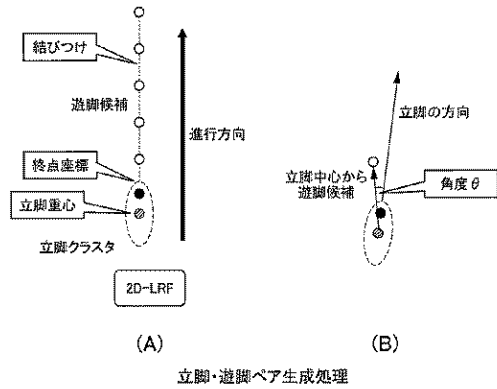
【図12】



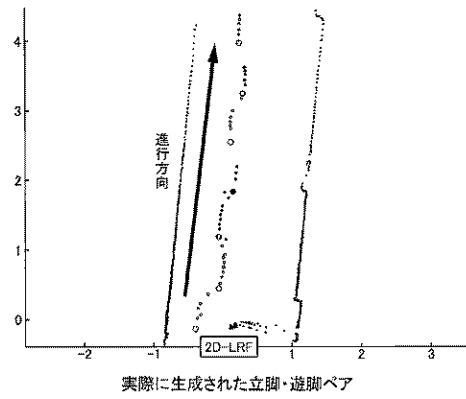
【図11】



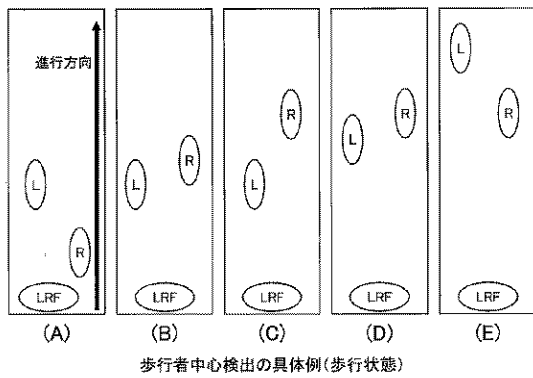
【 図 1 3 】



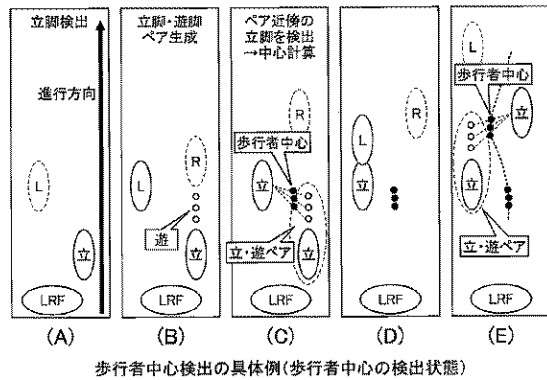
【 図 1 4 】



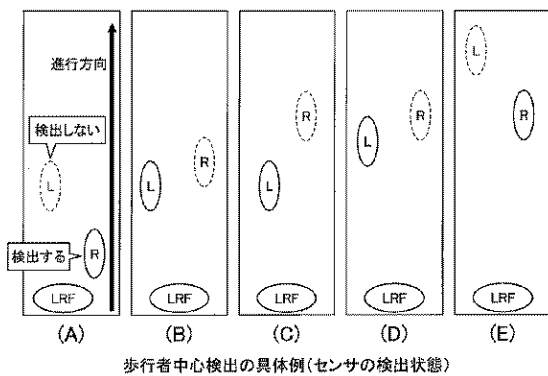
【 図 1 5 】



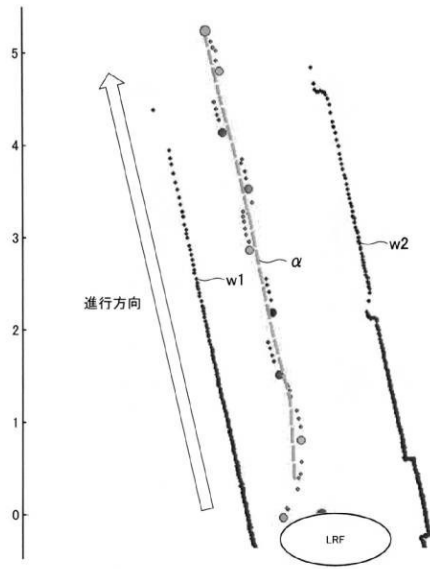
【 図 1 7 】



【 図 1 6 】



【 図 18 】



歩行者中心計算例