

## 複数の IR センサを用いた人物位置追跡システムに関する研究 Human tracking system using multiple IR sensor

末澤 佑規†  
Yuki Suezawa

吉村 宏紀†  
Hiroki Yoshimura

松村 寿枝‡  
Toshie Matsumura

清水 忠昭†  
Tadaaki Shimizu

### 1. まえがき

近年、科学技術の発達により、機器の高性能化が進んでいる。家電製品などにおいても性能が高いだけでなく、周囲の環境を察知して働きかけるような製品も多く出回るようになってきた。こういった製品の多くは「ユビキタス・コンピューティング」と呼ばれているコンセプトのもとに設計されている。

「ユビキタス・コンピューティング」の元では、コンピュータということをも人に意識させることなく人の生活を支援する技術の開発、環境の構築が望ましいとされている。様々なセンサを取りつけ、ユーザが操作せずとも勝手に、且つ最適に動作する「ユビキタス・コンピューティング化」された製品はますます増えることが予想される。

人物位置追跡においても、「如何にしてユーザの位置情報をユーザに負担をかけずに察知するか」という問題が重要になってくる。ユーザが位置検出のためのデバイスをもたずとも、位置情報を取得できる研究として床圧力センサ[1]やステレオカメラ[2]などが挙げられるが、手間や費用がかかっているのが現状である。そこで本研究では比較的安価で手に入れやすい「Kinect」の IR エミッタと IR カメラを用いて人物位置追跡を行うシステムを考案する。またその際生じる人物位置情報の消失を自動で補完し、検出範囲をさらに拡大することを本研究の目標とする。

### 2. 使用機材

#### 2.1 Kinect

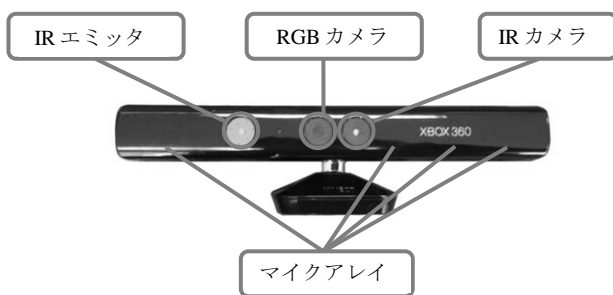


図 1 Kinect

Kinect とはマイクロソフト社が家庭用ゲーム機 Xbox360 用に 2010 年に発売した非接触型コントローラである。図 1 に Kinect の外観と内蔵されているセンサ等を示す。Kinect は主にプレイヤーの動きを読み取って合成するモーションキャプチャという技術を使用しているが、

†鳥取大学大学院工学研究科情報エレクトロニクス専攻

‡奈良工業高等専門学校情報工学科

一般的なモーションキャプチャとは異なり、通常のモーションキャプチャ時に着用する特殊なマーカー付きスーツと、マーカー検出時に使用するトラッカーは必要としない。つまり、何も持たないという理想に近い機能を有してあると言える。IR エミッタからは、特殊な赤外線パターンが正面に向かって照射されており、Kinect 前面の対象物にはそのパターンが投影される。IR エミッタによって照射されたパターンは対象物によって歪み、また距離が離ればパターンはより大きく広がり、これらの情報を統計分析することで Kinect は対象物との距離を計測している。

Kinect センサは様々な環境で物体（人）を認識できるように設計されている。暗い場所に強く、また人物の体の一部が完全に映っていない場合、欠落部分がある程度予測して捉えることも可能である。ただし欠点として、明るすぎる対象物や、反射性・屈折性の高い衣服やアクセサリ、着ぐるみのようなフワフワとした吸光性の高い素材を身につけている場合は深度情報の精度が落ちてしまう。骨格追跡の場合、状況や環境に合わせてノイズが発生する可能性がある。

#### 2.2 Kinect-SDK

Kinect for Windows SDK (以降「Kinect SDK」と表記する)とは、Kinect の技術を PC 上で扱うための開発キットである。研究や教育、および個人用途のアプリケーションを開発する目的での使用が許可されている。Kinect SDK を使うことによって、RGB 画像や奥行き情報、音声情報などを PC に入力することができる。また、奥行き情報から人体の骨格の位置座標を出すことができる。

### 3. Kinect を人物追跡に使う際の問題点

Kinect が人物を認識できるのは赤外線パターンが照射できる範囲である。また、近すぎる場合は人物の認識を行う事が出来ないため、人物の位置を検出できる範囲は奥行き 80cm~4m の範囲に限られる。また、Kinect の前に赤外線を遮るものがあつた場合はそこから後ろは把握できなくなってしまうという問題点がある。

また、Kinect の人物追跡は「TrackingID」によって行われている。Kinect の情報は 1 フレームごとに更新されるため、人物の位置情報も当然 1 フレームごとに変化する。そして過去と現在の人物情報を判断して妥当だと判断した場合 TrackingID は継承される仕組みになっている。この問題点として 1 秒でも検出が

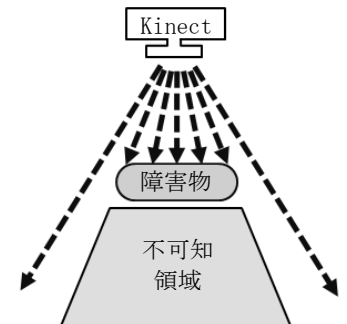


図 2 Kinect を人物追跡に用いる際の問題点

不可になると TrackingID は消失してしまうという問題が挙げられる。図3は TrackingID ごとに色分けを行ったプログラムの動作画面である。TrackingID が変わっていることが分かる。



図3 TrackingID ごとに色分けを行ったプログラム動作画像

## 4. 提案手法

### 4.1 提案手法の概要

Kinect は USB ケーブルを用いてパソコンに接続する事が出来るが、仕様上1台のパソコンに複数台の Kinect を接続し、同時に動作させる事は望ましくない。また、1台では人物を追跡する範囲として狭すぎる。よって検出範囲を拡大するための方法として「複数のパソコンをネットワークで繋ぎデータを送る」という方法を取る[3]。また位置情報の途切れを補う方法として TrackingID の記憶と継承を行えるシステムの設計を目指す。なお本研究では実時間性を考慮し、IR センサからの3次元位置情報のみを用いることにした。

以下に研究目標と提案システムの概要図にまとめる。

- ① 複数の Kinect をネットワークで繋ぐことによって同期させ、検出範囲の拡大を行う
- ② 途切れた人物の位置情報を自動で繋ぐプログラムを作成する (TrackingID の記憶と継承)

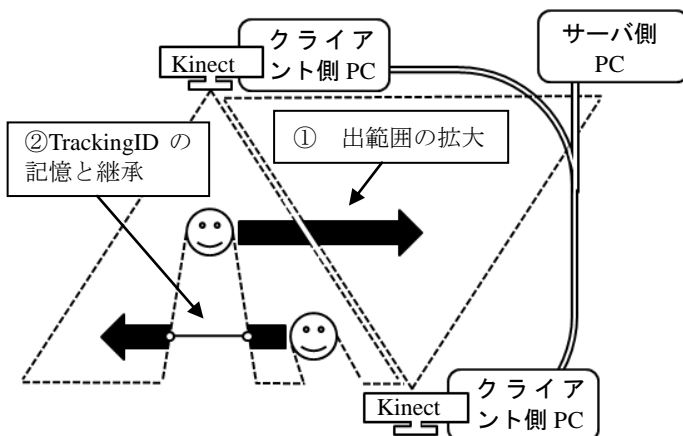


図4 提案システムの概要図

### 4.2 ソケット通信

図4のように Kinect が接続されたパソコンを「クライアント側 PC」、クライアント側 PC から情報を受け一括して扱うパソコンを「サーバ側 PC」として設計する。ネットワークでつながったパソコン間はソケット通信により情報を共有する[4]。0.1秒間隔で各パソコンは人物の XYZ 座標と TrackingID を一か所に送り、サーバ側が処理を行う。その際、複数の Kinect から送られてくる情報を一括して扱えるようにするために、予め位置情報に校正を行った。

### 4.3 評価関数の設定

Kinect が人物を検出した際 TrackingID 等を自作の配列に格納し随時更新する。

検出が途絶えた場合、座標情報と TrackingID を過去データとして別の配列に記憶しておく (本研究ではメモリ配列と呼ぶ)。

新しく人物が検出された場合はメモリ配列の中にそれらしい人物がないか評価関数で照らし合わせ、尤もらしい該当 ID がある場合はその TrackingID を引き継ぐように設計する。

以上の操作を最大検出人数 (6人) でも行えるように設計した[5]。

評価関数として次の4つを作成した。

- ・進行方向評価
- ・歩行スピード評価
- ・消失位置評価
- ・推定位置評価

#### 4.3.1 進行方向評価

Y座標を除いた XZ 空間において、人物が動く 360°の方向を 36分割し、それぞれに番号を割り振り、人物が動いた際はその進行方向の番号が配列に格納されるように設計し随時更新されるようにする。

メモリ配列に格納された番号と新たに検出された人物の番号の類似度によって同一人物であるかの評価を行う。

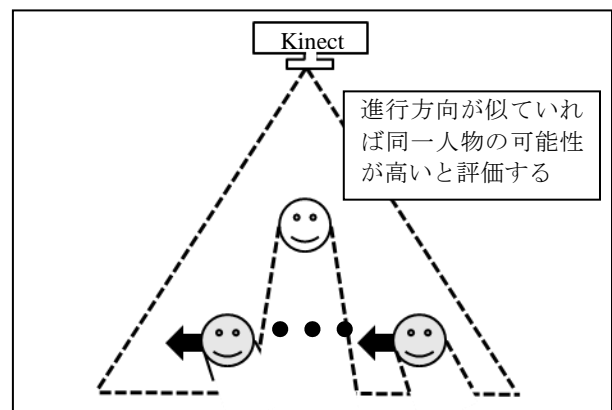


図5 進行方向評価概要

#### 4.3.2 歩行スピード評価

人物の歩行速度の類似度によって重みをつけて評価する。今回 0.1秒ごとにクライアント側から送られる人物位置情報を用いて過去の数値とのユークリッド距離を出した。

$$S = \sqrt{(x' - x)^2 + (y' - y)^2}$$

上記の式により歩行スピードを算出し、この値の類似度によって評価を行う。

#### 4.3.3 消失位置評価・推定位置評価

上記の評価関数で扱った「人物の消失位置」と「歩行スピード」から人物の位置を推定し、随時更新していく。消失位置や推定位置の近くに人物が現れた場合評価を行う。

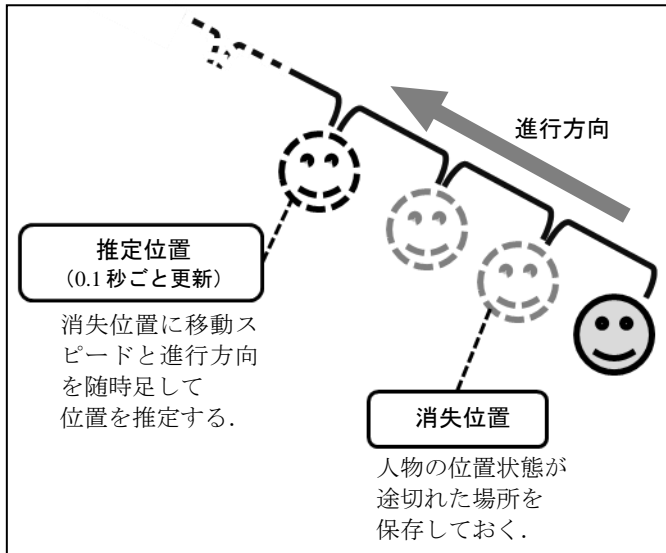


図6 消失位置評価・推定位置評価概要

以上の4つの評価関数で別々に評価を行い、多数決を取り、尤もらしい人物にメモリ配列の情報を継承させる。多数決を取る際、票の割れを回避するため経験的結果から正答率の良かった消失位置評価に2倍の重みをつけて評価している。

### 5. 評価実験

評価実験として以下の3つの実験を行う。

- I. 本システムのソケット通信における通信回数
- II. 複数 Kinect による人物位置範囲の拡大
- III. 評価関数を用いた TrackingID の継承成功率

「I」は Visual Studio2010 上で本システムを1分間実行させた際のクライアント側からサーバ側へ送られるデータの更新回数を調べる。送るデータの種別は float 型の数字である X 座標・Z 座標, int 型の数字の TrackingID, プログラム処理に必要な変数2つの計5つである。

「II」は Visual Studio2010 上で本システムを実行させ、人物の位置情報のうち XZ 座標をファイルに書き出し、検出範囲の拡大が行われているか検討する。Kinect の配置方法は図4のように向かい合わせ、その間を歩くように実験を行う。

「III」は、『歩行者が立ち止まっている人の「前を通る」か「後ろを通るか」』に場合分けしたうえで、検出範囲の人数と消失人数を変化させて各10回ずつ、計100回のすれ違いを行う。

また、「I」と「II」に関してはパソコンの台数の関係上サーバ側のパソコンにも Kinect を繋げ、2台のパソコンを用いて実験を行った。「III」に関しては1台のパソコン上で実験を行った。

## 6. 実験結果

### 6.1 通信回数

Visual Studio2010 上で本システムを1分間実行させた際のクライアント側からサーバ側へ送られるデータの更新回数を図7に示す。

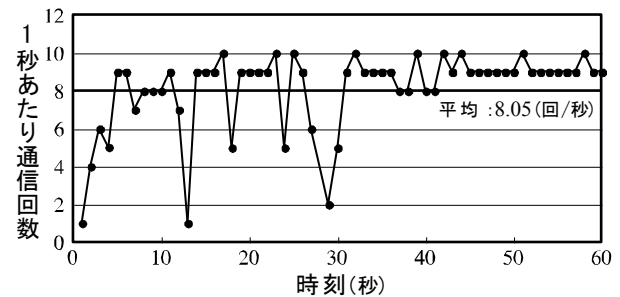


図7 ソケット通信更新回数

サーバ側での1秒あたりのデータ更新速度は平均8.05回であった。プログラム実行直後や、Kinectの動作が滞った際に更新スピードの低下が見られたが、30秒を境に安定した。

### 6.2 人物位置範囲の拡大

人物の位置情報のうち XZ 座標をファイルに書き出した結果を図8に示す。

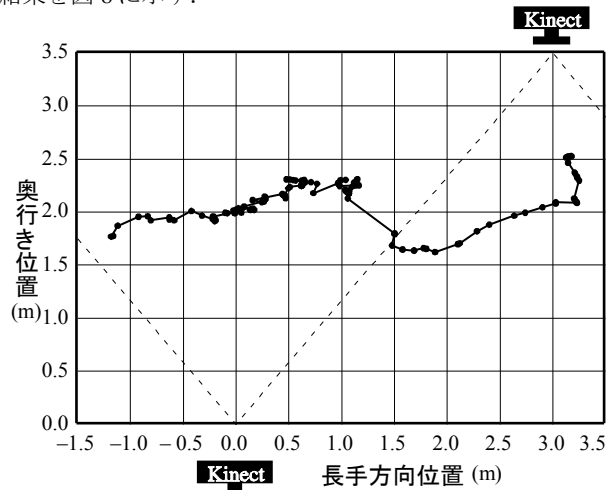


図8 サーバ側で扱った人物位置情報の軌跡

図8の結果より、検出範囲が横方向に4m近く広がっていることから、サーバ側でクライアント側の位置情報を取得出来ていることが分かる。

### 6.3 すれ違いにおける性能評価

評価関数を用いた TrackingID の実験中の映像を図 9 に示す。

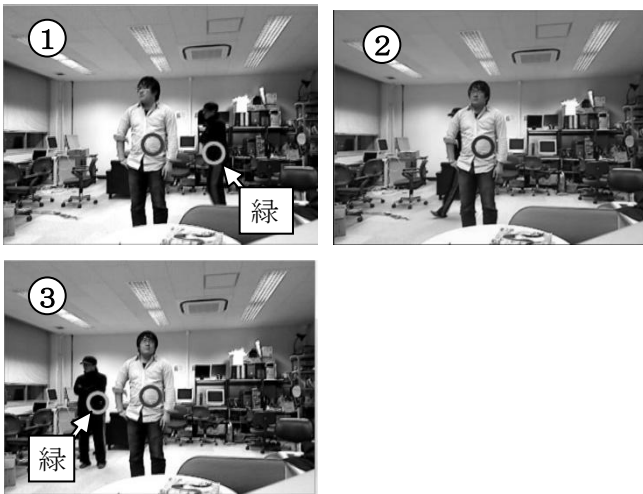


図 9 すれちがいに対し評価関数を用いたプログラム動作画像

表 1 すれ違いにおける TrackingID の継承成功確率

検出人数 消失人数	1	2	3	4
1(歩行者が消える)	100%	100%	100%	90%
1(直立者が消える)		100%	100%	100%
2(歩行者が消える)			50%	30%
2(直立者が消える)				50%

評価関数を用いた TrackingID の継承成功確率に関しては検出範囲内に何人いるかに関わらず消失人数が 1 人の場合、高い確率で成功している。

しかし、同時に 2 人が消えた場合の正解率が下がっている事が分かる。ここでの誤答はすべて個人の TrackingID が交換されるという不具合であった。

### 7. 考察

表 1 のすれ違いにおける TrackingID の継承実験では、消失人数が 2 人の場合に誤答が多く発生した。この原因は評価関数の重み付けの仕方にあると考えられる。今回の実験では進行方向評価等より消失位置評価の方に重みを置いてプログラムを組んだため図 10 のような TrackingID の交換が高確率で起こってしまった。

直立者の位置情報が消失した場合、人物は動かないので消失位置評価に重みをおいたプログラムが有用であると考え

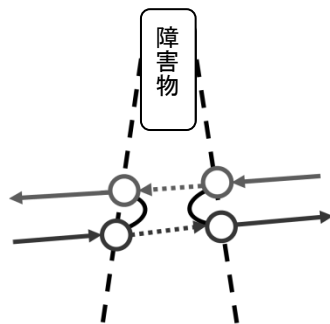


図 10 消失位置評価における誤答例

られる。また 2 人が同じ方向に歩きながら「追い抜き」のようなすれ違いが起こった場合、進行方向評価は有用ではないと考えられる。このように検出範囲内の人間の動きに合わせてどの評価関数を重視するか動的に変化させる事が TrackingID 継承の成功率を上げる事につながると考える。

これまでの結果を以下にまとめる。

- ソケット通信による Kinect 検出範囲拡大は有用である。
- 検出範囲内に人物が複数いる場合でも、すれ違うタイミングが重ならない限り本システムは有用である。
- 評価関数の重み付けは検出範囲内の人物の動き方によって動的に変化させる必要がある。

今後の課題として以下の点を挙げる。

- 今回は検証を行っていない Kinect 間のデータ受け渡しの正答率等を精査する

### 8. 参考文献

- [1] 村北 卓也, 石黒 浩 “複数仮説化周期歩行モデルに基づく床センサネットワークを用いた実時間複数人物追跡”, 電子情報通信学会論文誌. D, 情報・システム J90-D(6), 1555-1567, 2007-06-01
- [2] 原口 真悟, 福司 謙一郎, 熊澤 逸夫, “多数のステレオカメラと色情報を用いた人物追跡手法”, 電子情報通信学会技術研究報告 (20100308)
- [3] 劉斌, 田中二郎, “Kinect サーバおよびサンプルクライアント研究開発 -Kinect HTTP サーバ構築とクライアント用ライブラリの作成-”, 筑波大学大学院博士課程システム情報工学研究科特定課題研究報告書 2012 年 3 月
- [4] MFC 編 DDX の基本 [http://www.g-ishihara.com/\(2013/01/17 最終閲覧\)](http://www.g-ishihara.com/(2013/01/17 最終閲覧))
- [5] 中村薫/齋藤俊太/宮英人” KINECT for Windows SDK” (秀和システム)/(2012)
- [6] 浅井 有美, 山口 達也, 水野 舜, 永瀬 宏, “赤外線センサ群を用いた人物位置特定・追跡システムの検討”, 電子情報通信学会技術研究報告 (20070517)