

人物誤追跡に対応した  
エージェントによる人物追跡システム  
の構築に関する研究

2020年 7月

塩塚 大

# 目次

第1章	序論	1
第2章	人物追跡システムについて	6
2.1	序言	6
2.2	関連研究	6
2.2.1	電波に着目した研究	7
2.2.2	電磁波に着目した研究	8
2.2.3	画像に着目した研究	8
2.2.4	追跡手法に着目した研究	9
2.2.5	周辺環境や行動履歴に着目した研究	9
2.2.6	複数の技術要素を組み合わせた研究	9
2.3	エージェント技術を用いた人物追跡システム	11
2.3.1	システム構成	11
2.3.2	エージェントを用いた人物追跡の利点	13
2.3.3	人物追跡の流れ	13
2.3.4	隣接関係	15
2.4	結言	19
第3章	人物追跡システムの実装	20
3.1	序言	20
3.2	Raspberry Pi およびビーコンセンサ利用した実装	20
3.2.1	システム構成	21
3.2.2	モジュールの実装	21
3.2.3	人物追跡の流れの詳細	25
3.3	実験	27

3.3.1	実験環境	27
3.3.2	実験1：遅い速度で移動する人物の追跡	29
3.3.3	実験2：普段の速度で移動する人物の追跡	32
3.3.4	考察	33
3.4	結言	34
<b>第4章</b>	<b>人物誤追跡への対応</b>	<b>35</b>
4.1	序言	35
4.2	未検出と誤検出	37
4.3	誤検出の具体例	41
4.4	エージェントのグループによる誤追跡への対応	43
4.4.1	提案手法の要点	43
4.4.2	評価値	44
4.4.3	2つの閾値	44
4.4.4	グループ	45
4.4.5	グループ拡張	46
4.5	グループによる追跡の流れ	48
4.6	シミュレーション実験	50
4.6.1	実験環境	50
4.6.2	実験1：評価値の効果の確認	53
4.6.3	実験2：グループ拡張の効果の確認	55
4.6.4	閾値設定のガイドライン	58
4.6.5	考察	60
4.7	結言	61
<b>第5章</b>	<b>人物存在確率の計算手法</b>	<b>63</b>
5.1	序言	63
5.2	グループを用いた追跡および問題点	63
5.3	人物の存在確率による誤追跡への対応	65
5.3.1	人物の移動パターン	65
5.3.2	提案手法の要点	66
5.3.3	サブグループ内存在確率	66

5.3.4	グループ内存在確率 . . . . .	69
5.3.5	存在確率を用いた人物追跡の流れ . . . . .	70
5.4	シミュレーション実験 . . . . .	73
5.4.1	実験環境 . . . . .	73
5.4.2	予備実験 . . . . .	75
5.4.3	実験1：提案手法評価 . . . . .	77
5.4.4	実験2：従来手法との比較 . . . . .	81
5.4.5	考察 . . . . .	83
5.5	結言 . . . . .	84
<b>第6章</b>	<b>結論</b>	<b>86</b>
	謝辞	<b>88</b>
	参考文献	<b>89</b>
	公表論文	<b>96</b>

# 図版目次

2.1	提案システム	12
2.2	人物追跡の流れ	14
2.3	隣接関係	16
2.4	隣接関係の局所化	18
3.1	システム構成	22
3.2	ソフトウェア構成	23
3.3	人物追跡の流れの詳細	26
3.4	人物追跡システムの監視端末表示	28
3.5	ノードの設置例	29
3.6	人物追跡の実験結果	30
3.7	コピーエージェントの派遣結果	30
3.8	ノード No.5 時点のエージェントの派遣結果	31
3.9	人物追跡の実験結果	32
4.1	未検出の例	38
4.2	誤検出の例	39
4.3	隠れ隣接関係	40
4.4	距離のイメージ	41
4.5	同一人物間と異なる人物間の距離の分布	41
4.6	誤検出と未検出	42
4.7	画像間の人物の対応付け	43
4.8	確定閾値と再評価閾値	45
4.9	グループの例	46
4.10	グループ拡張の例	47

4.11	グループによる追跡の流れ	49
4.12	シミュレーションマップ	51
4.13	評価値の効果の確認	54
4.14	グループ拡張の効果の確認	56
4.15	平均グループ拡張回数	57
4.16	閾値を変化させた場合の追跡成功率	59
4.17	距離の分布	60
5.1	グループ拡張	64
5.2	追跡対象の移動パターン	65
5.3	サブグループの例	67
5.4	存在確率による追跡の流れ	72
5.5	シミュレーションマップ	74
5.6	予備実験	76
5.7	エージェント数の上限設定時	77
5.8	存在確率に基づくエージェント削除時	78
5.9	グループ縮小有りの場合の累積 $n$ 位正解率	80
5.10	グループ縮小無しの場合の累積 $n$ 位正解率	81
5.11	エージェント数固定時	82

# 表目次

3.1	機材 . . . . .	21
3.2	ノード間の隣接関係 . . . . .	27
4.1	センサの検出結果と現実の事象の分析結果 . . . . .	37

# 第1章 序論

企業や学校ではセキュリティ対策として、不審者を監視する入退室管理システムなど、様々な監視システムが導入されている。また、東京オリンピックの開催など多くの人が行き交う中において、特定の人物の移動を監視しリアルタイムに追跡したいというニーズが高まっている。ここでの人物の追跡とは、カメラ、GPS (Global Positioning System)、あるいはBLE (Bluetooth Low Energy) ビーコンのような人物を検知するセンサを利用し、追跡したい人物、すなわち追跡対象の移動経路を特定することを指す。特定の人物の移動を追跡することができれば、たとえば、防犯の面では不審者がどこから来てそして現在どこにいるかを監視員は把握することができる。

人物を監視するためのシステムとして、たとえば監視カメラを用いた映像監視システムが挙げられる。映像監視システムでは監視したいエリアに監視カメラを設置し、監視員が監視カメラから受信した映像で追跡対象の存在有無を確認する。映像監視システムは事務所、駅、ショッピングモールのような様々な場所で利用されている。プライバシーの問題からカメラの設置が好まれない場所では、たとえば、介護施設等ではカメラで監視されていることを利用者は好ましく思わず、ストレスをあたえてしまうことが考えられる。そのような場所では、BLE ビーコンタグのような電波を発する小型の機器を身に付け、電波受信機を施設内に配備することで利用者の位置を特定する監視システムが検討されている。IoT (Internet of Things) の発達により、Raspberry Pi のような超小型計算機が安価で購入できるようになり、監視カメラを用いた本格的な監視システム以外にも簡易的に監視システムが構築できるようになってきている。

監視システムにおいて取り分け重要なことは、先に述べたように特定の人物をリアルタイムに追跡することが挙げられる。そこで人物の追跡を実現する手法として、エージェント技術を用いた人物追跡システムの研究が行われている。映像監視システムでは監視員が監視カメラから受信した映像で追跡対象の存在有無を

確認する。エージェント技術を用いた人物追跡システムでは、エージェントが監視員に代わって人物の追跡をおこなう。監視員が映像を確認し人物を追跡していた作業は自動化され、監視員の作業負荷の軽減が期待できる。エージェントによる人物追跡システムは、映像監視システムあるいは映像以外の監視システムにおいて監視員の人物追跡行為を代替するものという位置づけとなる。

著者らはエージェント技術を用いた人物追跡システム<sup>1)2)3)</sup>を提案している。提案しているエージェント技術を用いた追跡では次のようにして人物を追跡する。まず、監視員は追跡対象を検出しているセンサのもとへ最初のエージェントを派遣する。すると最初のエージェントは自身が派遣されているセンサの次に人物を検出すると予測されるセンサへさらにエージェントを派遣する。そして、エージェントが派遣されているどこかのセンサにおいて人物が検出されたとき、さらにその次に人物を検出すると予測されるセンサへエージェントを派遣する。このとき、人物を検出しなかったセンサに派遣されていた古いエージェントは削除する。以上のように、人物の検出をもとにエージェントの派遣と削除を繰り返すことで、人物の移動に追従しエージェントをセンサ間を移動させることができる。エージェントを利用した追跡システムを実現するためには、センサによって人物を検知し特定するための技術、エージェントを効率よくセンサのもとへ派遣するため次にとどのセンサが人物を検出するかを予測する技術等が必要になる。これらの技術要素を組み合わせることにより、たとえば、小型計算機にセンサを搭載し、その小型計算機を監視エリア上の入り口や分岐点といった監視したい場所へ配備すれば、人物の移動にともないその小型計算機上をエージェントが移動する人物追跡システムが容易に構築できる。

エージェント技術を用いた追跡システムには次のような利点がある。まず、ネットワーク上の各計算機の計算資源の利用は、人物を検出している計算機のみに限られて処理負荷を各計算機に分散できる。人物を監視したいというニーズは高まっており、今後センサの設置台数の強化や追跡対象の人数の増加が予想される。そのような場合に、中央サーバで人物の追跡を一括処理するようなシステムでは非常に高価なサーバが必要となる。さらに、特定の計算機が故障した場合であっても、その影響は局所化され他の計算機における追跡には影響を与えず追跡が継続できる。また、エージェントの移動にともなって古いエージェントの情報は削除されるため、情報漏洩のようなセキュリティ上のリスクを軽減することができる。

以上のような利点を踏まえ、多数のセンサを用い、さらに多くの人数を追跡する実用的なシーンを想定すると、エージェントを用いた追跡システムの実用化への価値は高く、本論文ではエージェントを用いた人物の追跡をリアルタイムにおこなうための人物追跡システムの実用化に焦点をあてる。

エージェント技術を利用した追跡システムの実用化にむけて、様々な課題があり解決のための取り組みが行われている。たとえば、あるセンサにおいて人物を検出し損なった場合に適切にリカバリし追跡を継続する研究<sup>2)</sup>が行われている。センサとしてRFID (Radio Frequency Identification) センサを使用する場合、センサの設置場所によって人物を見失う場所、すなわちRFID タグからの信号を受信できない場所がある。その結果、センサ情報を取得できず人物を見失い追跡に失敗する場合がある。そこで、追跡対象を見失うと追跡対象が見失われたセンサと追跡対象が見つかったセンサとの間に新しい接続関係を構築し、エージェントを派遣するための新たなルートを構築する。このような方法によりセンサが人物を見失った場合に対処することができる。一方、センサとしてカメラを用いた場合では、カメラによって撮影された画像から抽出した人物の特徴は常に正確に抽出されとは限らず、人物を一意に特定できない場合がある。たとえば、茶色の髪色の人でも光の強度によって髪の色が黒く識別されることがある。その結果、人物の識別を誤り見逃しや誤追跡を引き起こす可能性があり追跡を継続できない場合がある。RFIDのようなセンサからの信号に固有のIDが含まれ、人物を一意に特定できるときは人物の識別誤りは発生しない。しかし、カメラのような必ずしも人物を正確に識別できないセンサを使用する場合には誤追跡が発生し得る。先行研究<sup>4)</sup>において、たとえば単一視点の顔認証の場合で90.39%、歩容認証の場合で91.14%、これらを組み合わせた複数の特徴による個人認証で99.17%の精度を得ているが、識別誤りは必ず発生し回避することが難しい。

そこで、本論文ではセンサによって人物の識別が適切に為されず誤検出が発生する状況下においても、エージェントを利用したシステムにおいて人物の追跡を継続するための手法を提案する。既存の追跡手法では、人物が正確に識別できるという前提があり、人物が追跡対象であることの尤もらしさ、すなわち距離が0(追跡対象)か1(非追跡対象)かを識別することができた。しかし、識別誤りが発生する状況では必ずしも0か1にはならずその間の値をとる。距離が間の値をとるとき、検出した人物が追跡対象かそうでないかを判定する方法として、ある閾

値を設けその閾値より距離が小さければ追跡対象と判断する方法が一般的である。しかし、追跡対象がある閾値より常に距離が小さくなるとは限らない。距離が逆転した場合に本来追跡すべきではない人物を追跡することや、本来追跡すべき人物を追跡しないという問題が発生する。

この問題に対処するために、本論文では従来提案されていたエージェントによる追跡手法を拡張し、2つの閾値を用いた追跡手法を提案する。1つ目の閾値は追跡対象であると確定するための確定閾値、2つ目の閾値は人物の確定を先送りすることを目的とした再評価閾値である。2つの閾値を用いることで、確定に至らないが確定の見込みがある人物を検出した際には、一時的に確定を保留し別の機会に再評価することが可能になる。また再評価の仕組みを実現するにはエージェントの派遣先を一時的に拡大する必要がある。そこで、従来は人物の移動にともないエージェントが移動していたが、人物がある範囲に居るとみなすエージェントのグループによる追跡を提案する。

エージェントのグループによる追跡により、人物の識別誤りによる見逃しや誤追跡を抑えつつ人物を追跡することができた。しかし、エージェントのグループによる追跡では追跡対象がグループ内には存在しているが、グループ内のどこにいるかまでは考慮していなかった。また、再評価の際にグループを拡張するが、グループが無限に拡張し続ける懸念があった。そこで、グループによる追跡の効率を改善すべくグループ内のどこに人物がいる可能性が高いかの存在確率を計算する手法を提案する。存在確率を求める際には、まずグループをさらに細かいサブグループへ分解し、サブグループ単位で人物の存在確率を求める。そしてサブグループ内で求めた存在確率を集約していくことで、グループ内のどこに人物がいる可能性が高いかを求めた。存在確率の計算により、グループ内のどこに追跡対象が存在する可能性が高いかを知ることができる。追跡対象が存在する可能性が低いセンサをグループによる監視対象から外すことにより、グループの拡張を抑えつつ、効率的に追跡対象を追跡することができる。これらの手法を用いることにより、より現実に即したエージェントを用いた追跡システムの構築が可能になる。

本論文の構成としては、まず2章で既存の人物追跡手法について紹介し、つづいてエージェントを利用した人物追跡について述べる。3章で計算機として超小型計算機を用いたエージェントによる人物追跡の実装例を示す。そして、4章でカ

メラを用いた場合に発生する人物の誤追跡について説明し誤追跡を対応するための追跡手法を提案する。さらに、5章でその手法の利便性を高めるため利用する計算資源を削減し追跡効率を改善した追跡手法を提案する。最後に6章で本論文をまとめる。

## 第2章 人物追跡システムについて

### 2.1 序言

本章では現在における人物追跡システムとしてどのようなものが存在するかまた研究が行われているかを紹介し、つづいて著者らが提案しているエージェントを用いた人物追跡システムについて説明する。

### 2.2 関連研究

人物追跡に関する研究としては、Wi-Fiの信号強度<sup>5)6)7)8)9)</sup>、BLEビーコン<sup>10)11)12)15)</sup>、電子タグ<sup>14)</sup>、GPS<sup>16)</sup>、赤外線センサ<sup>20)21)22)</sup>などの電波や電磁波に着目した研究、カメラや撮影された映像から得られる画像に着目した研究<sup>24)25)26)27)28)29)30)32)</sup>、追跡手法に着目した研究<sup>33)34)35)36)</sup>、周辺環境や行動履歴に着目した研究<sup>37)38)</sup>、複数の技術要素を組み合わせた研究<sup>39)40)41)42)43)45)47)</sup>などが存在する。

それぞれの研究では追跡の精度の向上を狙って、単純にセンサから得られた情報をもとに人物を追跡するのではなく、たとえば得られた情報をもとに学習させることや高度なアルゴリズムを適用し加工することで追跡精度を上げている。

電波や電磁波を用いた手法によって人物の位置は精度よく追跡できるが、カメラを用いた画像ベースの追跡システムが活発に研究されている。これは、Rubioら<sup>31)</sup>が示唆しているように、カメラを用いた追跡では環境内に特別なマーカールとなる装置を埋め込む必要がないためである。そして、視覚が非常に豊富なデータ源を提供し、そこから位置を推定することも可能であることである。たとえば、障害物の周囲をナビゲートすることや、ペットなどの重要なオブジェクトを識別し衝突を回避することにも利用しやすい。

### 2.2.1 電波に着目した研究

Xiaら<sup>5)</sup>, Jiangら<sup>6)</sup>, Pulkkinenら<sup>7)</sup>はWi-Fi信号強度のフィンガープリントを用いて, 人物の位置を推定する方法を提案している. これらの手法は, 事前にフィンガープリントを作成する必要がある. また, 谷内ら<sup>8)</sup>はWi-Fi信号強度のフィンガープリントを定期的に自動で更新する手法と, フィンガープリントを用いた人物の位置推定方法を提案している. 久保田ら<sup>9)</sup>は電波が通過する壁の枚数に応じて電波の減衰幅を学習し位置推定の精度を改善する手法を提案している. 宮崎ら<sup>10)</sup>はBLEビーコンの電波が人体の影響を受ける点を考慮し位置推定を補正する手法を提案している. 駒井ら<sup>11)12)</sup>はデイケアセンターにおける患者の行動履歴の自動生成を目標とし, BLEビーコンの電波をもとに人物の位置推定をおこなっている. さらに, 機械学習アルゴリズム(Random Forests<sup>13)</sup>)を取り入れることで精度の向上をおこなっている. 浅野ら<sup>14)</sup>は視覚障害者の歩行を支援するため対象者の足につけたリーダが床や道路に埋め込まれた電子タグを認識することで正確な位置と進行方向を測定する手法を提案している. 位置の特定にはGPSの利用が挙げられるが屋内やビルの間のような電波条件の悪い環境では精度が出ないという点を指摘している. これらの研究では追跡したい人物が予め無線LANデバイスやBLEビーコンのような装置を所持している必要がある.

杉野ら<sup>15)</sup>は所定の場所に設置したBLEビーコンにより人物が存在するかどうかを判定する方法を提案している. しかし人物を識別することができず, 複数人がいる状況で特定の人物を識別することはできない.

Nirjsonら<sup>16)</sup>は屋内においてGPSを利用した人物を追跡する手法を提案している. GPS信号は地表に到達する時点で非常に弱くなっており室内においては利用することは困難である. そこでGPS信号を屋内で受信するための専用のアンテナの開発と受信した信号から追跡対象の位置を推定する手法を提案している. ただし, 高層ビル内や地下室のようにGPS信号を受信できない場所では利用できない. GPS信号だけではなくWiFi信号等の複数の電波信号を利用した商品<sup>17)18)19)</sup>が多数存在する. これらの商品は追跡対象が電波を受信するための小型装置を所持する必要がある. 子どもの行動の見守り等に利用されている.

### 2.2.2 電磁波に着目した研究

保苺ら<sup>20)</sup>は赤外線センサーを利用した人物の位置を推定する手法を提案している。個人を識別することはできないが追跡対象が特別な装置を持つ必要がなく位置を推定できるという利点がある。Shuaiら<sup>21)</sup>、Suzuuchiら<sup>22)</sup>は机の位置や人の移動方向にもとづいて、限られた人数であれば個人を識別し追跡できる手法を提案している。カメラであれば明るすぎることや暗すぎる場合に正確に人物を識別できない場合があるが、赤外線センサーは光の影響を受けにくいという利点がある。また赤外線センサーを用いて人物の室内行動のセンサデータを採取したデータセット<sup>23)</sup>も公開されている。

### 2.2.3 画像に着目した研究

画像処理による人物の追跡に関する研究として、Isardら<sup>24)</sup>はパーティクルフィルタを用いた人物追跡手法を提案している。Babenkoら<sup>25)</sup>、Zhangら<sup>26)</sup>は単一オブジェクトの追跡精度を向上させるオンライン分類機を提案している。Wenxiら<sup>27)</sup>は高次パーティクルフィルタとオンライン学習を用いて移動経路を予想する手法を提案している。Jinら<sup>28)</sup>はカメラの検出範囲が重複している場合の追跡精度を向上させるグループ構造を提案している。しかし、これらの研究ではカメラの撮影範囲に重なりが無くセンサが離散的に設置されている状況を想定していない。

Choら<sup>29)</sup>はカメラ間の隣接関係を自動的に作成する方法を提案している。しかし、カメラからのデータを収集し管理するための中央サーバを必要とする。Yoonら<sup>30)</sup>はカメラが離散的に配置されており、複数のカメラで撮影範囲を共有しない場合の追跡手法を提案している。しかし、設置されている全てのカメラ間でのデータ共有を必要とする。

鄭ら<sup>32)</sup>はプライバシーを配慮して顔が判別できない低解像度映像を用いた人物の識別手法を提案しており、低解像度映像であっても正解率72%で識別が可能であった。

### 2.2.4 追跡手法に着目した研究

追跡手法に焦点を当てた研究として、中沢ら<sup>33)34)</sup>はカメラが離散的に配置された環境において連続的に人物を追跡する手法を提案している。この手法では各画像処理装置にエージェントを配置しエージェント間の連携により人物の追跡をおこなう。しかし、各画像処理装置のモジュールやパラメータを最新に保つための管理用の特別なエージェントを必要とする。また、Wuら<sup>35)</sup>、Hanounら<sup>36)</sup>は限られた数のカメラで監視範囲を最大化するためのカメラの配置方法について検討している。

### 2.2.5 周辺環境や行動履歴に着目した研究

鈴木ら<sup>37)</sup>はパーティクルフィルタによる追跡をベースとし、レンジセンサから室内の障害物の情報を取得しパーティクルフィルタ単体で推定する位置精度を向上させる手法を提案している。杉村ら<sup>38)</sup>はパーティクルフィルタによる追跡をベースとし、人物が頻繁に行き来する場所や滞留する場所の情報を組み込み追跡精度を高める手法を提案している。実験により追跡対象が一時的に見失われたとしても行動履歴の情報をもとに追跡が継続できることを示している。ただし、人物という括りで履歴を用いており、異なる追跡対象ごとに履歴を使い分けるには至っていない。

### 2.2.6 複数の技術要素を組み合わせた研究

Allettoら<sup>39)</sup>は、博物館において入場者の閲覧を支援するシステムを提案している。本システムでは入場者が展示物に接近すると展示物に応じてガイドの情報を提供する。入場者の位置推定と展示物の場所の特定するために、入場者はBluetoothの電波と画像処理を行うためのWearableな装置を所持する必要がある。

Alejandroら<sup>40)</sup>は、高齢者介護のために無線LANネットワークに基づくナビゲーションシステムを開発している。現在の無線LANデバイスの電波強度に基づく位置決めアルゴリズムでは、人物追跡に必要な精度を達成が難しい点に着目し位置決めアルゴリズムの性能を向上させるために、慣性センサも組み合わせた手法を提案している。無線LANデバイスの電波強度に加え慣性センサから得られる

速度や方位も利用し人物の追跡をおこなっている。

岩崎ら<sup>41)</sup>は、人物の位置を特定するのにレーザレンジスキャナ (LRS) と追跡対象がモバイルカメラを装着し、絶対位置を特定する手法を提案している。位置特定の精度を増すことで拡張現実感 (AR) への活用を狙っている。シミュレーション評価によって最大で83%の精度によってモバイルカメラで撮影された画像を、LRSで検出された人物とを対応付けることができた。

白山ら<sup>42)</sup>は、近赤外線LEDと近赤外線フィルターを装着したビデオカメラを用いて人物を追跡する手法を提案している。提案手法では人物が近赤外線LEDを装着し屋内に設置したビデオカメラで人物を追跡する手法、および人物がビデオカメラを装着し屋内に近赤外線LEDを設置し人物を追跡する手法を提案している。カメラを用いる場合プライバシーの問題を考える必要があるが、ビデオカメラに近赤外線フィルターを装着することで近赤外線のみが撮影されることになりプライバシーの問題が解決される。撮影される映像は近赤外線LEDの光源のみが映るような映像になる。ただし、提案手法では2名の被験者を対象に評価されており追跡対象が増えた場合において手法の改良が必要になる。

小牧ら<sup>43)</sup>は、GPSと無線LANを組み合わせた人物の追跡手法を提案している。複数の異なる装置からのデータを用いる場合どちらの結果を採用するかが課題となる。提案手法ではGPSを用いた位置推定と無線LANを用いた位置推定を用いている。無線LANの位置推定の誤差の予測にディープラーニング<sup>44)</sup>を利用し、誤差の大小でGPSを用いた位置推定を利用するか無線LANを用いた位置推定を利用するかの切り替えをおこなう。適したタイミングで切り替えをおこなうことで追跡精度を向上させている。しかし、ディープラーニングを利用するに当たっては事前に膨大なサンプルデータを用いた学習が必要となる。

森ら<sup>45)</sup>は、Kinect<sup>46)</sup>を利用し人物の歩容を抽出することで人物を識別し追跡する手法を提案している。Kinectは内部にRGBカメラや深度センサー等の複数のセンサを搭載しており、人物の骨格を点座標として取得することができる。提案手法では撮影対象が1人である場合にEER (Equal Error Rate : 等価エラー率) が0.048の精度で識別可能であること示している。

星ら<sup>47)</sup>は、BLEとカメラを組み合わせた位置推定の手法を提案している。BLEのみでの位置推定ではBLEの送信機と受信機の間には遮蔽物がある場合、位置推定の誤差が大きくなるという問題がある。そこでカメラを利用し遮蔽物の有無を

判断し、影響が少なそうな受信機を選択することで位置推定の精度を向上させている。

以上のように人物追跡に関して様々な研究がおこなわれているが、フロアに人物が複数人に存在しかつ多数のセンサを用い人物の識別誤りが発生する状況下における人物追跡システムの検討は十分になされていない。

## 2.3 エージェント技術を用いた人物追跡システム

### 2.3.1 システム構成

著者らはエージェント技術に基づいた自動人物追跡システムを提案している<sup>1)2)3)</sup>。本システムでは監視員に代わってエージェントが追跡対象である人物を追跡する。すべての追跡対象は各エージェントによって自動的に追跡される。監視員は対応するエージェントを監視端末上で確認することによって各追跡対象の位置を知ることができる。

システム構成を図 2.1 に示す。システムは、センサ、ノード<sup>1</sup>、エージェント<sup>2</sup>、および監視端末で構成される。センサとしては、カメラやビーコン等の電波を受信する機器が担う。ノードはセンサと接続されカメラから画像を収集し得られた画像を分析する機能、およびエージェントの実行環境を提供する。ノードは計算機が担う。エージェントはセンサからの情報を利用して人を識別し追跡をおこなうプログラムである。エージェントは画像の分析結果をノードから受け取り追跡対象に沿ってノード間を移動する。追跡対象の位置はエージェントの位置を介して監視端末に表示される。

---

<sup>1</sup>ノードという言葉は P2P ネットワークにおける 1 端点を表す用語である。本システムは中央サーバを必要としない分散型の追跡システムである。そこでネットワーク上における計算機をノードと表現している。

<sup>2</sup>エージェント<sup>48)</sup>という言葉は監視員に代わって自律的にネットワーク上を移動し、人物を追跡するプログラムという意味で用いている。

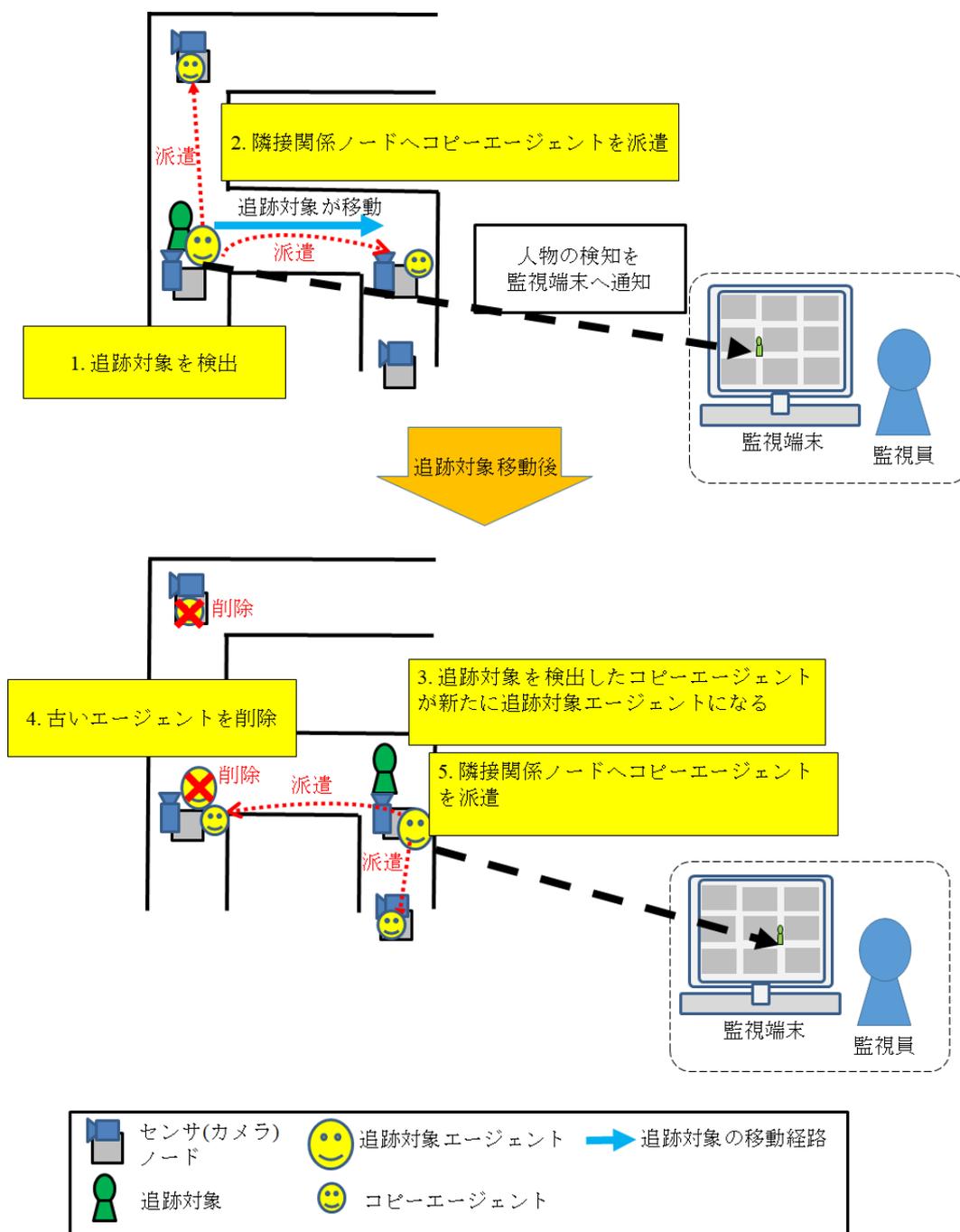


図 2.1: 提案システム

### 2.3.2 エージェントを用いた人物追跡の利点

エージェント技術を用いた追跡システムでは、人物の移動に沿ってエージェントが計算機間を移動することで追跡を実現する。エージェントを計算機上を移動させることで次のような利点がある。

**利点1** ネットワーク上の各計算機の計算資源の利用は、人物を検出している計算機のみ限定でき処理負荷を各計算機に分散できる。人物を監視したいというニーズは高まっており、今後センサの設置台数の強化や追跡対象の人数の増加が予想される。そのような場合に、中央サーバで人物の追跡を一括処理するようなシステムでは非常に高価なサーバが必要となる。対して提案するエージェントによる人物追跡は、処理が各計算機に分散されるため各計算機は低スペックで低価格の小型計算機の利用で構築できる。安価に人物追跡システムを構築できると期待できる。

**利点2** 特定の計算機が故障した場合であっても、その影響は局所化され他の計算機における追跡には影響を与えず追跡が継続できる。さらに、ネットワークを構成するノードはシステムが運用された後からでも追加、削除が可能である。つまり、ノードは自身がどのノードに繋がっているかという隣接するノードの情報のみを保持すれば良く、各ノードはシステム全体としてどのようなノードが存在しているかは知る必要は無い。システムの利用者は容易に監視範囲を拡大することができる。

**利点3** エージェントの移動にともなって古いエージェントの情報は削除されるため、情報漏洩のようなセキュリティ上のリスクを軽減することができる。各ノードはエージェントの実行環境を提供する役割を担う。また、各エージェントは自身が追跡対象とする人物のみの情報を保持し追跡する。自身が対象としない人物の情報は保持しない。ノード間においてやり取りする通信データ量もコンパクトなものとなる。

### 2.3.3 人物追跡の流れ

人物追跡の流れを図2.2に示す。監視員は人物の追跡を始めるとき、その人物の特徴量を有するエージェントをその人物がいるノードへ派遣する。以下ではそれ

## 第2章 人物追跡システムについて

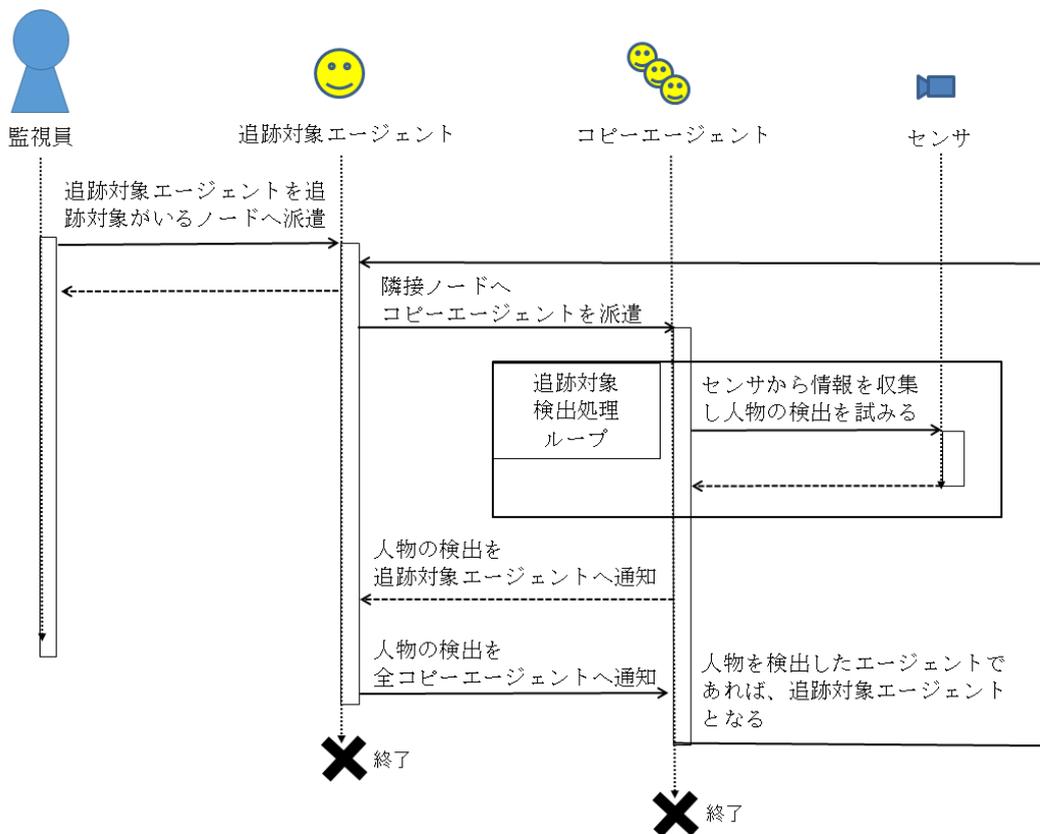


図 2.2: 人物追跡の流れ

を追跡対象エージェントと呼ぶ。追跡対象エージェントはコピーエージェントと呼ぶ自身のコピーをその隣接ノードに派遣する。隣接ノードは次節で説明する方法で決定する。もし隣接ノードが複数個ある場合、自身のコピーを隣接ノードの数だけそれぞれのノードへ派遣する。コピーエージェントはセンサから情報を収集し人物の検出を試みる。コピーエージェントは追跡対象を検出すると人物の検出を追跡対象エージェントへ通知する。追跡対象エージェントは人物の通知を全コピーエージェントへ通知する。追跡対象を検出したコピーエージェントが新しい追跡対象エージェントになる。その後、元の追跡対象エージェントとコピーエージェントを消去する。新しい追跡対象エージェントは、自身のコピーエージェントをその隣接ノードに派遣する。これらのステップを繰り返すことで、エージェントによる人物の追跡を実現する。監視員は追跡対象エージェントの位置を確認することで人物の位置を知ることができる。

### 2.3.4 隣接関係

#### 隣接関係の計算

屋内にセンサを設置する場合を考えると、屋内全体をカバーするようにセンサを設置するにはコストが掛かりすぎる。現実的には入口や交差点など限られた場所にセンサは設置される。このため追跡対象がセンサ間にいる場合ではどのセンサにも検出されない瞬間がある。そこで、センサが追跡対象を次に捕らえる可能性のあるセンサを隣接センサと定義しセンサ間の関係を隣接関係と定義した<sup>1)</sup>。隣接関係は各センサの検出範囲、監視範囲のマップ、および設置されたセンサの位置に基づいて計算する。隣接関係を計算することで追跡対象が次にどのセンサで捕捉される可能性があるかを求めることができる。これにより屋内の全ノードにエージェントを派遣する必要がなくなり、人物が次に現れる可能性があるノードだけにエージェントを派遣するだけで効率的に人物を追跡できる。

図2.3に隣接関係の例を示す。赤い矢印はカメラの隣接関係を表す。灰色の三角がカメラの撮影範囲を表す。図2.3の左側は、 $C_1$ と $C_2$ が、 $C_1$ と $C_3$ に隣接関係があることを表す。図2.3の右側では、 $C_1$ と $C_2$ が隣接関係を持つため、 $C_2$ のみが $C_1$ の隣接関係ノードとなる。隣接関係を計算することで、次に追跡対象を捕らえる可能性のあるカメラを決定できる。次にエージェントの派遣先を計算するにあ

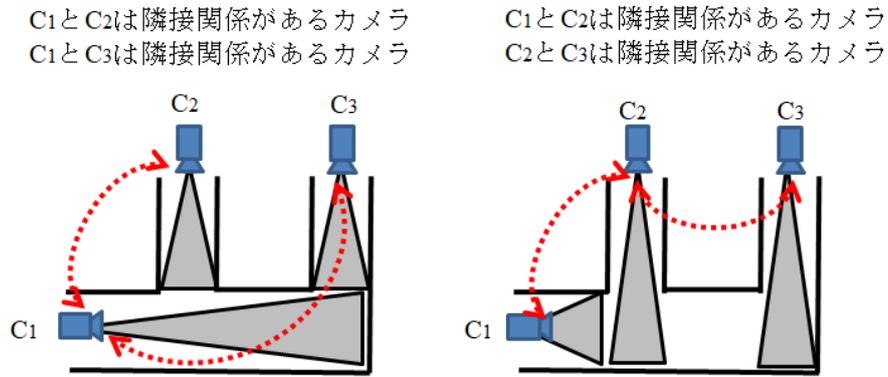


図 2.3: 隣接関係

たって隣接関係の計算方法を説明する。

センサの隣接関係を計算するために、分岐点、センサ点、検出点を定義する。分岐点は移動経路で分岐が発生する場所であり、管理エリアの地図などにより予め設定することができる。センサ点はセンサの設置場所であり、センサの設置および撤去に伴い変化する。検出点は、検出範囲を表現可能とするために、分岐点間、センサ点間、分岐点-センサ点間にそれぞれ一つずつ設ける。

これらの点により、センサの検出範囲を  $|C| \times |P|$  の行列  $X$  とし式 (2.1) のように表現する。ここで、 $C$  はセンサ点、 $P$  は分岐点、センサ点、または検出点とする。

$$X_{ij} = \begin{cases} 0, & \text{センサ } C_i \text{ の検出範囲に点 } P_j \text{ を含まない場合} \\ 1, & \text{センサ } C_i \text{ の撮影範囲に点 } P_j \text{ を含む場合} \end{cases} \quad (2.1)$$

また、分岐点、センサ点、検出点間の接続関係を  $|P| \times |P|$  の行列  $Y$  とし式 (2.2) のように表現する。

$$Y_{ij} = \begin{cases} 0, & \text{点 } P_i \text{ と点 } P_j \text{ が隣り合っていない場合} \\ 1, & \text{点 } P_i \text{ と点 } P_j \text{ が隣り合っている場合} \end{cases} \quad (2.2)$$

このとき、式 (2.3) を計算し  $E_{ij} \geq 1$  となる要素を確認することで、センサ  $C_i$  の撮影範囲から  $(n-1)$  個以内の点を隔てた点にセンサ  $C_j$  の撮影範囲が存在することが確認できる。

$$E = X \times Y^n \times X^T \quad (2.3)$$

さらに行列  $X$ ,  $Y$  からどのセンサの検出範囲にも含まれていない点を削除した行列  $X'$ ,  $Y'$  を作成する. 行列  $X'$  は, 式 (2.4) を満たす, すべての列  $j$  を行列  $X$  から削除することで作成できる.

$$\sum_{k=1}^m X_{kj} = 0 \quad (2.4)$$

また, 式 (2.4) を満たす, すべての  $j$  について, 行列  $Y$  から行  $j$  と列  $j$  を削除する. このとき, 行  $j$  と列  $j$  の削除により接続関係が断たれることを防ぐために,  $Y_{ij} = 1$ ,  $Y_{jk} = 1$  を満たす  $Y_{ik}$  を 1 とする. これにより行列  $Y'$  を作成する. これらの行列  $X'$ ,  $Y'$  から, 式 (2.5) を計算することで隣接センサを求めることができる.

$$E' = X' \times Y' \times X'^T \quad (2.5)$$

### 隣接関係の局所化

センサの設置場所や検出範囲に基づいてセンサ間の隣接関係を計算する方法を説明した. しかし, カメラのようなセンサではパン (左右), チルト (上下), およびズームの操作ができる機器もあり, 監視員の操作によってセンサの検出範囲は変化する. 上述の計算方法では, 検出範囲の変化やセンサの追加・削除が行われた場合には, 行列  $X'$ ,  $Y'$  の内容を全て更新しなくてはならず, 監視対象のフロアが広くなりセンサ台数が増えるほど, 情報取得と隣接関係の計算量のコストが大きくなる. そこで隣接関係の計算を局所化する. それぞれのセンサがそれぞれの隣接関係を求めるためには, 対象となるセンサの検出範囲内の点から他のセンサの検出範囲内の点までに存在する分岐点, センサ点, 検出点が必要となる. そこで, それぞれのセンサはこれらの点だけからなる行列  $Y_c$  を生成する.

$$Y_{c_{ij}} = \begin{cases} 0, & \text{点 } P_{c_i} \text{ と点 } P_{c_j} \text{ が隣り合っていない場合} \\ 1, & \text{点 } P_{c_i} \text{ と点 } P_{c_j} \text{ が隣り合っている場合} \end{cases} \quad (2.6)$$

$P_{c_i}$ ,  $P_{c_j}$  は, 対象のセンサの検出範囲内から隣接センサの撮影範囲内までの経路に含まれる点である. 同様に, これらの点だけからなる行列  $X_c$  を生成する.

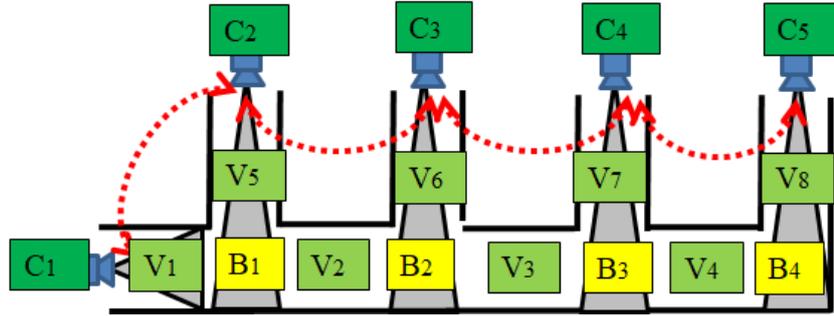


図 2.4: 隣接関係の局所化

$$X_{c_{ij}} = \begin{cases} 0, & C_i \text{の撮影範囲に点 } P_{c_j} \text{を含まない場合} \\ 1, & C_i \text{の撮影範囲に点 } P_{c_j} \text{を含む場合} \end{cases} \quad (2.7)$$

行列  $X_c$ ,  $Y_c$  から, どのセンサの検出範囲にも含まれていない点を削除することで行列  $X_{c'}$ ,  $Y_{c'}$  を作成し, 式 (2.8) により隣接センサを求めることができる.

$$E_{c'} = X_{c'} \times Y_{c'} \times X_{c'}^T \quad (2.8)$$

図 2.4 を用いて局所化の計算の例を示す.  $C_x$  がセンサ点,  $B_x$  が分岐点, および  $V_x$  が検出点を表す. 図 2.4 ではセンサ点が 5 点, 分岐点が 4 点, および検出点が 8 点の合計 17 点となる. そのためセンサの検出範囲を表す行列  $X$  は  $5 \times 17$  の行列となり, センサ点, 分岐点, および検出点間の接続関係を表す行列  $Y$  は  $17 \times 17$  となる. ここで局所化をおこなう. 図 2.4 では赤点線が隣接センサを表しており,  $C_1$  の隣接センサは  $C_2$  である. センサ  $C_1$  の検出範囲を表す行列  $X_{C_1}$  は  $[C_1, C_2] \times [C_1, C_2, B_1, V_1, V_5]$ , つまり  $2 \times 5$  の行列になる. 行列  $Y_{C_1}$  は  $5 \times 5$  となり計算に必要な行列の要素を削減し局所化することができる. 監視範囲が広くなりセンサ点, 分岐点, および検出点が増加した場合, 行列  $X$  および行列  $Y$  のサイズは大きくなり隣接関係を計算するコストが増加する. しかし, 先に示したように計算を局所化することで隣接関係を計算するコストを低く抑えることが可能となる.

## 2.4 結言

本章では現在における人物追跡システムとしてどのようなものが存在するかまた研究が行われているかを紹介し，そして著者らが提案しているエージェントを用いた人物追跡システムについて説明した．提案するシステムにおいてエージェントがどのように人物を追跡し，さらに本システムにおいてエージェントの派遣先をどのように計算するかを説明した．次章では本システムを実装し屋内に設置し実験した結果を説明する．

## 第3章 人物追跡システムの実装

### 3.1 序言

本章ではエージェント技術を用いた人物追跡システムの超小型計算機での実装例を説明する。IoTの発展により超小型計算機が安価に入手できるようになってきた。Raspberry Pi<sup>49)</sup>のような超小型計算機はネットワーク接続やセンサと組み合わせた利用が可能である。さらに、隣接する機器同士の連携や機器の増築が容易である。超小型計算機を用いて人物追跡システムを実装することで、拡張性および可用性が高いシステムを低コストで実現できる。実装したシステムの有効性を確認するための評価実験をおこない超小型計算機においてどの程度自動で人物が追跡できるかを確認した。実際の事務所のフロアに人物追跡システムを設置し実験をおこなった。エージェント技術によって人物が高い精度で追跡できることを確認した。

実装例においてはセンサとしてビーコンセンサを利用しているが、実装の枠組みとしてはセンサに依存しないものとした。そのため、ビーコンセンサ以外にたとえばセンサとしてカメラも利用可能である。今回の実験ではカメラは未使用である。

以降の構成は次の通りである。3.2節で人物追跡システムの実装例を説明する。3.3節で実装したシステムを用いた評価実験について説明し、3.4節でまとめを述べる。

### 3.2 Raspberry Pi およびビーコンセンサ利用した実装

本節では人物追跡システムの超小型計算機を用いての実装方法を説明する。

表 3.1: 機材

機器	分類	詳細
Raspberry Pi	モデル	Raspberry Pi 2 Model B
	OS	Raspbian GUN/Linux 8
	CPU	ARM Cortex-A7 900MHz
	メモリー	1GB RAM
	Java	バージョン 1.8.0
Bluetooth ドングル	-	PLANEX BT-Micro4
無線 LAN 子機	-	BUFFALO WLI-UC-GNM2
ビーコンタグ	-	Aplix MB004
ルーター	-	NEC AtermWG1800HP2
監視端末	モデル	Dynabook 734/M
	OS	NEC Windows7
	ブラウザ	IE11

### 3.2.1 システム構成

システム構成を図 3.1 に示す。ノードは超小型計算機である Raspberry Pi を用いて実装した。1つの Raspberry Pi が1つのノードとして機能する。具体的には表 3.1 の機材を使用した。Raspberry Pi に Bluetooth ドングルを接続し Bluetooth<sup>50)</sup> によってビーコンタグからの電波を検知する。また、無線 LAN 子機を接続しルーターと Wi-Fi で通信を行う。ノード間の通信はルーターを使用し Wi-Fi で無線通信を行う。人物はビーコンタグを所持し、Raspberry Pi に接続した Bluetooth ドングルでビーコンタグからの電波を検知する。エージェントの追跡状況を確認するための監視端末を 1 台設置した。

### 3.2.2 モジュールの実装

ソフトウェア構成を図 3.2 に示す。モジュールは大きく 3 つあり、全てのノード上に図 3.2 の上 2 つのベースモジュールおよびエージェントモジュールを配置する。これらのモジュールを搭載されたノードはセンサからのデータの受信やノードへエージェントの派遣が可能となる。そして、任意の 1 台に図 3.2 の最下部の監

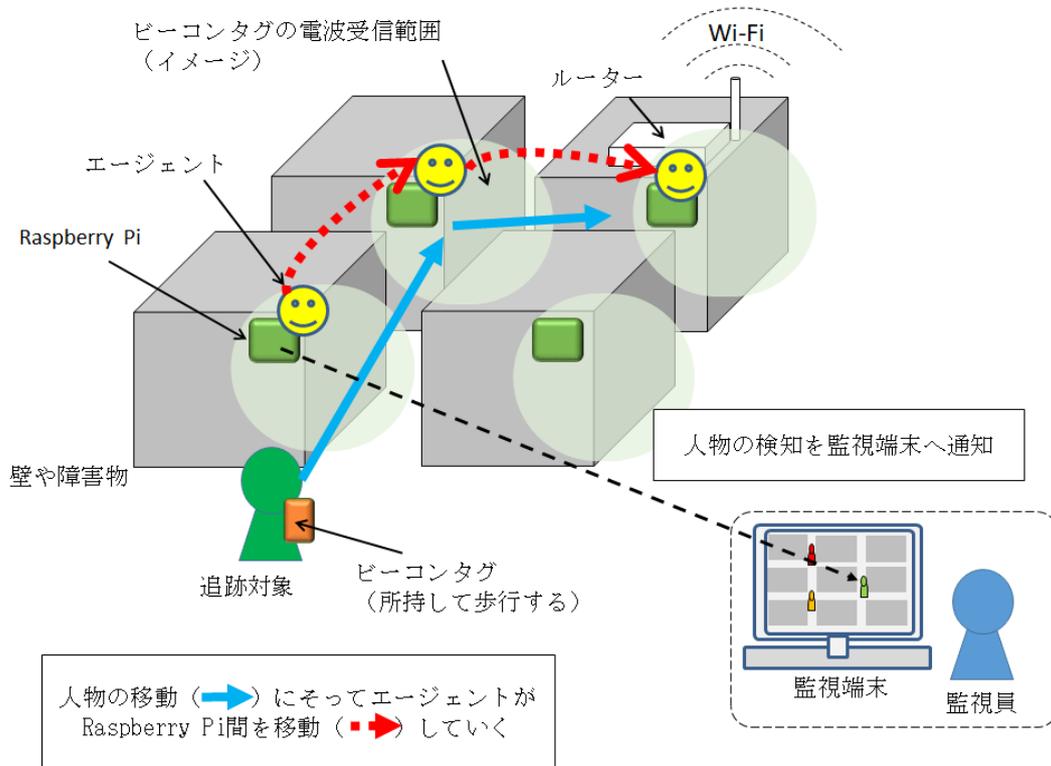


図 3.1: システム構成

### 第3章 人物追跡システムの実装

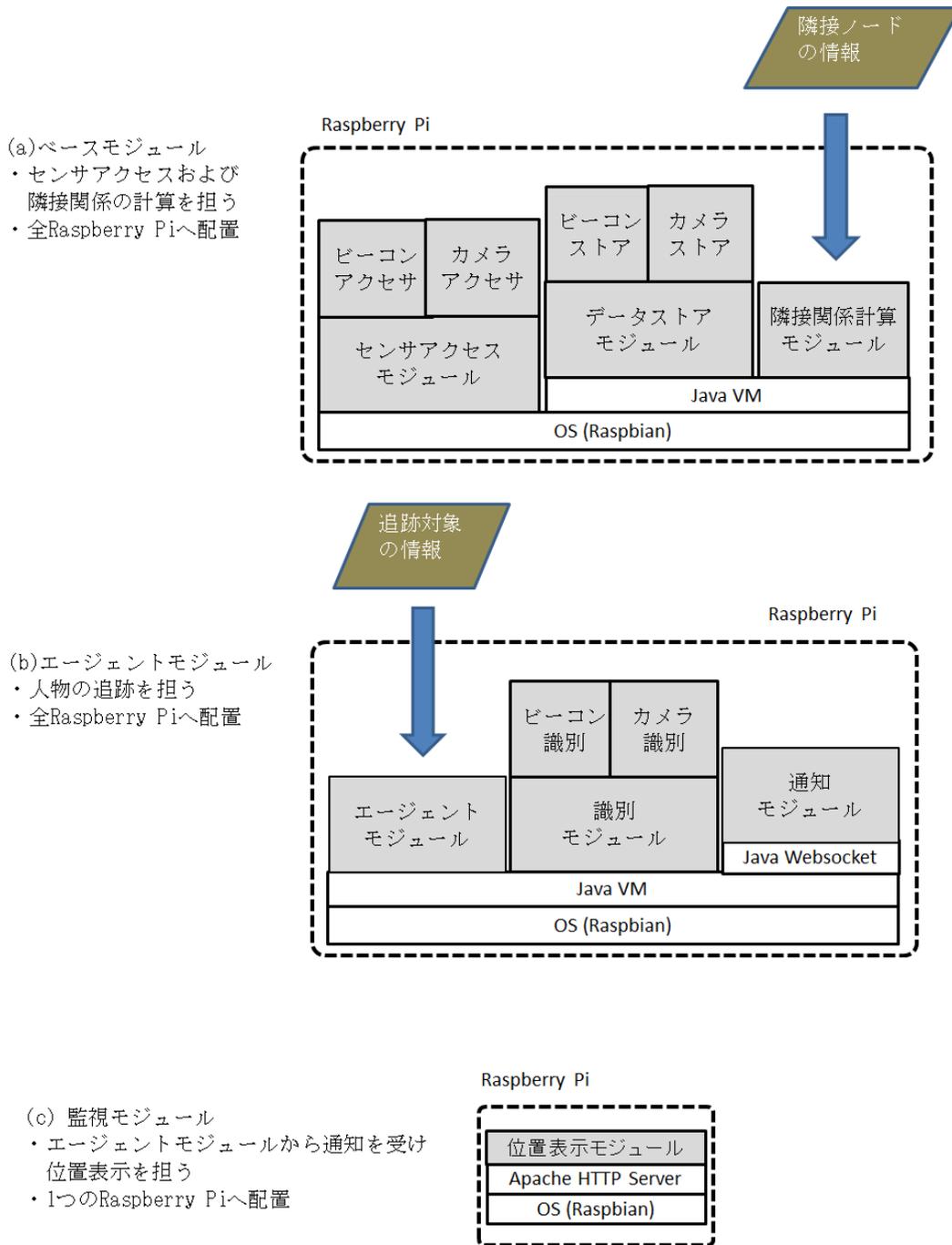


図 3.2: ソフトウェア構成

視モジュールも配置する。監視端末のブラウザから人物の位置取得のリクエストを受け、レスポンスとして人物の位置を表す追跡対象エージェントやコピーエージェントの位置をフロアのマップ上にプロットし返す。

#### (a) ベースモジュール

エージェントによる追跡を実現するための共通的な機能を実装する。下記3つのモジュールによって実装する。

**センサアクセスモジュール** センサと直接通信しデータを取得する機能を実装する。ビーコンアクセサモジュールではビーコンタグから電波を受信する機能を実装する。オープンソースの Bluetooth スタックプロトコルライブラリ BlueZ<sup>51)</sup> を使用し C 言語で実装した。ビーコンタグからの電波を受信し、受信した電波をデータストアモジュールに送信する。カメラアクセサモジュールではカメラから画像を受信する機能を実装する。

**データストアモジュール** センサアクセスモジュールで受信したデータを読み込む機能を実装する。ビーコンストアモジュールではビーコンアクセサモジュールに TCP ソケットで接続し、受信したデータをエージェントモジュールに送信する。カメラストアモジュールではカメラアクセサに接続し、同様に受信したデータをエージェントモジュールに送信する。

**隣接関係計算モジュール** 隣接関係を計算しエージェントの派遣先を決定する機能を実装する。入力として隣接関係のノードに接続するための情報を必要とする。たとえば隣接関係ノードの IP アドレスが必要となる。

#### (b) エージェントモジュール

エージェントによる追跡を実現するための機能を実装する。下記3つのモジュールによって実装する。

**エージェントモジュール** 以下の3つの機能を実装する。

1. 任意の Raspberry Pi と人物の特徴情報を指定しエージェントを派遣する。入力として追跡対象の情報を必要とする。人物ごとに採番した人物 ID とビーコン ID のペアの情報を必要とする。

2. データストアモジュールからセンサのデータを読み込む。
3. 隣接関係計算モジュールより隣接関係を取得しエージェントを Raspberry Pi に派遣する。派遣時にエージェントを表現するオブジェクトをシリアルライズし他の Raspberry Pi に送信する。到着時にデシリアルライズしエージェントの移動を実現する。

**識別モジュール** データストアモジュールからデータを読み込み人物の識別をおこなう。ビーコン識別モジュールではビーコン ID とエージェントモジュールがもつ人物 ID から人物の識別をおこなう。たとえば、ビーコン ID と人物 ID が一致していれば、ビーコンセンサによって検出した人物が追跡対象であると判断する。ビーコン ID と人物 ID が異なっていれば、ビーコンセンサによって検出した人物が追跡対象ではないため何もしない。カメラ識別モジュールでは画像から人物の識別をおこなう。

**通知モジュール** WebSocket<sup>52)</sup> を利用しエージェントがいる Raspberry Pi の IP アドレスおよび人物の ID の情報を監視画面アプリケーションに通知する。

### (c) 監視モジュール

WebSocket を利用しエージェントモジュールから通知される追跡対象エージェントやコピーエージェントの位置を監視端末のブラウザ上の地図に表示する。地図表示にはオープンソースの Openlayers<sup>53)</sup> を利用した。

### 3.2.3 人物追跡の流れの詳細

本実装を用いた人物の追跡の流れを図 3.3 に示す。まずシステムの利用者はエージェントモジュールを利用し、指定した Raspberry Pi に人物の特徴情報をエージェントに設定し追跡対象エージェントを派遣する。特徴情報は追跡対象の人物識別するための ID と、所持するビーコンタグの ID のセットとする。追跡対象エージェントは隣接関係計算モジュールから、自身が派遣されているノードの隣接関係ノードを取得する。追跡対象エージェントは隣接関係ノードへコピーエージェントを派遣する。コピーエージェントはデータストアモジュールからビーコンセンサの情報取得する。利用所が Raspberry Pi に接近すると、Bluetooth ドングルはビーコンタグの電波を検出し付近に人物が接近してきたことを検知する。識別モジュー

### 第3章 人物追跡システムの実装

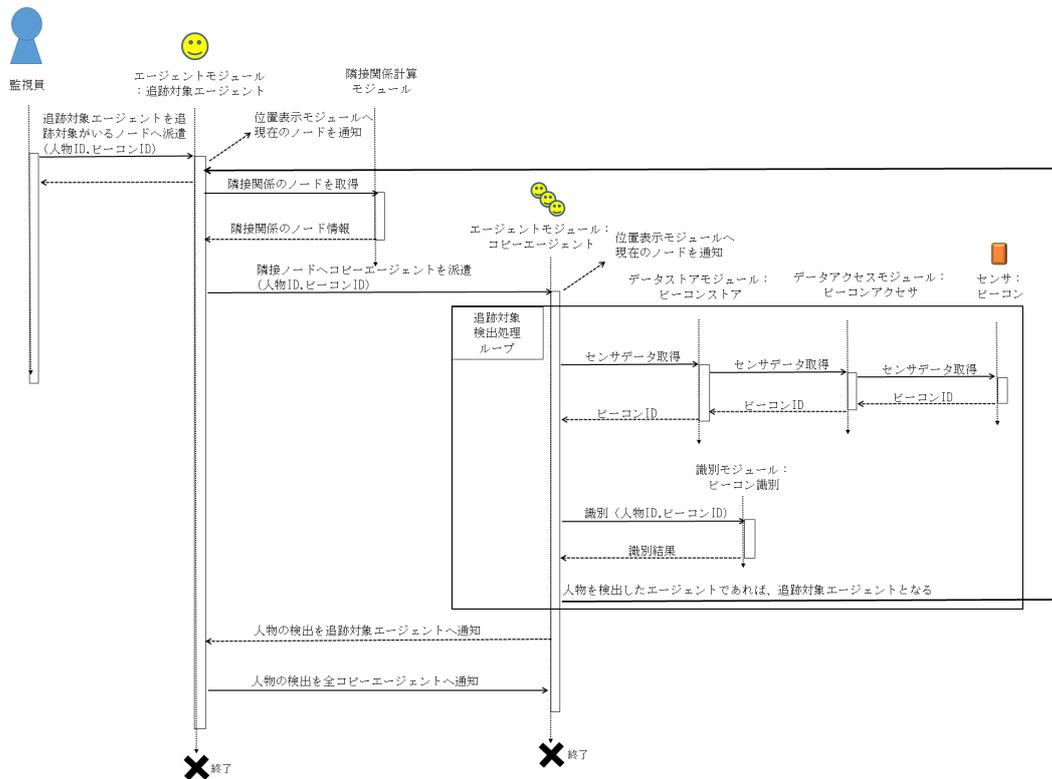


図 3.3: 人物追跡の流れの詳細

表 3.2: ノード間の隣接関係

ノード番号	隣接関係にある監視ノード
1	2, 3
2	1, 3
3	1, 2, 4, 5, 6, 8, 9
4	3, 5, 6, 8, 9
5	3, 4, 6, 8, 9
6	3, 4, 5, 7, 8, 9
7	6, 8
8	3, 4, 5, 7, 9
9	3, 4, 5, 6, 8

ルを利用し受信した電波を解析しエージェントが追跡するビーコンタグの ID と一致する場合、追跡している人物と判断する。以降では2章で説明した人物追跡の流れの通りにエージェントの派遣と削除を繰り返し人物を追跡する。

### 3.3 実験

前節で説明した人物追跡システムを実際に事務所のフロアに導入しエージェントによって人物の追跡が想定通り行われるかを確認した。

#### 3.3.1 実験環境

##### マップ

図 3.4 に監視端末のブラウザでフロアを表示した様子を示す。表示は実際のフロアをデフォルメして表示したもので、灰色の格子が壁、白色が通路、緑色がセンサ検知範囲を表し、数字が書かれた緑色の丸がノードであり Raspberry Pi を表す。フロアは横が 30m, 幅が 25m のサイズである。ノードの設置の様子を図 3.5 に示す。ノードは壁や棚に設置した。フロアに合計 9 台設置している。フロアの中央付近にルーターを 1 台設置している。表 3.2 にノード間の隣接関係を示す。たとえば、ノード番号 1 番に 2 番と 3 番のノードが隣接していることを表す。エー

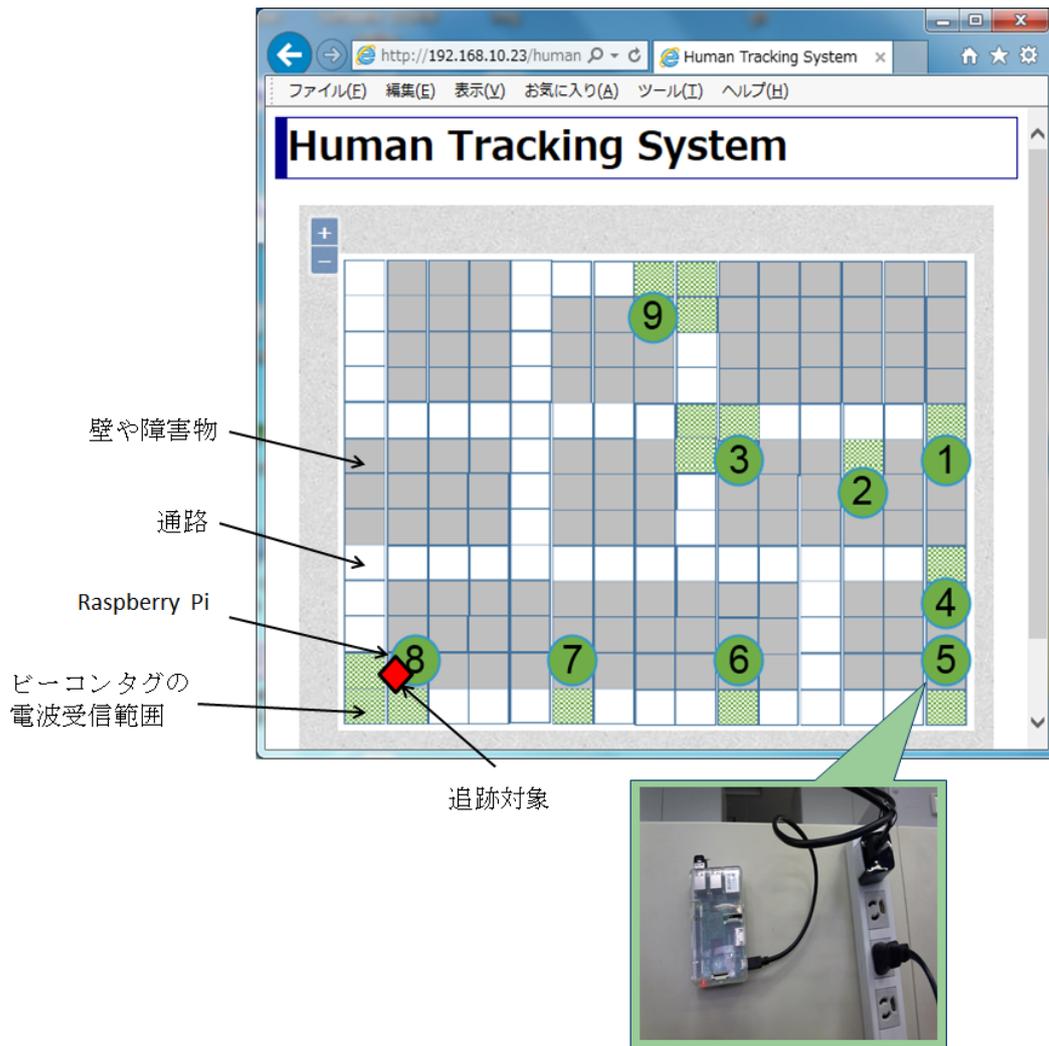


図 3.4: 人物追跡システムの監視端末表示

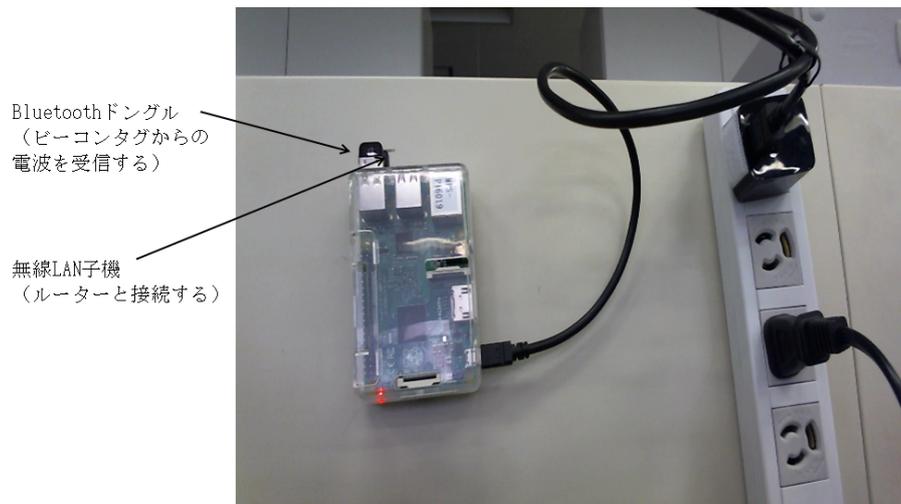


図 3.5: ノードの設置例

エージェントがノード番号1番で人物を検出した場合、そのエージェントはノード番号2番と3番のノードへコピーエージェントを派遣する。

### 人物の移動

追跡対象の人物はビーコンタグを所持しフロア上を歩行する。正解データとして歩行の際にノードに到着したときの時刻とノードの番号を記録する。同時に表示端末上での人物の位置と時刻を記録する。監視端末上での人物の位置が正解データと近ければ、エージェントによって正確に追跡できていると判断できる。実験は下記の通り2回実施した。いずれも、人物の移動に伴いどれだけエージェントが正確に追跡できるかを確認することを目的としている。

- 1回目：追跡対象の人物がビーコンタグを手に持ちゆっくり歩行する。ノード付近で数十秒待機しセンサがビーコンタグの電波を確実に検知してから次のノードに移動する。
- 2回目：追跡対象の人物がビーコンタグを手に持ち普段の速さで歩行する。

### 3.3.2 実験1：遅い速度で移動する人物の追跡

1回目の実験結果を図3.6に示す。人物はノードを以下の順番で移動した。

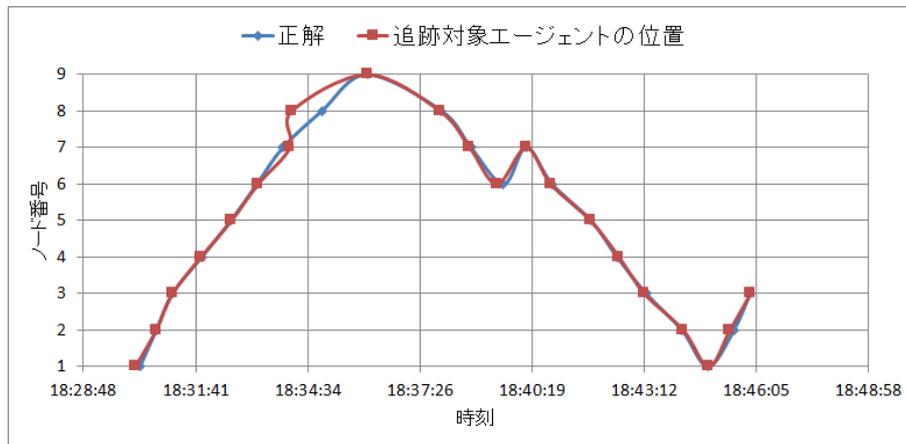


図 3.6: 人物追跡の実験結果

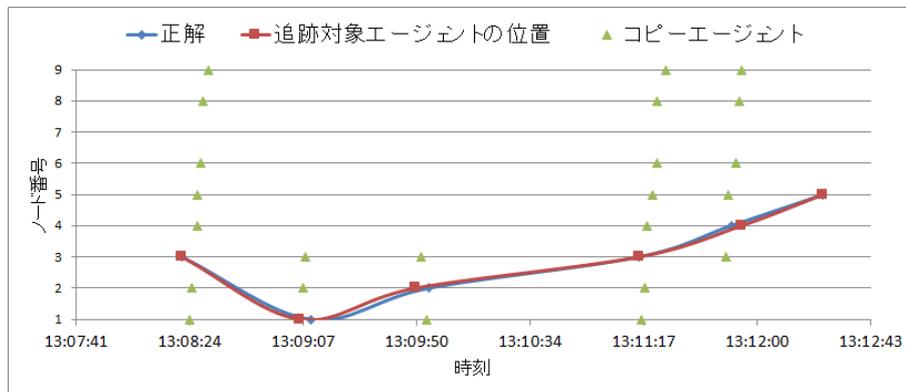


図 3.7: コピーエージェントの派遣結果

[人物が移動したノード番号]

1 ⇒ 2 ⇒ 3 ⇒ 4 ⇒ 5 ⇒ 6 ⇒ 7 ⇒ 8 ⇒ 9 ⇒ 8 ⇒ 7 ⇒ 6 ⇒ 7 ⇒ 6 ⇒ 5 ⇒ 4 ⇒ 3 ⇒ 2 ⇒ 1  
⇒ 2 ⇒ 3

1 回目では、人物の移動に伴い追跡対象エージェントによって人物が追跡できていることが分かる。また、2 章で示したアルゴリズムにより追跡が行えたかを確認した結果を図 3.7 に示す。図 3.7 ではノードからコピーエージェントを派遣した結果を示す。人物はノードを以下の順番で移動した。

[人物が移動したノード番号]

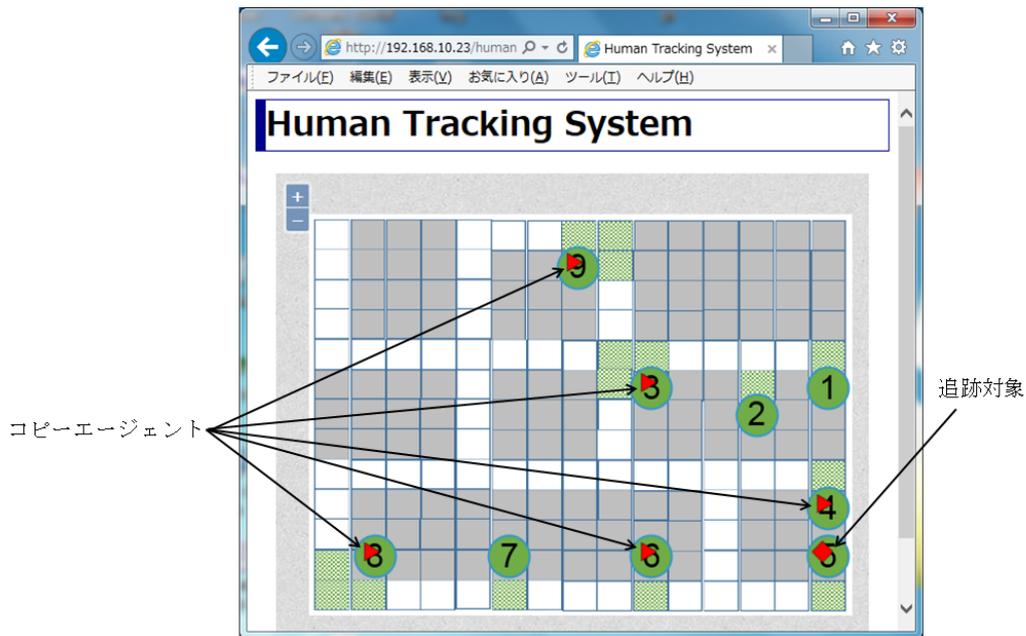


図 3.8: ノード No.5 時点のエージェントの派遣結果

$3 \Rightarrow 1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 4 \Rightarrow 5$

図 3.7 の三角で示した点がコピーエージェントを表す。同時間列に並ぶコピーエージェントの点が、その時点のひとつの前の追跡対象エージェントが派遣したコピーエージェントを表す。コピーエージェント存在するノードに人物が移動していることから、正しい隣接関係で追跡できたことが分かる。

実際の監視端末表示の例として、図 3.8 に人物がノード No.5 に到着した時点での監視端末表示を示す。ノード No.5 上の赤色の四角が人物を表し、No.3,4,6,8,9 の赤色の三角がコピーエージェントを表している。

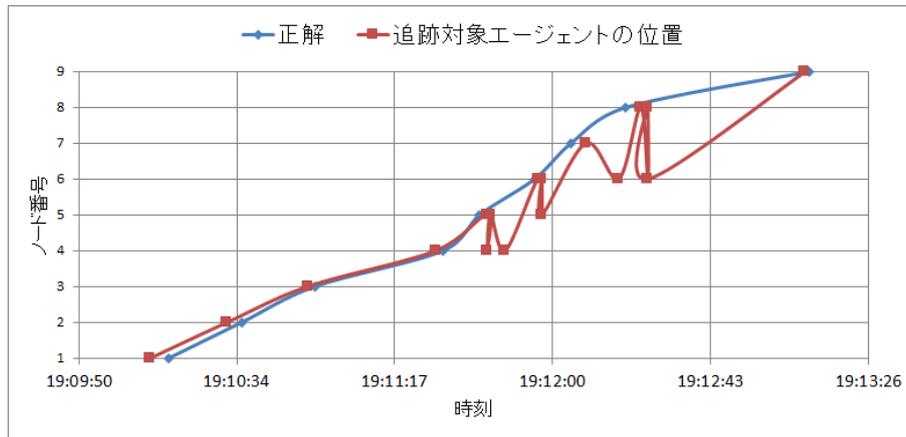


図 3.9: 人物追跡の実験結果

### 3.3.3 実験 2：普段の速度で移動する人物の追跡

2 回目の実験結果を図 3.9 に示す。人物はノードを以下の順番で移動した。

[人物が移動したノード番号]

1 ⇒ 2 ⇒ 3 ⇒ 4 ⇒ 5 ⇒ 6 ⇒ 7 ⇒ 8 ⇒ 9

2 回目では最終的には人物の位置と追跡対象エージェントの位置が一致しているが、途中で追跡対象エージェントが正解とは異なる位置を示していることが分かる。原因としては、コピーエージェントが人物を検知し追跡対象エージェントに切り替わるタイミングが起因していると考えられる。ノード間の通信は非同期でおこなっている。そのため、新しい追跡対象エージェントが起動するとき一時的に前の追跡対象エージェントが存在している瞬間があり、複数地点で追跡対象エージェントが存在する状況が発生した。ただ、最終的には人物が追跡できているため大きな問題ではないと考える。

### 3.3.4 考察

本節では提案手法の制限について述べる。追跡対象は予め決まったビーコンタグを所持する必要があるため、たとえば、追跡対象がビーコンタグを所持し忘れていることや、他人が誤って追跡対象のビーコンタグを所持している場合、追跡対象は追跡できない。このような場合に対処するには代替りのセンサとしてカメラを利用するといった、いわゆるマーカーレスの追跡を行う必要がある。ビーコンタグを利用した場合、ビーコンタグから受信した電波には固有のIDが含まれるため、人物の識別は容易である。一方、カメラを利用した場合、カメラから得られた画像で人物を識別する必要があり、識別の過程で誤検出が発生し得る。誤検出へ対処するための手法を次章で提案する。

エージェントが人物を検知するとエージェントは隣接するノードへエージェントを派遣する。人物が隣接するノードへ到着するより先に、エージェントは隣接するノードへ先回りして派遣されている必要がある。人物の到着よりエージェントの到着が遅い場合、人物が検知できず見逃しが発生する。追跡対象とする人物が少なく、隣接関係のノード数が少ないうちは問題とならないが、いずれかの数が多くなってくると、エージェントの到着が人物の到着に間に合わない場合が発生すると想定される。このような場合、人物が集中する分岐点にはノードA,Bのようにノードを複数設置し、たとえばノードAは人物1,2,3,...,10までを担当し、ノードBは人物11,12,...,20までを担当する、のような負荷の分散の工夫が必要となる。今回の評価実験では追跡対象が1人であった。1人であれば問題無く追跡できたが、人数を増やしていった場合にも問題無く追跡ができるか、また普段の移動速度以上に走って移動した場合にも問題無く追跡ができる評価が必要である。人物が走ることや、センサの死角を移動する場合、センサが人物を見逃す可能性も考えられる。このような場合に対してはセンサの死角ができないようにセンサを敷き詰める対処方法が考えられるが、センサが人物を見逃した際にも追跡を継続して行うための追跡手法<sup>2)</sup>を提案している。本手法については次章で説明する。また、本手法も組み込んだ上での実環境での評価を予定している。

ビーコンタグの電波強度によっては、最寄りのノード以外の離れたノードでも人物を検知してしまう可能性がある。実験に際しては、ビーコンタグを袋で囲み電波強度を調整した上で実施した。人物がいる最寄りのノードではなく遠方のノードでセンサデータが誤って受信された場合でも、正確に人物追跡ができる技術の

検討が必要である。たとえば、歩行速度を追跡時に用いる方法が考えられる。ある歩行速度で移動していて急に遠く離れた場所に出現するというのは考えにくく、次に人物が現れる範囲というのは物理的に移動できる範囲に限られる。歩行速度に着目し移動先の候補を絞り込むことも有効であると考えられる。

### 3.4 結言

本章ではエージェントを用いた人物追跡システムを超小型計算機である Raspberry Pi 上に実装した。評価実験によって、実装したシステムを実際のフロアに配置し普段の速度の歩行であれば人物を追跡できることを確認した。本章ではセンサとしてビーコンセンサを利用したが、実装の枠組みとしてはセンサに依存しないものであり、ビーコンセンサ以外にたとえばセンサとしてカメラも利用可能である。USB カメラのような安価な小型カメラを利用した追跡実験を今後予定している。先行研究に Raspberry Pi および USB カメラを利用した研究<sup>54)</sup>も存在しており、先行研究と提案するエージェントを用いたシステムでどのような差異があるか評価する予定である。

## 第4章 人物誤追跡への対応

### 4.1 序言

従来のエージェントによる人物追跡では、ノードに設置されているセンサにおいて正確に人物を検知し特定できる必要があった。センサで人物を検知できないことや追跡対象ではない人物を追跡対象と見なすことが発生すると、エージェントは人物の追跡に失敗していた。この問題に対してセンサの精度や人物の識別手法の改良によるアプローチが考えられる。しかし、確実に人物を検出できるセンサや、検知した人物を誤り無く識別するというのは現実的には困難である。たとえば、センサとしてRFIDセンサを使用する場合、センサの設置場所によって人物を見失う場所、すなわちRFIDタグからの信号を受信できない場所がある。その結果エージェントは追跡に失敗する。また、センサとしてカメラを用いた場合ではカメラによって撮影された画像から抽出した人物の特徴は、常に正確に抽出されとは限らず人物を一意に特定できない。たとえば、茶色の髪色の人でも光の強度によって髪の色が黒く識別されることがある。その結果、人物の識別を誤り見逃しや誤追跡を引き起こす可能性がある。

著者らはセンサにおいて人物の見失いが発生した際、隠れた隣接関係を見つける方法<sup>2)</sup>を提案した。この方法ではエージェントが追跡対象を見失うと追跡対象が見失われたノードと追跡対象が見つかったノードとの間に新しい隣接関係を構築し、エージェントがバイパスする新たなルートを構築する。この方法によりセンサが人物を見失った場合に対処することができた。RFIDのようなセンサからの信号に固有のIDが含まれ人物を一意に特定できるときは隠れ隣接関係により人物の見逃しに対応できた。

本章ではカメラを用いた場合に発生する人物の識別誤りに対応するための追跡手法を提案する。カメラのような必ずしも人物を正確に識別できないセンサを使用する場合には、追跡対象の特徴とカメラで撮影した追跡対象の画像等から抽出

した特徴との特徴間距離をもとに人物を識別する。以降では特徴間距離を単に距離と表現する。しかし、光の加減や撮影角度によって追跡対象の距離が大きく評価されることや、逆に非追跡対象の距離が小さく評価される場合が発生する。その結果、追跡対象間と非追跡対象間で距離が逆転する現象が発生する。

従来の追跡方法では人物が正確に識別できるという前提があり、人物の距離が0(追跡対象)か1(非追跡対象)かを識別することができた。しかし、識別誤りが発生する状況では距離が必ずしも0か1にはならずその間の値をとる。距離が間の値をとるとき追跡対象かそうでないかを判定する方法として、ある閾値を設けその閾値より距離が小さければ追跡対象と判断する方法が一般的である。しかし、追跡対象がある閾値より常に距離が小さくなるとは限らない。距離が逆転した場合に本来追跡すべきではない人物を追跡することや、本来追跡すべき人物を追跡しないという問題が発生する。

そこで、複数枚の画像から距離を取得し評価値を求め2つの閾値を用いて人物を追跡する手法を提案する。1つ目の閾値は追跡対象であると確定するための確定閾値、2つ目の閾値は人物の識別を先送りすることを目的とした再評価閾値である。エージェントは評価値が確定閾値と再評価閾値の間するとき人物の決定を見送る。このとき見送った人物が次に現れる可能性があるノードを予想し見送った先のノードで人物を再評価する。これにより見逃しとなっていた人物を再評価する。シミュレーション実験の結果、本手法により追跡精度が改善されることを確認した。

以降ではまず人物の未検出と誤検出について整理し本章で解決しようとしている誤検出の例を説明する。つづいて誤検出に対応した追跡手法について説明し、最後にシミュレーション実験によって提案手法の有効性を確認する。

表 4.1: センサの検出結果と現実の事象の分析結果

センサ\現実の事象	人物は存在せず	人物 A が存在	人物 B が存在	人物 A と B が存在
検出せず	正常	未検出	未検出	未検出
人物 A を検出	誤検出	正常	誤検出	正常 +未検出 <sup>a</sup>
人物 B を検出	誤検出	誤検出	正常 正常	正常 +未検出 <sup>a</sup>
人物 A と B を検出	誤検出	正常 +誤検出 <sup>b</sup>	正常 +誤検出 <sup>b</sup>	正常

<sup>a</sup> 1人の人物が未検出であり未検出の人物の追跡に失敗する。

<sup>b</sup> 追跡中の人物を正常に検出しており追跡は継続できる。

## 4.2 未検出と誤検出

利用するセンサによって人物が正確に識別されず人物の識別誤りが発生する。その結果、追跡対象ではない人物を追跡することや追跡対象である人物を追跡しないことが起こる。前者は「誤検出」を引き起こし後者は「未検出」を引き起こす。表 4.1 にセンサの検出結果と現実の事象の分析結果を示す。ここで、「正常」は追跡中の人物を正しく検出できた場合、「未検出」は追跡中の人物を検出できなかった場合、「誤検出」は追跡中の人物と異なる人物を検出した場合を示している。「未検出」と「誤検出」が含まれている場合に人物の追跡に失敗する可能性がある。たとえば、ビーコンのようなセンサとして一意に人物を識別可能な機器を用いた場合は「誤検出」は発生しない。そこで「未検出」に対応するための隠れ隣接関係を<sup>2)</sup>において提案し追跡率が向上することを確認した。

未検出と誤検出の例をそれぞれ図 4.1 および図 4.2 に示す。図 4.1 は未検出を表しており、人物がビーコンタグを所持しセンサ B の前を通過したが、センサの周囲に追跡対象以外の人物がいるためセンサ B においてビーコンタグの電波の受信できない状況が発生している。そのため、本来であればセンサ B に派遣されてい

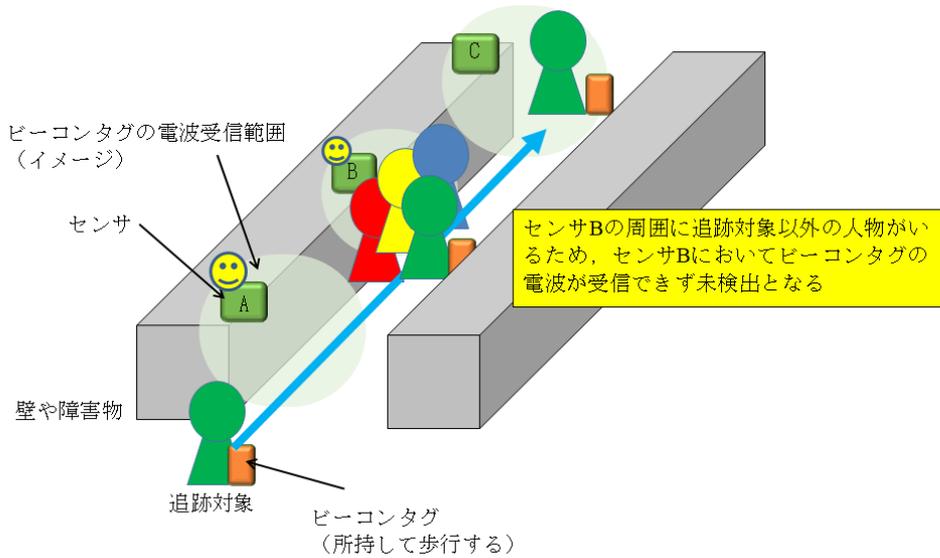


図 4.1: 未検出の例

たコピーエージェントは追跡対象を検出すると、追跡対象エージェントとなりさらにセンサCへコピーエージェントを派遣し追跡を行う。しかし、センサBにおいて追跡対象が検出できなかったため、センサC以降において追跡を継続できない。また、図4.2は誤検出を表しており、センサCにおいて追跡対象に似た人物を検出し、誤ってセンサCに検出した人物を追跡対象と判断した状況を表している。ここで具体的に似た人物を検出し誤検出が発生する例については次節において説明する。センサCのコピーエージェントは追跡対象エージェントとなりセンサC以降を追跡するが、追跡対象は実際にはセンサDへ移動している。そのためセンサD以降において追跡を継続できない。

著者らは未検出が発生した場合については隠れ隣接関係を<sup>2)</sup>用いて追跡を継続する手法を提案した。本手法について説明する。図4.3に未検出が起きた際の隠れ隣接関係を用いた追跡の流れを示す。図4.3(a)では、あるセンサにおいて追跡対象が検出され、図4.3(b)において隣接関係のノードへコピーエージェントが派遣された状態を表す。その後、図4.3(c)では追跡対象が移動した際に移動先のセンサにおいて未検出が発生し追跡対象が検出されなかったため、コピーエージェントが新たな追跡対象エージェントになれない状態となっている。その結果、図4.3(d)では追跡対象がどのエージェントにも追跡されていない状態となり人物を見逃し

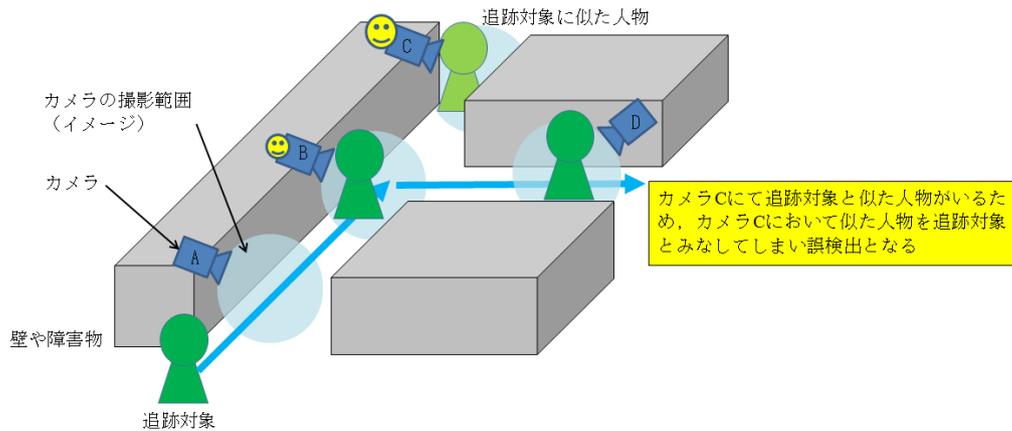


図 4.2: 誤検出の例

てしまう．そこで，隠れ隣接関係を用いた追跡では図 4.3 (d') において，どのエージェントにも追跡されていない人物が検出されたときに新たな追跡対象エージェントを生成する．追跡対象エージェントは隣接関係のノードへコピーエージェントを派遣する．ここで，新たに派遣されたコピーエージェントは同じ追跡対象を追跡している元のコピーエージェントと合流する．合流が発生すると新たな追跡対象エージェントが生成されたセンサと元の追跡対象エージェントが生成されているセンサとの間に隣接関係を構築する．このときの隣接関係を隠れ隣接関係と呼ぶ．このように未検出が発生したセンサをバイパスするルートを構築することで，未検出が発生したセンサが存在する場合であってもその先にエージェントを派遣することが可能となり追跡を継続できる．隣接関係が構築された後では，たとえばセンサの不調等によって図 4.3 (b) に示したようにセンサで人物を検出できない状態が繰り返し発生したとしても，その先にエージェントを派遣できるようになるため人物の見逃しを防ぐことができる．

未検出が頻繁に発生すると隠れ隣接関係によって隣接関係の数が増加することが懸念される．この点に関しては LRU (Least Recently Used)，あるいは LFU (Least Frequently Used) 方式を用いて隣接関係を削除する方法も検討されている．LFU 方式において隣接関係を削除した方が追跡の精度は高く維持できる結果が実験<sup>2)</sup>において得られている．

## 第4章 人物誤追跡への対応

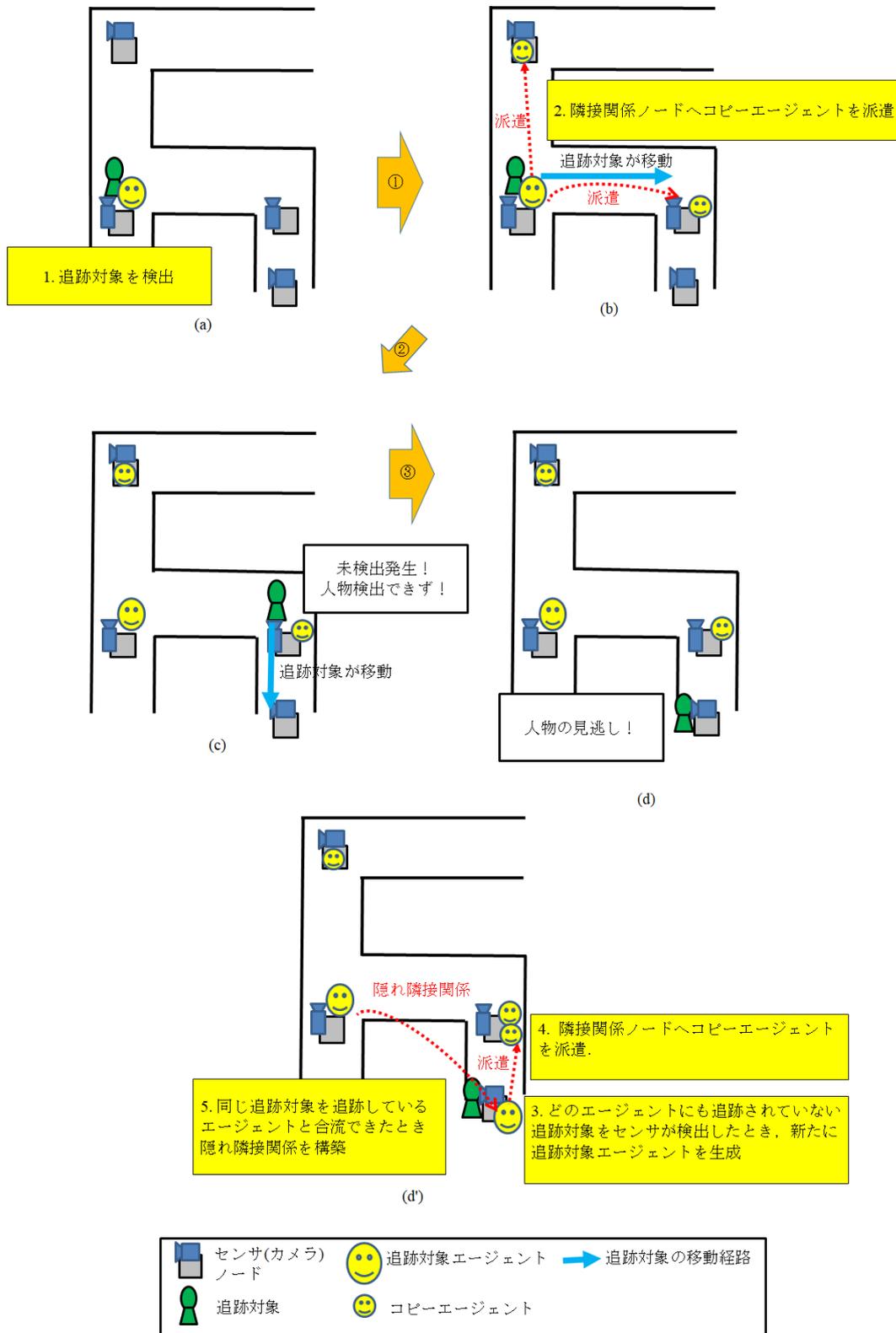


図 4.3: 隠れ隣接関係

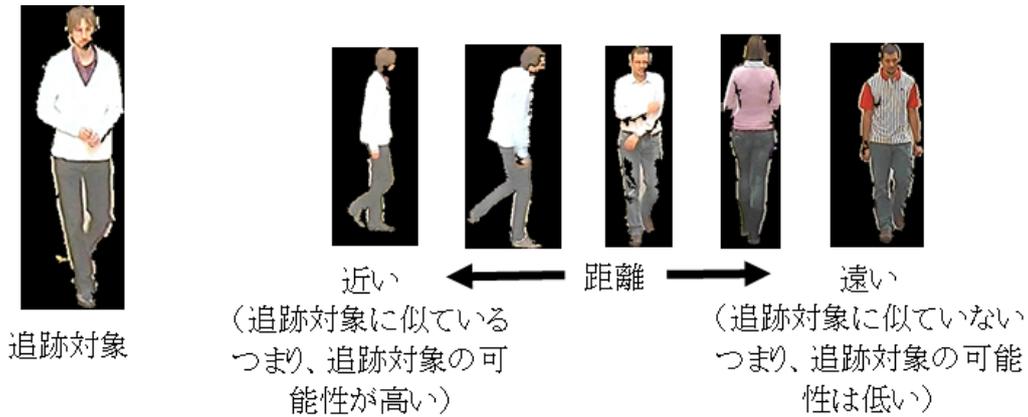


図 4.4: 距離のイメージ

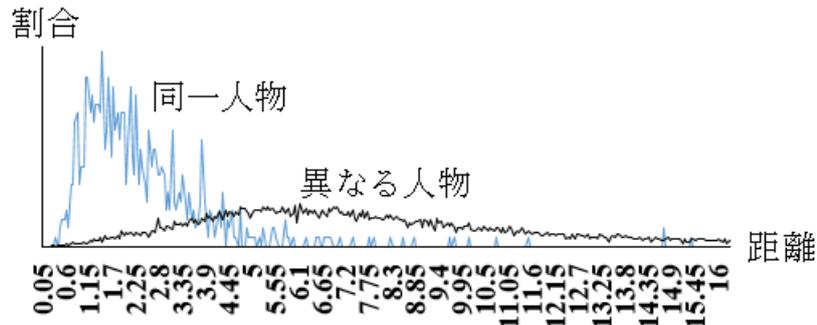


図 4.5: 同一人物間と異なる人物間の距離の分布

### 4.3 誤検出の具体例

具体的に誤検出が発生する例を西山ら<sup>55)</sup>が提案した手法を用いて説明する。西山らの実験では公開されている画像データベース SARC3D<sup>56)57)</sup>の画像を用いている。SARC3Dは50人の前方、後方、左側、右側の4方向から撮影した合計200枚の画像から構成される。これらの画像の一部の例、および距離のイメージを図4.4に示す。西山らの手法を用いると画像間の距離が求まる。距離が近いほど2枚の画像の人物は似ていることを意味し、距離が遠いほど2枚の画像の人物は似ていないことを意味する。

これらの画像に対して2枚の画像間の距離を求めた。同一人物の画像間の距離と異なる人物の画像間の距離の分布を図4.5に示す。図4.5から分かるように距離

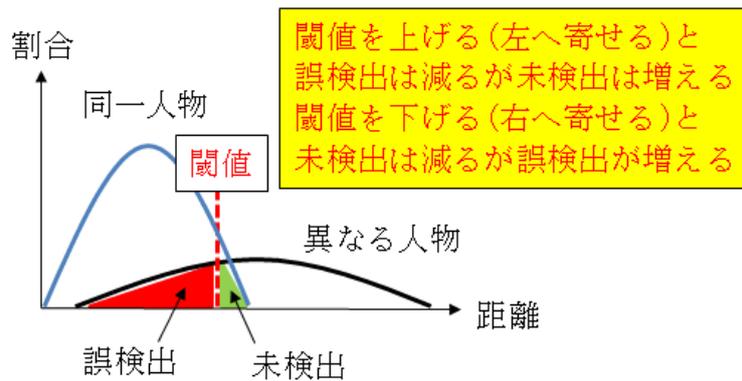


図 4.6: 誤検出と未検出

が小さいほど同一人物である割合は大きい。しかし、同一人物/異なる人物間の分布には重なりがある。このため、同一人物であると判断する閾値を低く設定すると誤検出は減るが未検出も増える。逆に閾値を高く設定すると未検出は減るが誤検出が増加する傾向があると分かる。この関係を図4.6に示す。西山ら<sup>55)</sup>以外の手法を用いた場合でも、たとえば松村ら<sup>4)</sup>の手法を用いた場合でも、単一視点の顔認証の場合で90.39%、歩容認証の場合で91.14%、これらを組み合わせたマルチモーダル特徴による個人認証で99.17%の精度を得ているが識別誤りは必ず発生する。

そこで、1つの閾値で人物を識別するのではなく、2つの閾値を導入することで識別誤りを軽減する追跡手法を提案する。提案手法の前提条件として以下2点を仮定する。

- 仮定1: 同一人物に対して距離は小さく、異なる人物に対して距離は大きくなる傾向があると仮定する。たとえば双子のように、異なる人物であっても距離がほぼ同じになるようなケースは想定しない。
- 仮定2: 同一のカメラで撮影した2枚の画像間の人物の対応付けは出来るものと仮定する。つまり、 $k$ 枚目の画像に写っている人物は $k+1$ 枚目のどの人物か対応付けはできる。ただし、その人物が誰かはわからない。イメージを図4.7に示す。画像間の人物の対応付けに関しては背景に着目した研究<sup>58)59)</sup>や抽出した人物間の色成分から判断する方法<sup>60)</sup>が提案されている。

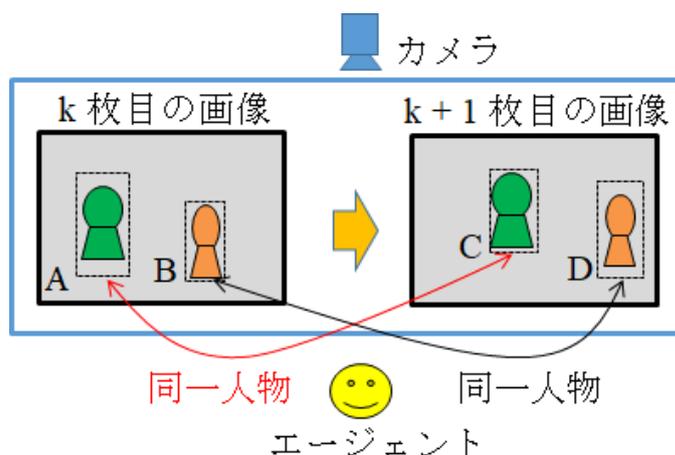


図 4.7: 画像間の人物の対応付け

## 4.4 エージェントのグループによる誤追跡への対応

### 4.4.1 提案手法の要点

提案手法では検出した人物が追跡対象であると確定する閾値と、追跡対象であると確定するには至らないが可能性がある判断するための閾値の2つの閾値を用いて追跡をおこなう。これは、図 4.4 に示したように追跡対象と非追跡対象がある程度は識別できている点に着目しており、もし一時的に追跡対象が非追跡対象として分類されたとしても、他のカメラや映り方によっては追跡対象として分類できるはずである、という考えに基づいている。そこで、一時的に認識した結果が悪くなったとしても追跡対象である可能性があれば非追跡対象と見なすのではなく継続して追跡をおこない、あるタイミングで追跡対象であると見なせたときに確定する。

また、確定はできないが追跡対象の可能性のある人物を一時的に継続して追跡する必要があるため、従来はいわゆるあるエージェントによって「点」で追跡していたのに対し、人物がいる可能性がある範囲で追跡する必要があるため、新たに「面」で追跡する。

#### 4.4.2 評価値

検知したある人物 A がどの程度で追跡対象と見なせるか評価する尺度として評価値を定義する。人物が移動する際、1つのカメラで複数枚の画像が撮影される。しかし、取得した1枚毎の画像だけで人物を判定すると偶然に距離が小さくなることや大きくなる場合に識別誤りが発生する。一方で、すべての画像を利用すると、カメラから離れた状態の悪い画像なども使用され、その画像の影響を受け追跡対象と評価されない可能性が高まる。これらの影響を抑えるために、複数回の距離の移動平均を評価値として定義する。

$$v_A(n) = \frac{1}{n} \sum_{i=1}^n P_A(x_i) \quad (4.1)$$

$n$  は評価値を得るために必要な画像枚数で、たとえば  $n = 5$  の場合、最低5回の画像を取得し、6回目を取得したとき最初の1回目の画像は捨てて平均を求める。 $x_i$  はセンサから  $i$  回目に取得した距離で、 $P_A$  は取得した距離を適合率 (0.0~1.0) に置き換える関数である。センサから  $n$  回取得した距離を確率に置き換え移動平均をとった値を人物の評価値とする。評価値を求めるにあたっては最低  $n$  枚を必要とするため、必要枚数を取得できなかった場合の評価値は0となる。追跡対象の評価値が0となった場合、人物が検知されず未検出となるが<sup>2)</sup>で対応済みであり同等の手法で解決できる。

#### 4.4.3 2つの閾値

提案手法では2つの閾値を定義し人物の識別誤りをコントロールする。このイメージを図4.8に示す。提案手法では人物を追跡対象と確定するための確定閾値と、確定を見送り別のタイミングで再評価するための再評価閾値を導入する。2つの閾値を設けることで、以下の1~3のケースが発生する。

- ケース1: 評価値が確定閾値より大きい場合、検知した人物を追跡対象であると確定する。確定閾値を高く設定することで誤検出を減らすことができる。
- ケース2: 評価値が確定閾値と再評価閾値の間である場合、人物の確定を見送る。別カメラや別の状況ではケース1もしくはケース3となり確定できる可能性がある。

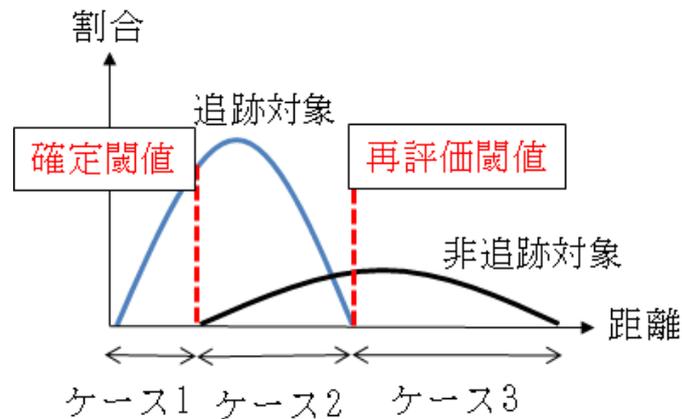


図 4.8: 確定閾値と再評価閾値

- ケース3: 評価値が再評価閾値より小さい場合、検知した人物を追跡対象ではないと確定する。再評価閾値を低く設定することで見逃しを減らすことができる。

ケース1では従来の追跡手法と同様に検知した人物を追跡対象と見なし追跡する。ケース3では従来の追跡手法と同様に検知した人物が非追跡対象であると思なし人物を追跡しない。閾値を1つしか設けない場合では、ケース2が発生しないためケース1もしくはケース3のいずれかに該当することとなり、識別誤りに繋がる。閾値を2つ定義しケース2に対応することで識別誤りの問題を軽減する。

#### 4.4.4 グループ

同一の人物を追跡しているエージェントの集合をグループと定義する。グループは「追跡対象エージェント」と「コピーエージェント」で構成される。監視員はグループをチェックし追跡対象のおおよその位置を知ることができる。1つのグループで1人の追跡対象を追跡する。2人の追跡対象が存在する場合、2つの異なるグループが存在する。グループ間に重複がある場合、グループが重複するノード上には異なる人物を追跡する複数のエージェントが存在する。図4.9にグループの例を示す。グループ1が人物P1を、グループ2が人物P2を、そしてグループ3が人物P3を追跡している。図4.9の点線箇所においてグループ1およびグループ2はオーバーラップしており、それぞれのグループのエージェントが人物を検

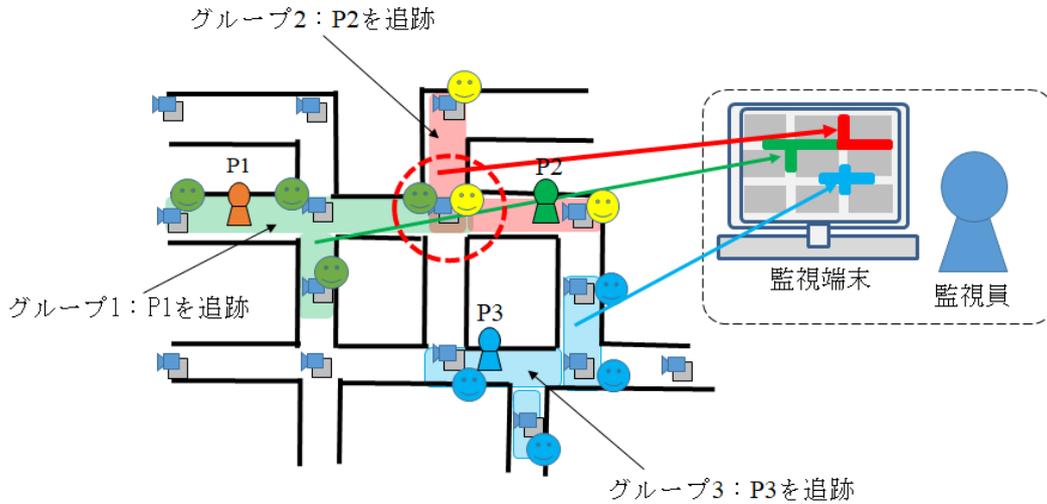


図 4.9: グループの例

出し追跡対象かどうか判定をおこない追跡する。

#### 4.4.5 グループ拡張

ケース2の対応として、人物の確定を見送り再評価を可能とするためグループを拡張する。ケース2で人物の確定/未確定を判定すると追跡対象と非追跡対象を区別する距離には重なる部分があり誤検出や未検出に繋がる。そのため、ケース2では人物を確定せず人物の確定を見送る。このとき、単純に見送るだけでは人物はエージェントが配置されていないノード、すなわちグループの外に移動する可能性がある。エージェントが配置されていないノードに人物が移動すると、エージェントは人物を検知できないため人物の追跡に失敗する。そこで、人物の確定を見送ったエージェントはグループ拡張をおこなう。グループ拡張では人物の確定を見送ったエージェントがいるノードの隣接関係にあるノードにコピーエージェントを配置しグループを拡張する。グループ拡張により人物の移動先がグループ内に含まれることになり人物の追跡を継続できる。

グループ拡張の例を図4.10に示す。ここで追跡対象がノード1におりコピーエージェントがノード2,3,4に存在するとする。ノード2のエージェントは追跡候補を検出しケース2として追跡対象を識別したとする。すると、ノード2のエージェ

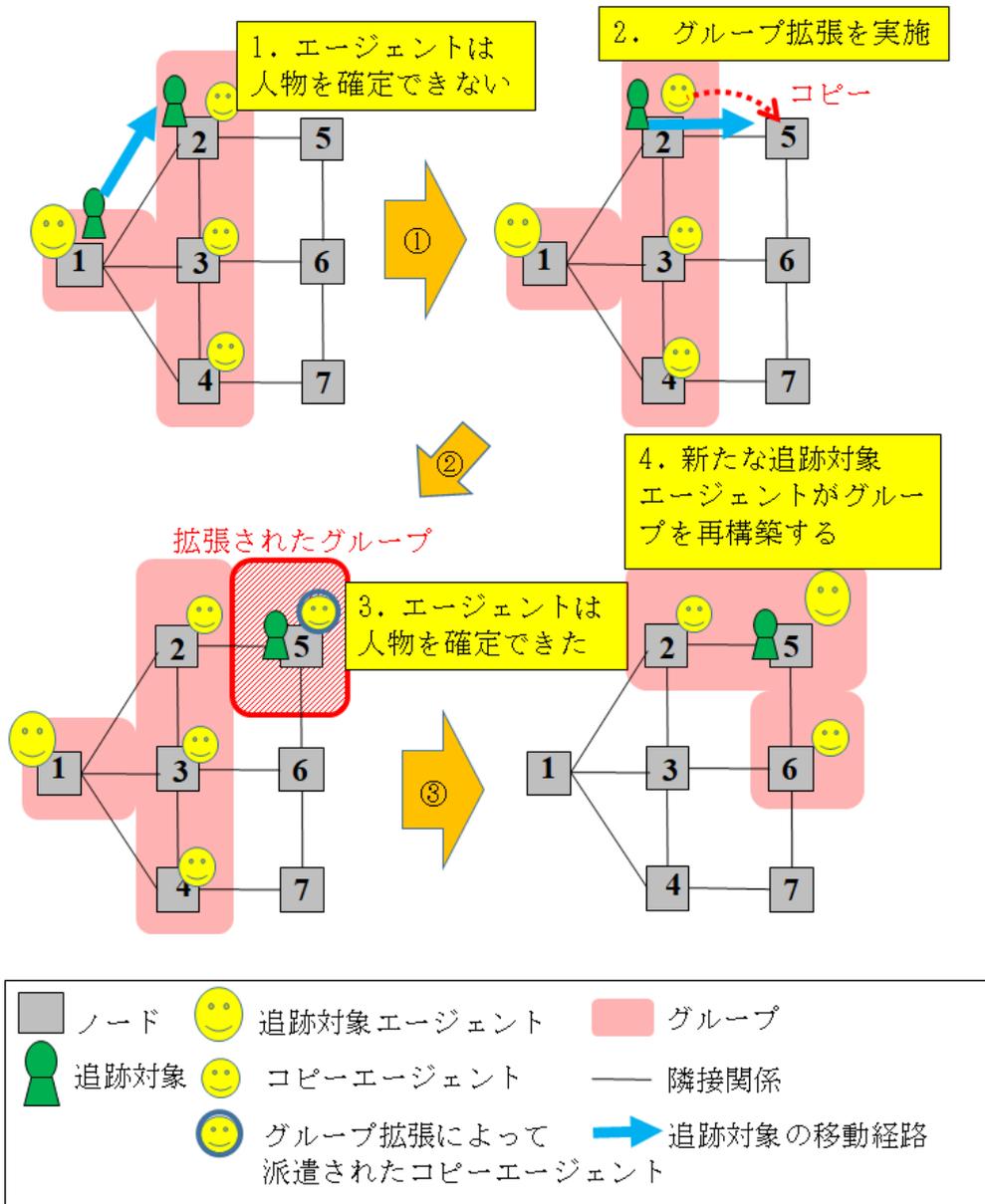


図 4.10: グループ拡張の例

ントは隣接関係にあるノード5をカバーするようにグループを拡張する。つづいて、追跡対象がノード5へ移動したとする。ノード5のエージェントは追跡候補を検出しケース1として識別したとする。すると、ノード5のエージェントは新しい追跡対象エージェントとなり、他の追跡対象エージェントおよびコピーエージェントは消去される。新しい追跡対象エージェントはコピーエージェントを隣接関係のノード2,6へ派遣し新たなグループを再構築する。以上のような処理を繰り返すことでエージェントは追跡対象を追跡する。

### 4.5 グループによる追跡の流れ

エージェントのグループを用いた場合の人物追跡の流れを図4.11に示す。図4.11では従来の人物追跡の流れから変わる部分を赤色に示している。監視員は人物の追跡を始めるとき、その人物の特徴量を有するエージェントをその人物がいるノードへ派遣する。追跡対象エージェントは、コピーエージェントを自身の隣接ノードに派遣する。コピーエージェントはセンサから情報を収集し人物の検出を試みる。人物を検出するとエージェントは評価値を計算する。評価値が確定閾値を超えた場合、追跡対象を検出したと判断し、人物の検出を追跡対象エージェントへ通知する。追跡対象エージェントは人物の通知を全コピーエージェントへ通知する。追跡対象を検出したコピーエージェントが新しい追跡対象エージェントになる。その後、元の追跡対象エージェントとコピーエージェントを消去する。新しい追跡対象エージェントは、自身のコピーエージェントをその隣接ノードに派遣する。また、評価値が確定閾値と再評価閾値の間の場合、グループ拡張をおこない人物を検出したエージェントは自身の隣接関係のノードへさらにコピーエージェントを派遣する。これらのステップを繰り返すことで人物の追跡を実現する。

## 第4章 人物誤追跡への対応

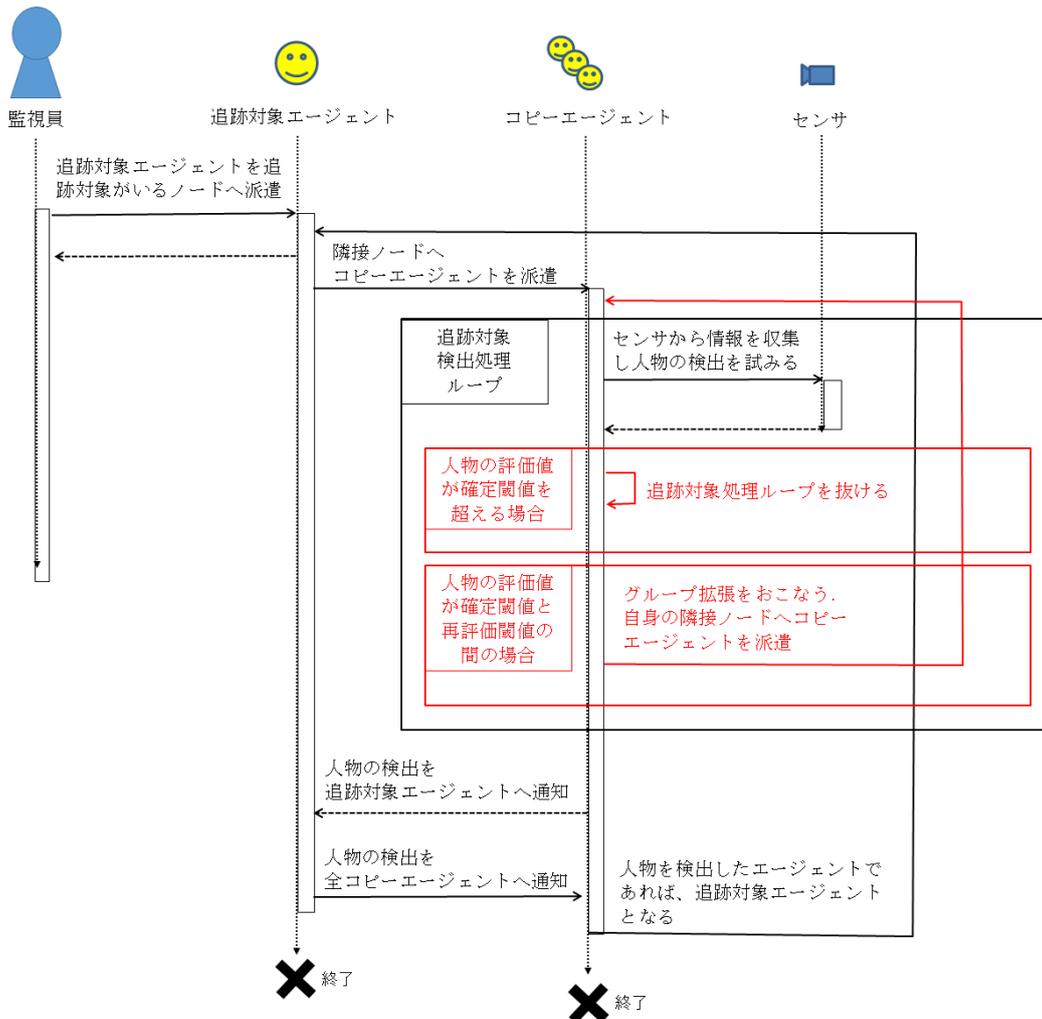


図 4.11: グループによる追跡の流れ

監視員は追跡対象エージェントの位置あるいはコピーエージェントの位置を確認することで、人物のおおよその位置を知ることができる。グループ拡張がおこなわれることで、コピーエージェントがいる位置にも人物がいる可能性が発生するため、人物がいる位置は追跡対象エージェントとコピーエージェントがいるノードのどこかということになる。従来の追跡では、いわゆるエージェントによって「点」で追跡していたのに対し、提案手法では人物がいる可能性がある範囲を新たにグループという「面」で追跡することになる。

### 4.6 シミュレーション実験

提案した追跡手法の効果を確認するためのシミュレーション実験を行った。

#### 4.6.1 実験環境

##### マップ

実験で使用するフロアのマップを図 4.12 に示す。マップは実際に存在するフロアを模擬したものである。白色四角が人物が移動する通路を表しており全部で 162 個あり、それぞれが 1.5m × 1.5m の大きさである。緑色丸がカメラを表しており全部で 18 台をフロアに配置している。カメラが配置されている白色四角内に人物が訪れたとき、その人物は撮影範囲内にいるとみなす。

##### 人物の移動

人物は開始点からランダムに選択した 2 地点である A 点、および B 点を經由し終了点まで移動する。開始点から A 点まで、A 点から B 点までおよび B 点から終了点までは最短経路で移動する。開始点と終了点はセンサが設置されている点からランダムに選択する。人物は 1.5 m/s の速度で移動する。つまり約 1 秒間で 1 点間を移動する。エージェントがカメラから距離を取得する頻度は 150ms 間隔とする。

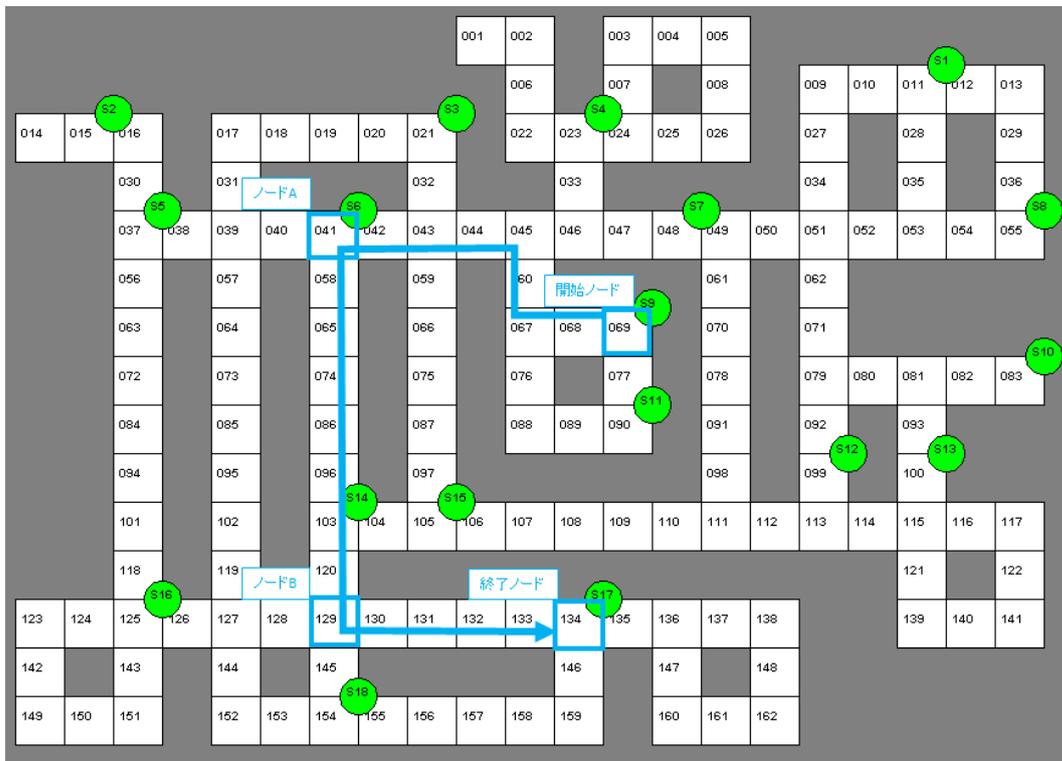


図 4.12: シミュレーションマップ

### 識別誤りの再現方法

エージェントは以下の状況1, 2に応じてシミュレータから距離を取得する。状況1, 2に応じて距離を取得することで、取得した距離が追跡対象と非追跡対象で逆転するケースが起き、識別誤りが発生する状況を再現できる。

- 状況1：追跡対象が撮影範囲内に存在する場合、図4.5に示した同一人物の分布の中からランダムに選択した距離を取得する。
- 状況2：非追跡対象が撮影範囲内に存在する場合、図4.5に示した異なる人物の分布の中からランダムに選択した距離を取得する。

### 追跡の精度

人物が移動を開始し移動を終了するまでの間の追跡を1回の追跡と数える。人物が開始点から移動し終了点に到着したとき、最後までエージェントが自身の追跡対象を追跡できていた場合にのみ追跡に成功したと判断する。もし追跡の途中で誤って1度でも非追跡対象を追跡してしまった場合、あるいは追跡対象がグループ外に出てしまった場合は、その場で追跡失敗と見なす。追跡成功率を以下として定義する。

$$\text{追跡成功率} = \frac{\text{最後まで追跡に成功した数}}{\text{追跡数}} \quad (4.2)$$

たとえば、図4.12に示したように追跡対象が移動する場合では、追跡対象は8台のカメラにおいて追跡対象をグループ外に逃さず、かつグループ内の非追跡対象を誤って追跡しなかった場合にのみ成功となる。追跡失敗は、追跡対象を非追跡対象と判断し未検出となった場合、もしくは誤検出によって非追跡対象を追跡対象と判断し別の人物を追跡した場合に発生する。そこで、追跡の失敗を分析するために見逃し率と誤追跡率を以下として定義する。

$$\text{見逃し率} = \frac{\text{追跡対象を見逃した数}}{\text{追跡数}} \quad (4.3)$$

$$\text{誤追跡率} = \frac{\text{非追跡対象を追跡した数}}{\text{追跡数}} \quad (4.4)$$

## 4.6.2 実験1：評価値の効果の確認

### 追跡成功率の評価

評価値の効果を確認するために、移動平均を使用せず1回の距離のみを利用した  $n = 1$  のときと、 $n = 5$  とし移動平均を用いたときで、確定閾値を 0.90, 0.80, 0.70 と変化させた場合の追跡成功率を求めた。同時移動人数を 1~8 人とし 100 回ずつ実施した。追跡成功率を図 4.13 上段に示す。

図 4.13 上段における (1, 0.9, -) はそれぞれ  $(n, d, r) =$  (評価値計算のための画像使用枚数, 確定閾値, 再評価閾値) を意味している。  $r = -$  の場合は、再評価閾値を設定しないためグループ拡張は行わないことを意味している。 $n = 1$  の場合では、同時移動人数が増えると追跡成功率が大幅に低下していることが分かる。一方で  $n = 5$  の方では同時移動人数を増やした場合でも、ある程度の追跡成功率を保っていることが分かる。

### 見逃し率と誤追跡率の評価

確定閾値を上げると非追跡対象が選択される可能性が減るため誤追跡率は低下するが見逃し率は増加する。このことを確認するために、 $n = 5$  の場合の見逃し率と誤追跡率を図 4.13 中段および図 4.13 下段に示す。実験結果を見ると想定した通りの結果が得られていることが分かる。

見逃し率は確定閾値を 0.90 とした場合は 0.7 程度であったものが、確定閾値を下げていくにつれ下がっていき、確定閾値が 0.70 の場合でほぼ 0.0 となった。また同時移動人数が増えた場合であっても傾向は一定であることが分かる。一方で誤追跡率は確定閾値を下げていくと増加していることが分かる。これは非追跡対象の評価値が偶然に確定閾値を超える可能性が高まるためである。また同時移動人数に比例して誤追跡率が増加していることが分かる。同時移動人数が増えることにより、確率的にグループ内に存在する非追跡対象者の人数も増える。それぞれの非追跡対象者について追跡対象かどうかの判定が行われ、その中の一人でも確定閾値を超えると誤追跡となる。この結果同時移動人数と比例して誤追跡率が上がっているものと想定される。

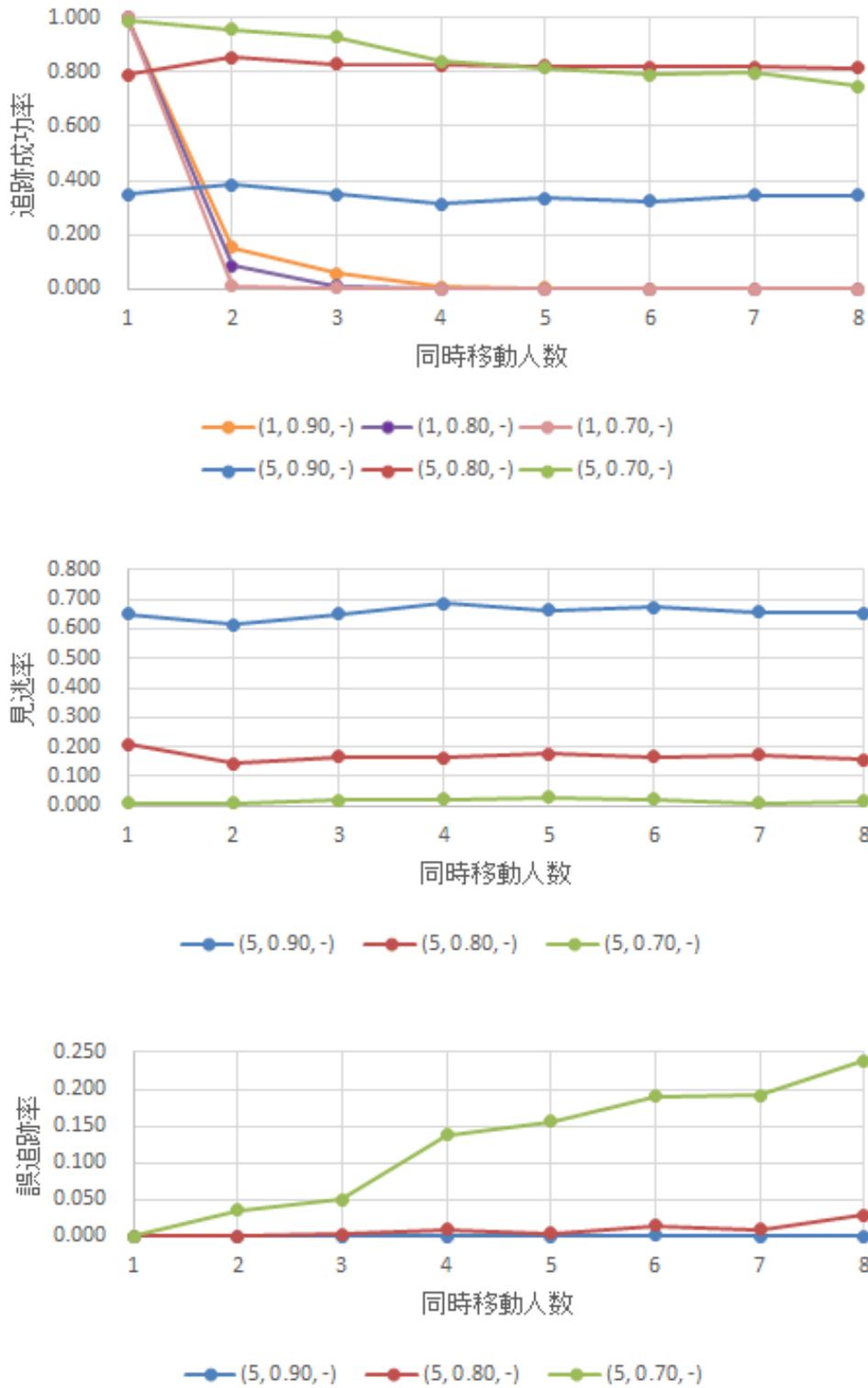


図 4.13: 評価値の効果の確認

### 4.6.3 実験2：グループ拡張の効果の確認

#### 追跡成功率の評価

グループ拡張の効果を確認するために、 $n = 5$ 、確定閾値を0.80、再評価閾値を0.65としグループ拡張なしの場合とグループ拡張ありの場合の追跡成功率を求めた。同時移動人数を1～8人とし100回ずつ実施した。追跡成功率を図4.14上段に示す。

グループ拡張の仕組みを組み込むことで追跡成功率が改善されており、グループ拡張なしの場合に0.85～0.80であったものがグループ拡張ありでは1.00～0.95となった。また、グループ拡張あり、なし、のどちらの結果も同時移動人数の多寡に関わらず、一定の追跡成功率が保たれていることがわかる。

#### 見逃し率と誤追跡率の評価

$n = 5$ における見逃し率と誤追跡率の結果を図4.14中段および図4.14下段に示す。

見逃し率については、グループ拡張なしの時に0.15程度であったものが、グループ拡張ありの時にほぼ0.00となった。これはグループ拡張によりエージェントが人物を再評価する機会が増えたことで見逃しが減ったためである。また、同時移動人数を増やしても見逃し率の傾向に変化がないことがわかる。

一方、誤追跡率についてはグループ拡張ありの方が大きく、また同時移動人数が増えた場合に増加していることが分かる。前者はグループが拡張することでグループ内に存在する非追跡対象者が増加したためであり、後者は同時移動人数が増えることにより確率的にグループ内に存在する非追跡対象者の人数が増えたためだと想定される。

グループ拡張により、見逃し率は低下するが誤追跡率は増加する。しかし、見逃し率が低下する効果が誤追跡率が増加する効果よりも高いため、追跡成功率が向上していることがわかる。

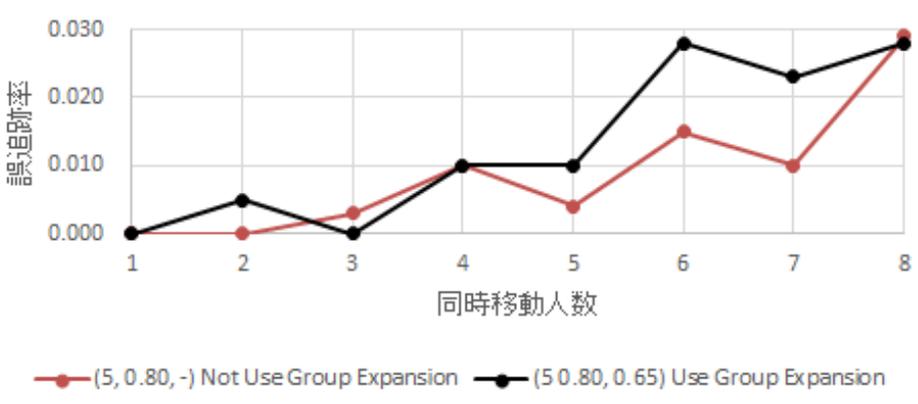
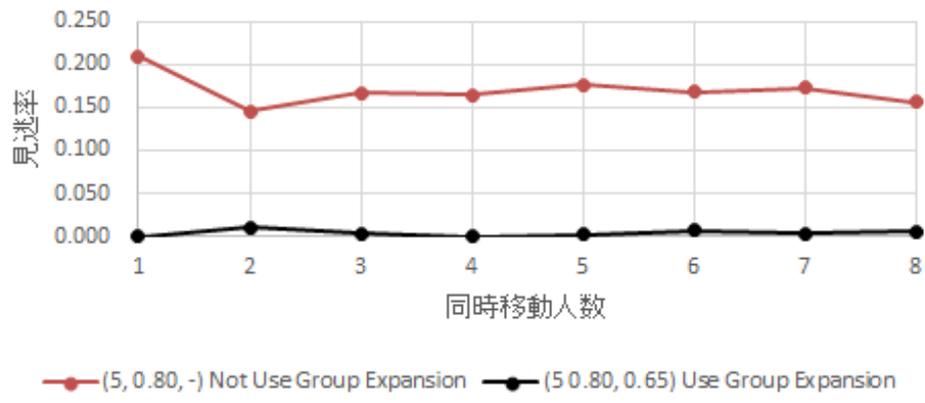
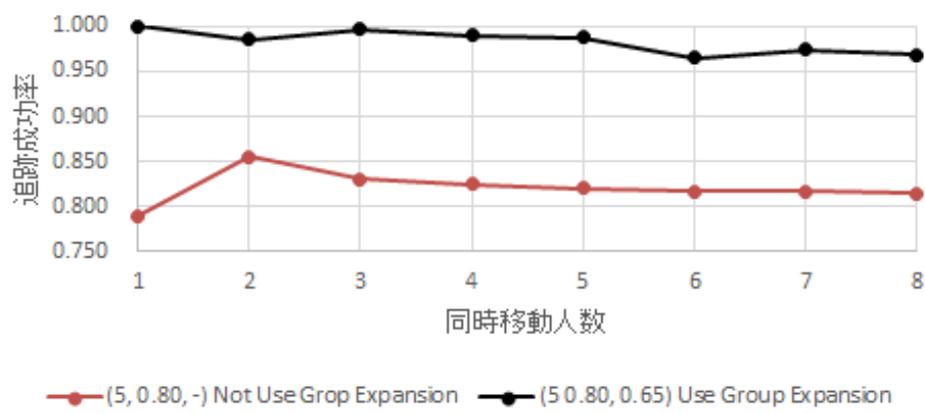


図 4.14: グループ拡張の効果の確認

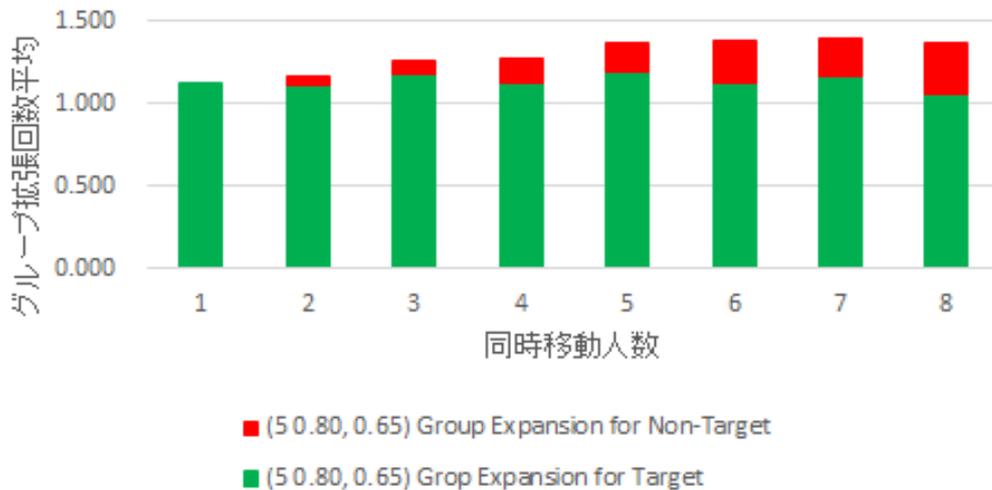


図 4.15: 平均グループ拡張回数

#### グループ拡張回数の評価

1回の追跡において何回のグループ拡張が行われたのか分析するためにグループ拡張回数の平均を求めた。平均グループ拡張回数は1回の追跡の開始から終了までの間に何回のグループ拡張が行われたかの平均値である。

図 4.15 に (5, 0.80, 0.65) のパターンにおける、グループ拡張回数の平均を示す。図 4.15 を見ると平均的に1回の追跡で1回強程度のグループ拡張が発生していることがわかる。グループ拡張は追跡対象に対して行われるケースと、非追跡対象に対して行われるケースがある。追跡対象に対するグループ拡張は、追跡対象が一人であるため、同時移動人数が増えても変化しないと想定される。

図 4.15 を見ても同時移動人数に関わらず一定のグループ拡張回数で推移していることがわかる。一方、非追跡対象に対するグループ拡張は、同時移動人数に比例して増加していることがわかる。これは誤追跡率の増加と同様に、同時移動人数が増えることにより確率的にグループ内に存在する非追跡対象者の人数が増えたためだと想定される。

#### 4.6.4 閾値設定のガイドライン

確定閾値および再評価閾値をどう設定すべきか確認するため、これらの閾値を様々に変化させたときの追跡成功率を確認した。n = 5 における確定閾値および再評価閾値を変化させた場合の追跡成功率を求めた。同時移動人数を1~8人とし100回ずつ実施した。確定閾値を0.85, 0.80, 0.75, 再評価閾値を0.70, 0.65, 0.60とした9パターンの実験結果を図4.16に示す。

実験結果はd = 0.85 およびr = 0.60 として設定した場合の追跡成功率が最も高かった。これは9パターンにおいてdを最も高く、かつrを最も低く設定したパターンである。確定閾値を変化させた場合の追跡成功率の傾向としては、d = 0.85の追跡成功率がd = 0.80よりも高いことを示しており、d = 0.80の追跡成功率がd = 0.75よりも高いことを示している。再評価閾値を変化させた場合の追跡成功率の傾向としては、r = 0.60の追跡成功率がr = 0.65よりも高いことを示しており、r = 0.65の追跡成功率がr = 0.70よりも高いことを示している。

シミュレーション結果に基づいてパラメータを設定するためのガイドラインを示す。まず、カメラの検出範囲や人の歩行速度などの環境に応じて、nを調整する必要がある。nを大きくすることで追跡成功率が向上するため、nはできるだけ大きく設定すると同時に、1台のカメラで評価値を計算するために必要な画像数は取得できる必要がある。提案手法ではカメラごとに異なる値にnを設定できるため、カメラの種類や設置環境によってnを調整する。フロアの要所毎に性能の異なるカメラが設置されているようなケースでは、たとえば性能の良い拠点のカメラはnを大きく取り、また狭い範囲を監視しているようなカメラではnを小さく取る等の調整をおこなう。

実験結果によると確定閾値の影響は、再評価閾値への影響よりも大きいことが分かった。この結果は、他の人物の認識アルゴリズムを使用した場合も同様の結果になると想定される。他の人物の認識アルゴリズムを用いた場合でも、図4.5のようなグラフになると考えるためである。また、提案手法の主な考え方は、確定するための閾値を高くすると誤検出の可能性は減少するが、未検出の可能性が増えるという点に着目している。そこで、未検出の可能性を減らすために、グループ拡張の再評価閾値を導入した。したがって、誤検出が許容されるように確定閾値を調整した後、再評価閾値を段階的に下げていくように調整することで期待する追跡精度を得られると考える。

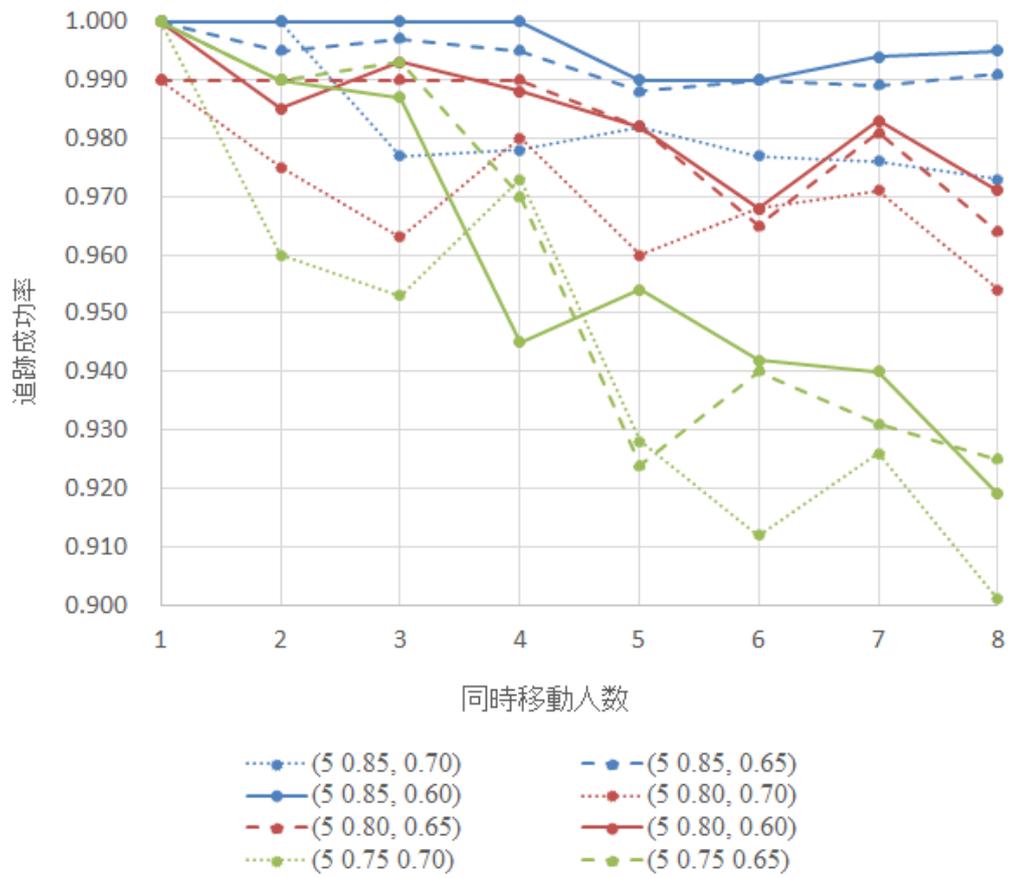


図 4.16: 閾値を変化させた場合の追跡成功率

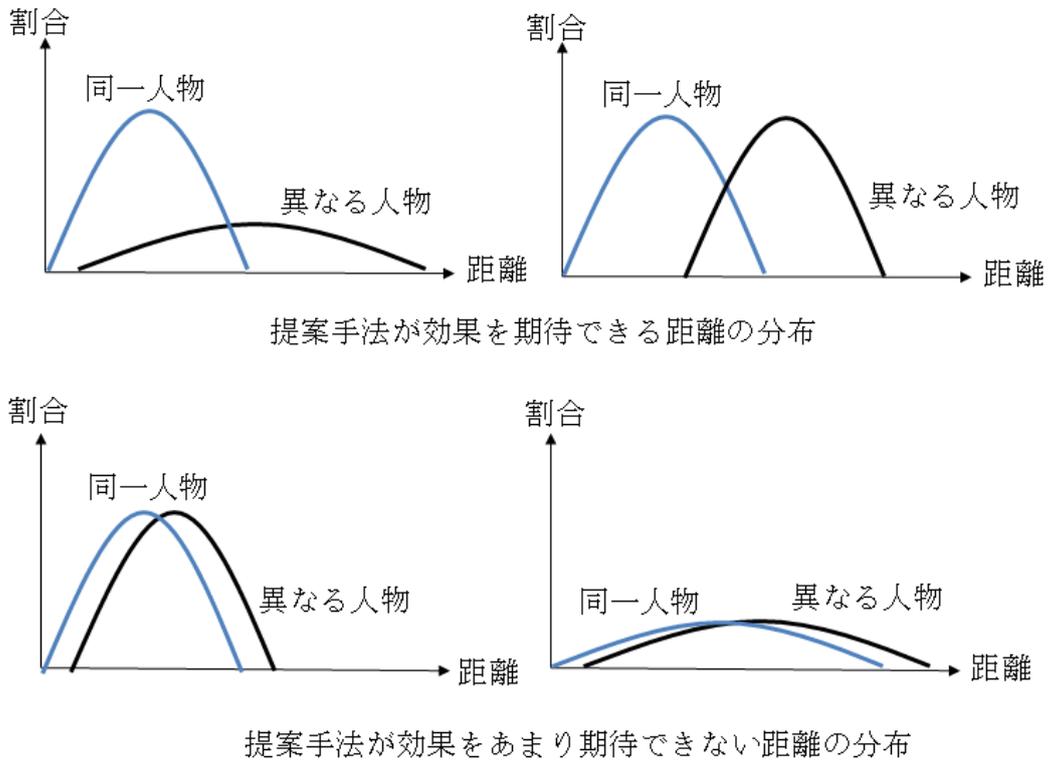


図 4.17: 距離の分布

#### 4.6.5 考察

本節では提案手法の制限について述べる。SARC3Dは人物の追跡評価でよく使用されるデータセットであるが、実際に人物が移動している最中にカメラで撮影した写真とは異なる点がある。実際のカメラによる追跡においては、カメラから撮影した画像が検出範囲外にあり人物が見切れることや光の強度が原因で画像がぼやける可能性がある。このような不正な画像はSARC3Dには含まれない。これらの状況はシミュレーションでは考慮されない。このような場合、評価値が正しく計算できず追跡対象を見失う要因となる。したがって、評価値を計算する前にこのような不正な画像を除去する手法を適用する必要がある。

4.3節の提案手法の前提条件で説明したように、提案手法では追跡対象が認識アルゴリズムによってある程度正しく識別できる必要がある。つまり、図4.5に示したように、西山らの手法等を使用して追跡対象と非追跡対象を分類できる必要がある。もし、双子のような似た人物がいる場合、追跡対象と非追跡対象の評価値

は同様になり誤検出が増加すると想定される。たとえば、追跡対象と非追跡対象を分類するとき図4.17に示すように、追跡対象と非追跡対象の距離の分布に重なりが大きい場合、確定閾値や再評価閾値を適切に設定することができないため提案手法の効果はあまり期待できないと考える。しかし、単一の閾値で追跡対象を追跡する場合、閾値が高いと未検出が増加し閾値が低いと誤検出が増加する。提案手法では、2つの閾値、すなわち確定閾値と再評価閾値を導入した。図4.16に示すように、確定閾値を高く設定すると誤検出は減少し再評価閾値を低く設定すると未検出が減少する。したがって、2つの閾値により類似した人物がいる場合であってもある程度は追跡精度の向上は見込める。今後の取り組みとして、他グループ間におけるエージェント間の連携により閾値を調整する機能を検討している。類似した人物が各エージェントによって追跡されている場合、エージェントは追跡している人物の特徴を交換し閾値の調整をおこなう。たとえば、似た人物を異なるエージェントが追跡している状況であれば、確定閾値を高く調整し識別誤りの軽減をおこなう。これにより類似した人物の追跡精度の向上をおこなう。

提案手法では検出した人物が追跡対象の可能性がある場合に、一時的に追跡範囲を拡張することにより人物の見逃しを軽減する。しかし、人物の確定がなされず追跡範囲が拡張され続ける可能性がある。そのため計算機の計算資源の利用の増大が懸念される。拡張を抑制する仕組みが必須である。この点に関しては次章で提案する監視範囲のどこに人物がいる可能性が高いか確率を求め、人物がいる可能性が低い範囲は監視対象から外すことで解消する。

## 4.7 結言

本章ではエージェントを用いた人物追跡システムにおいて発生する人物の識別誤りに対応した追跡手法を提案した。提案手法では人物を追跡するために確定閾値と再評価閾値の2つの閾値を設け、確定に至らないが確定の見込みがある人物を検出した際には別の機会に再評価できる仕組みを導入した。シミュレーション実験の結果、本手法により追跡精度が改善されることを確認した。シミュレータでは公開されている画像のデータセットをもとに人物から得られる距離を生成しており、実際に人物が移動する場合とは差異が発生すると想定される。今後実フロアで提案手法の妥当性を評価することを計画している。

提案手法を Raspberry Pi のような低スペックの超小型計算機へ搭載する際には、人物の識別をリアルタイムにかつ少ない計算量でこなす必要がある。関連研究として山内ら<sup>61)</sup>が整理しているように、Haar-like 特徴量<sup>62)</sup>や HOG (Histograms of Oriented Gradients) 特徴量<sup>63)</sup>に基づくアルゴリズムを利用することで、カメラを用いた人物追跡システムの構築が実現できるのではと考える。

提案手法は人物の識別誤りが発生する場合の例としてカメラから取得した画像を利用した場合を例に用いた。提案手法は画像に限らず追跡対象と非追跡対象が距離としてお互いがある程度分類できる場合であれば用いることができる。したがって、画像以外にも音声や歩行時の振動<sup>64)</sup>による個人の認識も研究がされており、提案手法が適用できるものと考えられる。

## 第5章 人物存在確率の計算手法

### 5.1 序言

著者らが提案するエージェントによる追跡システムは、カメラで撮影された画像から追跡対象の特徴を抽出する。しかし、追跡対象の特徴は必ずしも正確に抽出されるとは限らない。たとえば、カメラが茶色の髪色の追跡対象を追跡する場合、光の加減によっては、その髪色は黒色として認識されることがある。そのため、追跡対象ではない人物を追跡対象として認識することや、逆に追跡対象を追跡対象ではないと認識することがある。このような場合に対処するために、2つの閾値とエージェントのグループによる追跡を導入した<sup>3)</sup>。

しかし、グループによる追跡では、追跡対象がグループ内のどこにいるかを考慮していなかった。また、グループの拡張を制御する仕組みは無かったため、グループ内のエージェントの数が増大するという懸念があった。

そこで、本章ではエージェントのグループによる追跡効率の改善をおこなう。追跡効率の改善のために、まずグループ内における追跡対象の存在確率の計算手法を提案する。存在確率の計算により、グループ内のどのノードに追跡対象が存在する可能性が高いかを知ることができる。追跡対象が存在する可能性が低いノードをグループによる監視対象から外すことにより、グループの拡張を抑えつつ、効率的に追跡対象を追跡することができる。シミュレーション実験により、従来のグループによる追跡と同等の精度を保ちつつ、かつエージェント数を抑えた追跡ができることを確認した。

### 5.2 グループを用いた追跡および問題点

グループによる追跡では追跡対象がグループの外に出てしまうとその追跡対象を追跡できない。そこで2つの閾値を用いてグループを拡張する手法を提案した。

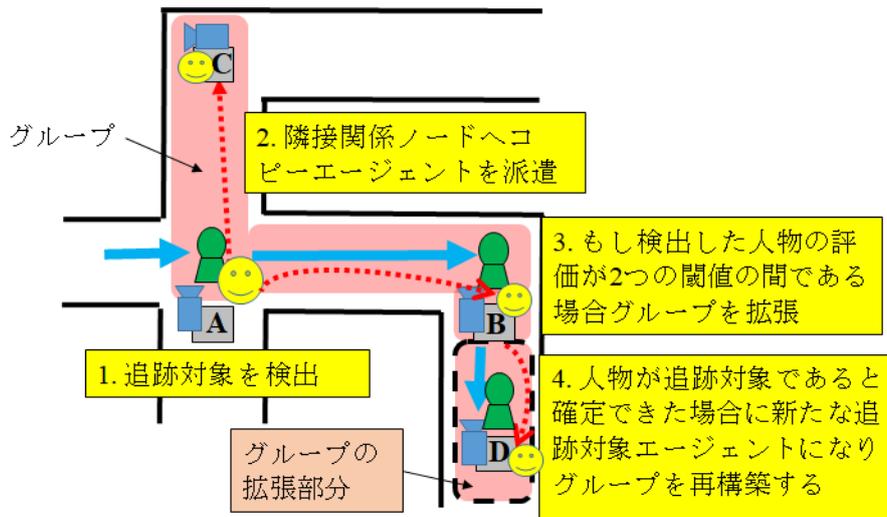


図 5.1: グループ拡張

2つの閾値は (1) 人物を追跡対象であると確定するための確定閾値、および (2) 確定を延期するための再評価閾値と定義した。ある人物の評価がこの2つの閾値の間であれば、人物の確定は延期されグループが拡張される。エージェントはコピーエージェントをその隣接関係ノードへ派遣し、その人物が次に現れるノードがグループ内に含まれるようにグループを拡張する。グループ拡張により人物が移動する可能性のあるノードがグループに含まれることになり人物の追跡を継続できる。確定閾値を高く設定することで誤追跡を抑え、再評価閾値を低く設定することで見逃しを抑えることができる。

グループ拡張による追跡の例を図 5.1 に示す。図 5.1 の例ではノード B に人物が現れて評価の結果2つの閾値の間になったときを表す。すなわち、現れた人物が追跡対象である可能性はあるが追跡対象であると確定できない状態を表している。検出した人物が追跡対象の可能性があるため、もし人物がノード D へ移動した場合に見逃してしまう。そこで、ノード B の隣接関係ノードであるノード D へコピーエージェントを派遣しグループを拡張する。拡張されたノード D にコピーエージェントが存在するため、ノード D へ人物が移動したときにその人物を再評価することが可能となり人物を継続的に追跡することができる。

しかし、人物の評価結果が二つの閾値の間にある状態が続いた場合にグループが拡張し続ける。その結果、グループ内のどこに追跡対象がいるかを把握するこ

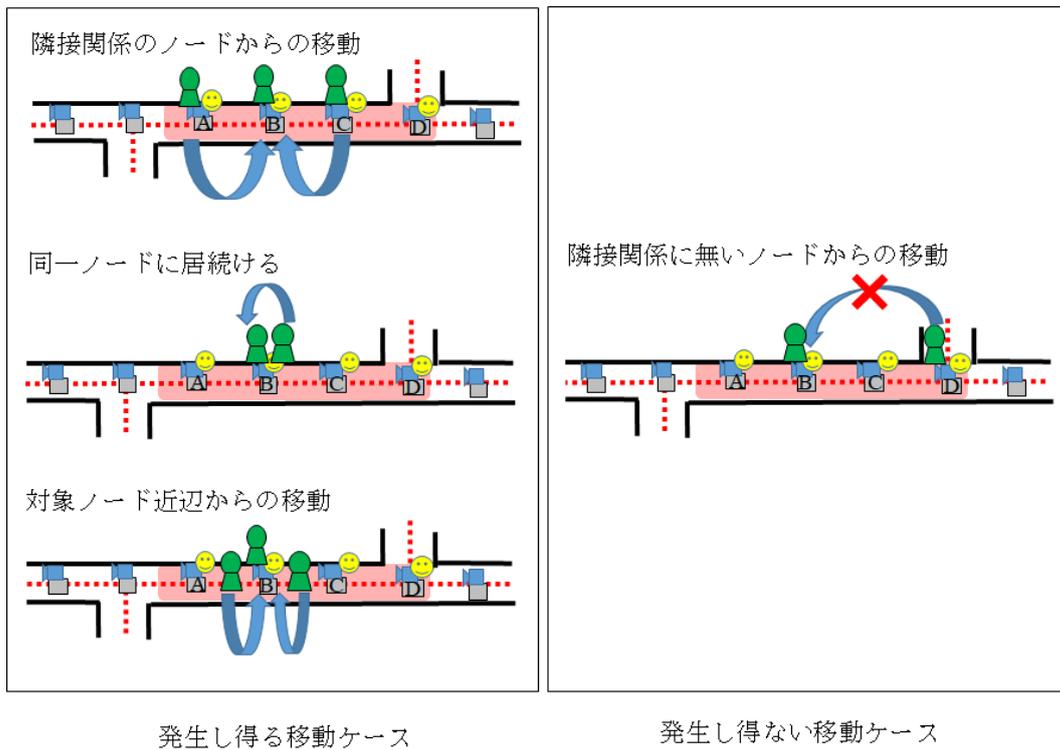


図 5.2: 追跡対象の移動パターン

とが困難になる。また、エージェントの増加により多くのノードの計算資源を使用することになる。そこで、追跡対象がグループ内の各ノードに存在する確率を計算する手法を提案する。確率の低いノード上のエージェントを削除することにより追跡効率を改善する。

## 5.3 人物の存在確率による誤追跡への対応

### 5.3.1 人物の移動パターン

追跡対象がグループ内の各ノードに存在している確率を計算するにあたりグループ内における追跡対象の移動ケースを考える。カメラは入口や交差点などの限られた場所に設置されるため、追跡対象がどのカメラにも映らない状態が発生する。たとえば、図 5.2 に示すノード B で追跡対象が観測されるケースは、ノード B の隣接関係ノードであるノード A や C から移動してくるケース、ノード B に観測され

続けるケース，およびノード B 近辺，すなわちノード A と B の間，または，ノード B と C の間から移動してくるケースによって発生する．一方，ノード B と隣接関係に無いノード D からの移動は必ずノード C を経由する必要があるため発生しない．そこで，まずグループをあるノードとその隣接関係ノードからなるサブグループへ分割する．サブグループ内に追跡対象が存在すると仮定しサブグループ内の各ノードやノード間に追跡対象が存在する確率を求める．そして，各サブグループ内で求めた存在確率をグループ内で集約することでグループ内における各ノードに追跡対象が存在する確率を求める．

### 5.3.2 提案手法の要点

提案手法ではグループ内の各ノードに加えてノード間に追跡対象が存在する場合も想定し確率の計算をおこなう．単純にグループ内のエージェントが検知している人物の確率の合計が1になるように正規化する方法も考えられるが，人物がノード間にいていずれのカメラにも検出されない場合に確率が正しく求まらない．そこで，グループを各ノードとそのノードを取り巻くノード間からなる細かい単位であるサブグループへ分割し，その単位の中で人物が存在する確率を求める．つづいて，追跡対象があるノードに存在していた場合，そのノード近辺に追跡対象が存在している可能性が高いと考えた．前節で示した様に人物はあるノードからその隣接するノードへ移動していき，急に離れたノードへ移動するということは無いためである．そこで，細かく分割した単位をグループ単位へ集約する際に，前回に追跡対象が各サブグループに存在していた確率で重み付けし最新の確率を求めるようにした．

### 5.3.3 サブグループ内存在確率

#### サブグループ

サブグループはある中心となるノードとその隣接関係ノードから構成される．図 5.3 にグループをサブグループへ分割する例を示す．ここではグループに3つのノード A, B, C が存在する．このときグループは3つのサブグループに分けられる．サブグループ1は中心ノード A とその隣接関係ノード B から構成される．サ

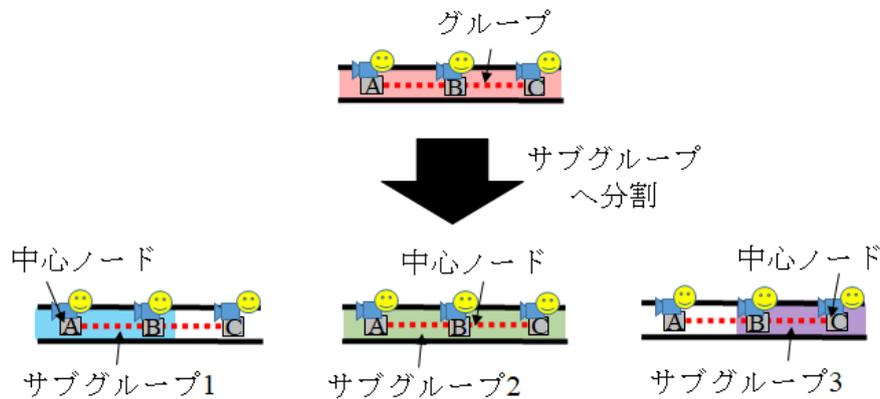


図 5.3: サブグループの例

サブグループ2は中心ノード B とその隣接関係ノード A および C から構成される。サブグループ3は中心ノード C とその隣接関係ノード B から構成される。

#### ノード上およびノード間の存在確率

サブグループでは、サブグループ内に追跡対象が存在すると仮定したときのサブグループ内の各ノードやノード間に追跡対象が存在する確率を求める。追跡対象がサブグループ内のノード、およびノード間に存在する確率を求めるために、中心ノードをノード A とするノード A, B からなるサブグループが存在し、人物 P1 がノード A で、人物 P2 がノード B で観測されたと仮定する。P1 が追跡対象である確率を  $s_A$ 、P2 が追跡対象である確率を  $s_B$  とする。このとき、P1 が追跡対象であるケース、P2 が追跡対象であるケース、どちらも追跡対象ではないケースが考えられる。

P1 および P2 のどちらも追跡対象でない確率は、P1 が追跡対象ではなく、かつ P2 が追跡対象ではない確率となるため式 (5.1) と表現できる。これはいずれのノードにも追跡対象がない確率、つまり中心ノードと隣接関係ノードとのノード間に追跡対象がいる確率となる。

$$(1 - s_A) \times (1 - s_B) \tag{5.1}$$

P1 が追跡対象である確率は、P1 が追跡対象であり、かつ P2 が追跡対象ではな

く、かつノード間にもいない確率となるため式 (5.2) のように表現できる.

$$s_A \times (1 - s_B) \times \{1 - (1 - s_A)(1 - s_B)\} \quad (5.2)$$

同様に、P2 が追跡対象である確率は式 (5.3) のように表現できる.

$$(1 - s_A) \times s_B \times \{1 - (1 - s_A)(1 - s_B)\} \quad (5.3)$$

式 (5.1) を一般化する. サブグループ内のノードが  $n$  個で、各ノードにおいて  $s_i$  ( $i$  は 1 以上  $n$  以下) の確率で追跡対象が存在すると検出されたとすると、ノード間に追跡対象がいる確率は式 (5.4) のように表現できる.

$$\prod_{i=1}^n (1 - s_i) \quad (5.4)$$

同様に、サブグループ内のあるノード  $m$  に追跡対象がいる確率を一般化すると式 (5.5) のように表現できる.

$$s_m \times \prod_{i=1, i \neq m}^n (1 - s_i) \times \left\{ \prod_{i=1}^n (1 - s_i) \right\} \quad (5.5)$$

サブグループ内に追跡対象が存在すると仮定しているため、サブグループ内における各ノードに追跡対象がいる確率、およびノード間に存在する確率の合計が 1 となるように正規化する. これにより、サブグループ内のノード  $m$  に追跡対象がいる確率  $S_m$  は式 (5.6) のように表現できる.

$$\begin{aligned} S_m &= \frac{(5.5)}{\sum_{i=1}^n (5.4) + (5.5)} \\ &= \frac{\{s_m \times \prod_{i=1, i \neq m}^n (1 - s_i) \times \{\prod_{i=1}^n (1 - s_i)\}\}}{\sum_{i=1}^n \{s_i \times \prod_{k=1, k \neq m}^n (1 - s_k) \times \{\prod_{k=1}^n (1 - s_k)\}\} + \{\prod_{i=1}^n (1 - s_i)\}} \end{aligned} \quad (5.6)$$

同様に、サブグループ内のノード間に追跡対象がいる確率  $S_{non}$  は式 (5.7) のように表現できる.

$$\begin{aligned} S_{non} &= \frac{(5.4)}{\sum_{i=1}^n (5.4) + (5.5)} \\ &= \frac{\{\prod_{i=1}^n (1 - s_i)\}}{\sum_{i=1}^n \{s_i \times \prod_{k=1, k \neq m}^n (1 - s_k) \times \{\prod_{k=1}^n (1 - s_k)\}\} + \{\prod_{i=1}^n (1 - s_i)\}} \end{aligned} \quad (5.7)$$

### 5.3.4 グループ内存在確率

各サブグループ内で求めた存在確率をグループ内で集約することでグループ内における各ノードに追跡対象が存在する確率を求める。ここでは、まず、図5.3に示すグループが存在していると仮定し、追跡対象がノードBに存在する確率を求める。グループ内において追跡対象がノードBに移動する確率は、下記のケース1~3の合計となる。

- ケース1: 最後に追跡対象がノードAで観測された後、ノードBで観測される確率。なお、ノードAで観測後は、ノード間に存在する場合を含む。以降のケース2,3も同様である。
- ケース2: 最後に追跡対象がノードBで観測された後、ノードBで観測される確率。
- ケース3: 最後に追跡対象がノードCで観測された後、ノードBで観測される確率。

ここで、各サブグループで求めたサブグループ内存在確率をサブグループ1は  $S1_A, S1_B, S1_{non}$ 、サブグループ2は  $S2_A, S2_B, S2_C, S2_{non}$ 、サブグループ3は  $S3_A, S3_B, S3_{non}$  とする。また、最後に追跡対象がノードA, B, Cで観測された確率を  $G_A, G_B, G_C$  と置き ( $G_A + G_B + G_C = 1$ )、前回評価時における確率を' (ダッシュ) をつけて  $G'_A$  のように表現する。このとき、ケース1は最後に追跡対象がノードAで観測された後 (確率:  $G'_A$ ) に、ノードBで観測される確率 ( $S1_B$ ) であるため式(5.8)のように表現できる。

$$G'_A \times S1_B \quad (5.8)$$

同様に、ケース2, 3は式(5.9), (5.10)のように表現できる。

$$G'_B \times S2_B \quad (5.9)$$

$$G'_C \times S3_B \quad (5.10)$$

ノードmの隣接関係ノードがn個あるとして一般化しケース1~3を合計するとノードmに追跡対象が存在する確率は式(5.11)のように表現できる。

$$\sum_{i=1}^n (G'_i \times S i_m) \quad (5.11)$$

次に、最後にノード B で観測された後、追跡対象がいずれのノードにも観測されないケース、つまりノード間に存在する場合を考える。前述の通りカメラは入り口や交差点などの限られた場所に設置される。そのため、どのカメラにも映らず観測されないケースがある。最後に追跡対象がノード B で観測された (確率:  $G'_B$ ) 後に、追跡対象が観測されない確率は  $S 2_{non}$  である。しかし、追跡対象が観測されないケースは、ノード間に存在するが観測されなかったケースとそもそもノード間に存在していなかったケースが考えられる。このため、最後にノード B で観測された後ノード間に存在していなかったケース、すなわち、追跡対象が最後にノード A またはノード C で観測された場合 ( $1 - G'_A - G'_C$ ) を除外して考える必要がある。これにより、最後にノード B で観測された後ノード間に追跡対象が存在する確率は式 (5.12) のように表現できる。

$$G'_B \times S 2_{non} \times (1 - G'_A - G'_C) \quad (5.12)$$

ここで、 $(G'_A + G'_B + G'_C) = 1$  であるため、数式 (12) を一般化しノード m での確率とすると数式 (5.13) となる。

$$G'_m \times S m_{non} \times G'_m \quad (5.13)$$

式 (5.11), (5.13) をグループ全体で正規化し、和を求めることで、それぞれのノードで最後に追跡対象が観測された確率  $G_m$  (式 (5.14)) を求めることができる。ここで、n はグループ内のサブグループ数、k はサブグループ内のノード数を表す。

$$\begin{aligned} G_m &= \frac{(5.11) + (5.13)}{\sum_{i=1}^n \{(5.11) + (5.13)\}} \\ &= \frac{\sum_{i=1}^k (G'_i \times S i_m) + (G'_m \times S m_{non} \times G'_m)}{\sum_{j=1}^n \{(\sum_{i=1}^k (G'_i \times S i_j)) + (G'_j \times S j_{non} \times G'_j)\}} \end{aligned} \quad (5.14)$$

### 5.3.5 存在確率を用いた人物追跡の流れ

グループ内存在確率を求めることにより、グループ内の追跡対象がいる可能性が高いノードや低いノードを知ることができる。このため、グループ内のエージェ

ント数が増加した場合に確率の低いノードのエージェントを削除する、または、確率が閾値以下の場合にそのエージェントを削除することで、追跡人物の見逃しを抑えつつ追跡を継続することができる。

存在確率を用いた人物追跡の流れを図5.4に示す。図5.4では従来の人物追跡の流れから変わる部分を赤色に示している。監視員は人物の追跡を始めるとき、その人物の特徴量を有するエージェントをその人物がいるノードへ派遣する。追跡対象エージェントは、コピーエージェントを自身の隣接ノードに派遣する。コピーエージェントはセンサから情報を収集し人物の検出を試みる。従来のグループ拡張による追跡では2つの閾値を用いて追跡をおこなった。対して、存在確率を用いた追跡ではこれらの閾値は使用せずに追跡をおこなう。エージェントが人物を検知した際に隣接関係にあるノードにエージェントを派遣しグループを拡張する。このとき、追跡対象がグループ内の各ノードに存在する確率が計算される。計算の結果、条件aあるいは条件bの方式を用いてエージェントを削除する。条件aではグループ内のエージェント数が上限に達した際に人物の存在確率が低いノード上のエージェントを削除する。条件bでは存在確率が閾値以下のノードのエージェントを削除する。これにより、グループ内のエージェントが増えすぎるのを抑えつつ、効率的に人物を追跡することができる。必ずエージェントの数を一定以下に抑えたいシーンでは条件aを利用し、ある程度の存在確率以上に着目して追跡したいシーンでは条件bを利用する想定である。従来の追跡とは異なり人物を確定することが無くなり、人物が存在しているノードが存在確率として求まる。監視員は得られる存在確率をもとに人物がいるおおよその位置を把握することができる。

## 第5章 人物存在確率の計算手法

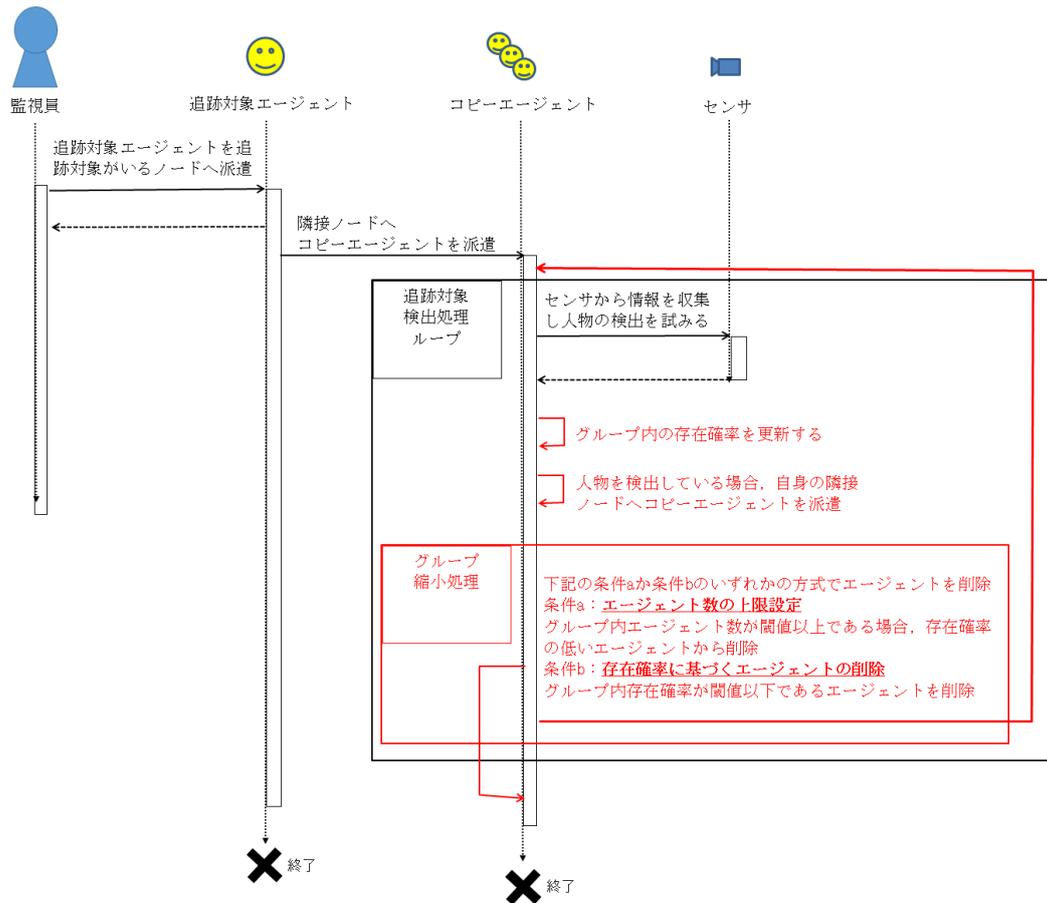


図 5.4: 存在確率による追跡の流れ

## 5.4 シミュレーション実験

提案したグループ内における人物の存在確率の計算手法の有効性を確認するためにシミュレーション実験を行った。

シミュレーションの実験環境は概ね4章で説明した実験環境と同じである。ただし、フロアに関してはセンサの台数を増やしている。

### 5.4.1 実験環境

#### マップ

実験で使用するフロアのマップを図5.5に示す。マップは実際に存在するフロアを模擬したものである。白マスは人物が通過する通路を表し、緑色丸がカメラを表す。カメラは30台がフロアに配置されている。エージェントはカメラがある30ノード上を移動する。フロア全体にエージェントが派遣された場合、1グループ内に最大で30個のサブグループおよびエージェントが存在することとなる。

#### 人物の移動およびグループ内確率の更新周期

人物は開始点からランダムに選択した2地点であるA点、およびB点を經由し終了点まで移動する。開始点からA点まで、A点からB点までおよびB点から終了点までは最短経路で移動する。人物の移動の例を図5.5に示している。図5.5の例では移動の開始から終了までに14個のカメラで検出される。図5.5中の赤字の数字が經由するカメラの順を表す。人物は8人が同時に移動をおこない、各人物は白マスを1つずつ移動する。

人物が白マスを1つ進む都度、追跡するエージェントのグループ内存在確率を更新する。

#### 識別誤りの再現方法

公開されている画像データに対し人物の認識アルゴリズムを適用し人物間のベクトル距離のデータを生成した。公開されている画像データとしてはSARC3D<sup>56)57)</sup>の画像を用いた。SARC3Dは50人の前方、後方、左側、右側の4方向から撮影し

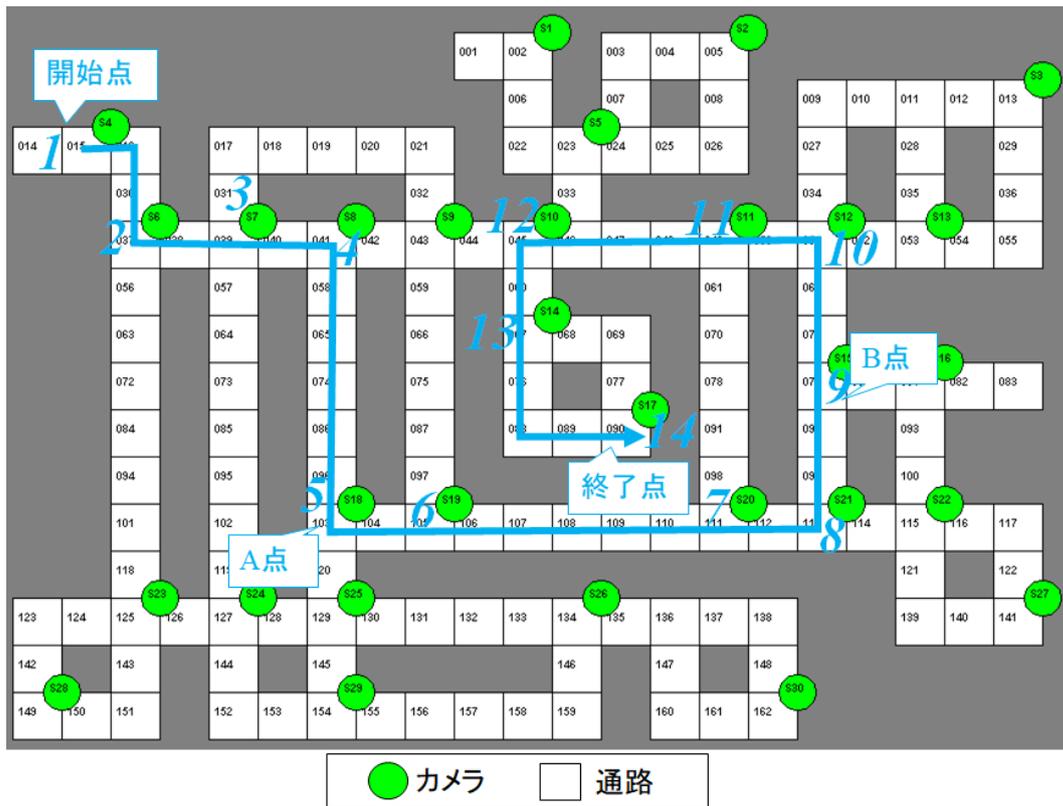


図 5.5: シミュレーションマップ

た合計 200 枚の画像から構成される。これらの画像に対して西山ら<sup>55)</sup>が提案している人物の認識アルゴリズムを使用し画像間の距離を求めた。同一人物の画像間の距離と異なる人物の画像間の距離の分布を図 4.5 に示す。エージェントは以下の状況 1, 2 に応じてシミュレータから人物の距離を取得する。

- 状況 1. 追跡対象がカメラの設置されたノードに現れた場合、図 4.5 に示した同一人物の分布の中からランダムに選択した距離を取得する
- 追跡対象ではない人物がカメラの設置されたノードに現れた場合、図 4.5 に示した異なる人物の分布の中からランダムに選択した距離を取得する。

状況 1, 2 に応じて距離を取得することで、取得した距離が追跡対象と非追跡対象で逆転するケースが起き、識別誤りが発生する状況を再現できる。取得した距離は距離の分布に応じた 0~1 の確率へ変換する。確率への変換に際してはシグモイド関数を用いて距離を 0~1 へ変換した。

### 実験内容

提案手法の追跡の精度を評価し、続いて従来手法である 2 つの閾値を用いた追跡との比較をおこなった。各実験結果は人物が 8 人同時に移動するシミュレーションを 100 セット行った結果の平均を示す。

#### 5.4.2 予備実験

実験の実施にあたって、まず追跡対象を 1 人とし図 5.5 に示した青矢印のルートで移動した場合に期待通り追跡が行えるかを確認した。図 5.6 に開始点直後、A 点到着後、B 点通過後、および終了点に到着したときの追跡の様子を示す。予備実験ではグループ内のエージェント数の上限を 9 に設定し、エージェント数が上限に達した際に最も存在確率の低いノードのエージェントを削除する方式で追跡をおこなった。まず図 5.6 (a) は開始直後の状況を示しており、赤色四角の A1 がエージェントを表し、P1 が追跡対象を表す。数字が人物の存在確率を表している。開始直後で追跡対象がいる場所は既に分かっているため、確率は 1.000 として追跡を開始している。つぎに図 5.6 (b) は追跡対象が A 点到着後の状況を示しており、A 点に人物がいる確率が 0.996、その前に通過したカメラ S9 では、存在確率

## 第5章 人物存在確率の計算手法

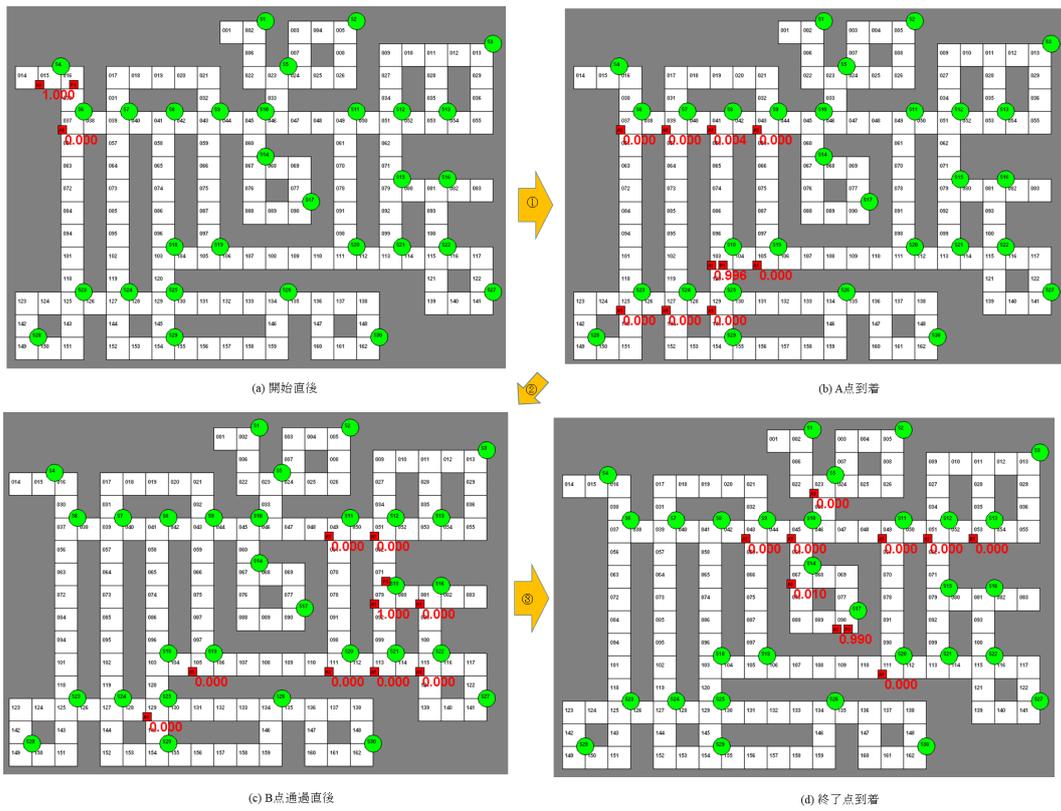


図 5.6: 予備実験

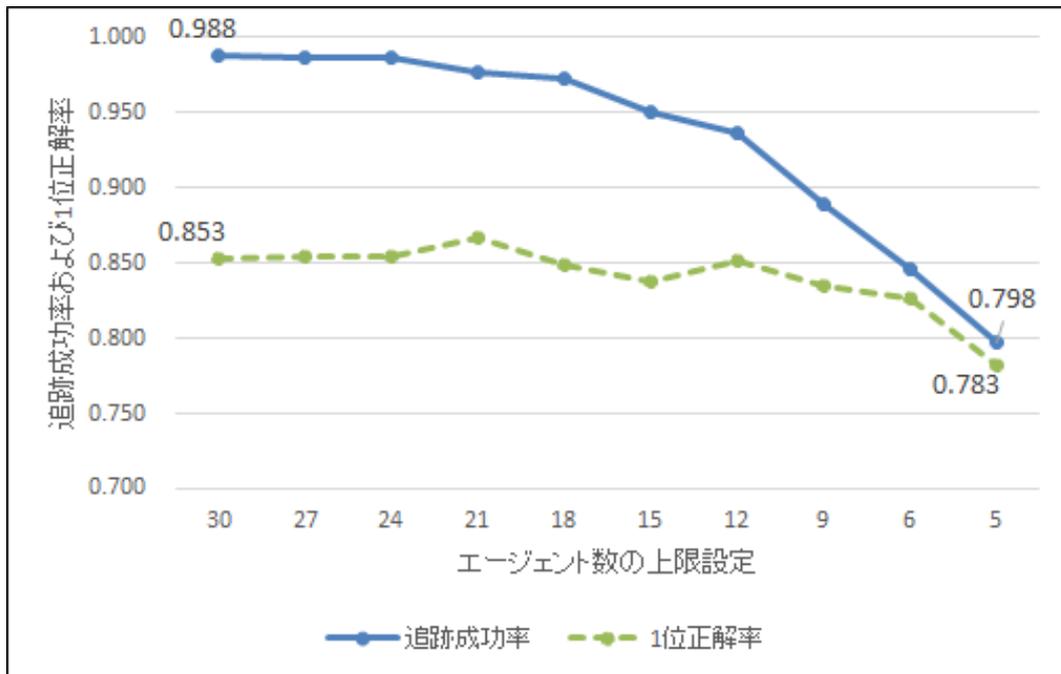


図 5.7: エージェント数の上限設定時

が 0.004 となっている。そして図 5.6 (c) は追跡対象が B 点通過後の状況を示しており、B 点に人物がいる確率は 1.000 となっている。最後に図 5.6 (d) は追跡対象が終了点に到着した状況を示しており、終了点にいる確率が 0.990、その前に通過したカメラ S14 では、存在確率が 0.10 となっている。以上に示した様に追跡対象の移動にともない追跡対象がいる地点の存在確率が最も高い値として得られていることが分かる。また各断面においてエージェント数が 9 以下に抑えられていることが分かる。

### 5.4.3 実験 1：提案手法評価

**エージェント数の上限設定** エージェント数が上限に達した際にグループ内存在確率が低いノード上のエージェントを削除する方式を実装し、追跡成功率として人物が終了点に到達したときに人物がグループ内に存在する確率を評価した。さらに、人物が移動を終えた時点での 1 位正解率として人物が存在するノードのグループ内存在確率がグループ内で 1 位である割合を評価した。エージェント数の上限を 30~5 まで変化させた場合の実験結果を図 5.7 に示す。

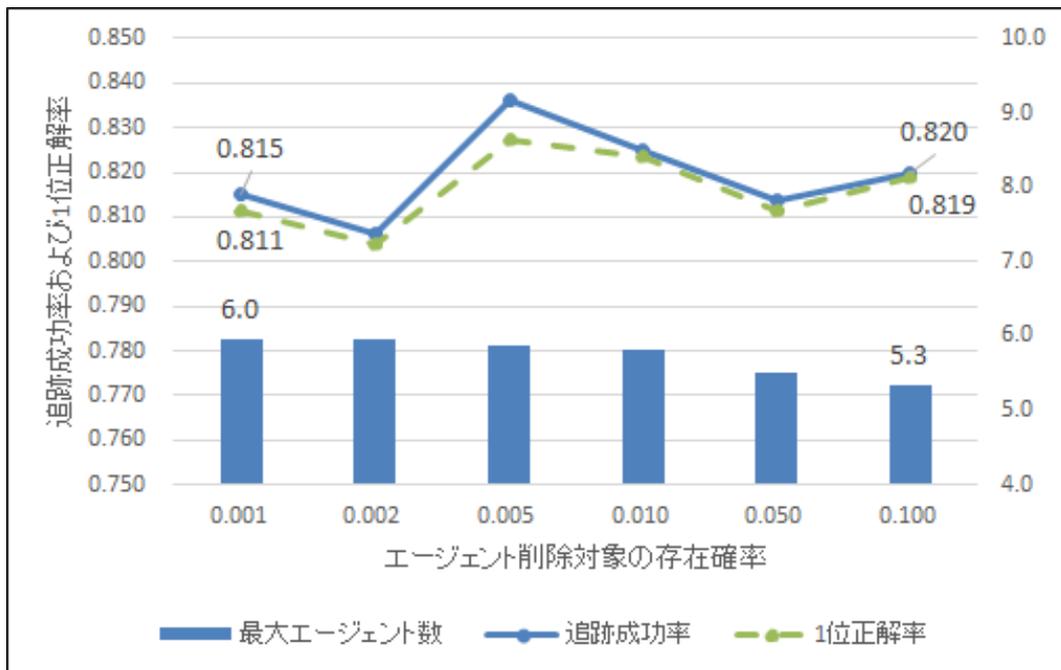


図 5.8: 存在確率に基づくエージェント削除時

提案手法ではエージェント数の上限を下げていくと追跡成功率が低下していくことが分かる。エージェント数の上限なし (30) では追跡成功率が 0.988 に対し、エージェント数の上限が 5 の場合では追跡成功率が 0.798 となった。一方、1 位正解率はエージェント数上限なし (30) の場合で 0.853、上限を 5 にした場合で 0.783 であり、グループ内のエージェント数を減らしても 1 位正解率を高く維持できていることがわかる。グループ内存在確率を計算することで、追跡対象がいる確率が高いノードのエージェントが残り、追跡対象がいる確率が低いノードのエージェントが削除される。この結果、グループ内存在確率が高いノードのエージェントは削除されずにグループ内のとどまることができ、グループ内のエージェント数を減らしても 1 位正解率を維持できたものと考えられる。

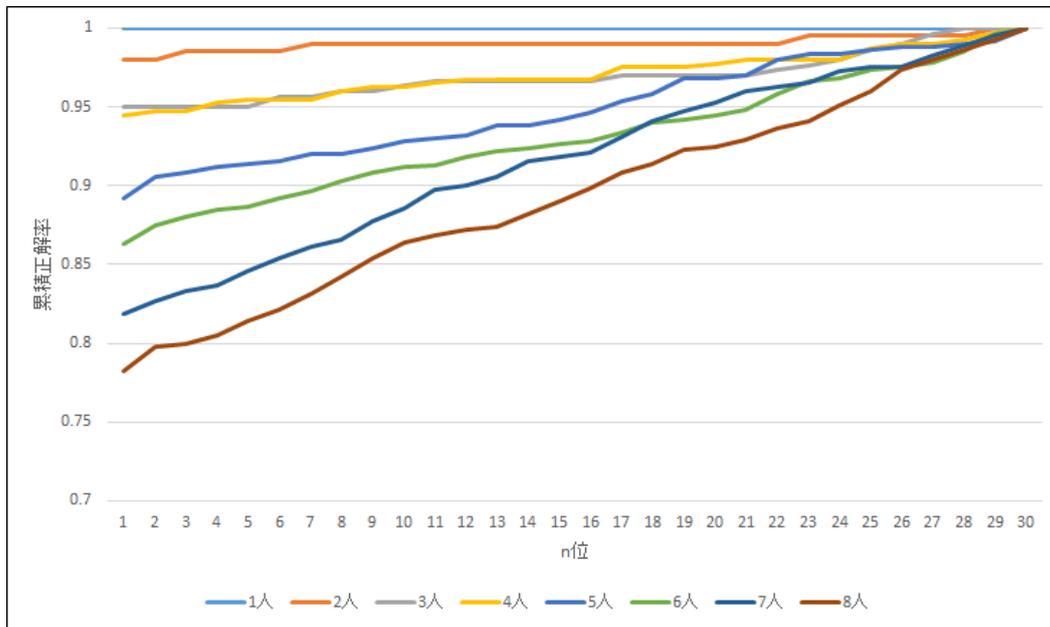
**存在確率に基づくエージェントの削除** グループ内のエージェント数に上限は設けるのではなく、グループ内存在確率が予め設定した値を下回った場合に当該ノードに存在するエージェントを削除する方式を実装し、追跡成功率、1 位正解率、およびグループ内のエージェント数を確認した。実験結果を図 5.8 に示す。

エージェント削除対象の存在確率が 0.001 を下回った時に削除した場合、追跡

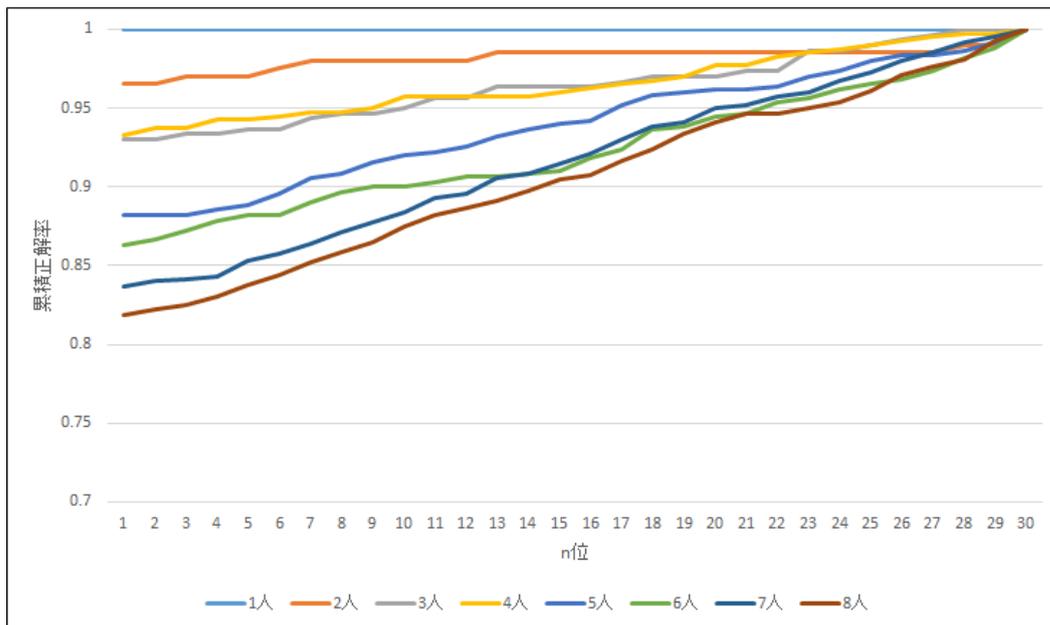
成功率は0.815に、1位正解率は0.811となった。0.100の場合では、追跡成功率は0.820に、1位正解率は0.819となった。また、エージェント削除対象の存在確率を低く設定するとエージェント数が若干増加する傾向にあるが5.3~6.0の間に収まっていることが分かる。エージェント数の上限を5,6と設定した場合の結果である図5.7と比較すると概ね近い追跡成功率、一位正解率となった。

**累積n位正解率** グループ内のエージェント数に上限を設ける方式とグループ内存在確率が予め設定した値を下回った場合に当該ノードに存在するエージェントを削除する方式のそれぞれについて、累積n位正解率を確認した。前者の方式ではエージェント数の上限を5に、後者の方式では存在確率の下限を0.10とし、さらに追跡対象の同時移動人数を1人から8人まで変化させ累積n位正解率を確認した。それぞれの結果を図5.9に示す。いずれの方式を用いた場合でも、累積n位正解率の傾向に大きな差異は無いことが分かる。また、同時移動人数を増やすごとに累積n位正解率は低下する傾向にあることが分かる。これはグループ内に存在する非追跡対象が増え、かつ非追跡対象の確率が偶然に高かった場合にそのノードの確率が一時的に高くなるケースが増えたためと考える。ついで、累積正解率の収束の傾向、つまり何位までで正解率が1.0となるかについては、収束は遅く下位までの確認が必要であることが分かる。

グループ内のエージェントの削除をまったくおこなわなかった場合の累積n位正解率を確認した。結果を図5.10に示す。エージェントの削除をおこなわない場合ではエージェントの削除をおこなう場合と比較して、累積正解率の収束は早いことが分かる。エージェントの削除をおこなう場合では、グループ内に存在する非追跡対象が増え、かつ非追跡対象の確率が偶然に高かった場合にそのノードの確率が一時的に高くなるケースが増える。相対的に追跡対象の確率は下がることになるため、そのタイミングで追跡対象を追跡していたエージェントが誤って削除されるケースが増える。対してエージェントを削除しない場合、誤って追跡対象を追跡しているエージェントが削除されることが無いため、累積正解率の収束は早まったものとする。しかし、エージェントを削除しないというのはエージェント数の増加を抑制するため存在確率を計算しているという趣旨から外れることとなる。誤って追跡対象を追跡しているエージェントを削除してしまった場合、どのグループにも追跡されている追跡対象が、あるノードにおいて検出されることとなる。隠れ隣接関係を用いることで人物を見失った場合に人物を継続して追跡



(a) エージェント数の上限設定=5



(b) エージェント削除対象の存在確率=0.1

図 5.9: グループ縮小有りの場合の累積  $n$  位正解率

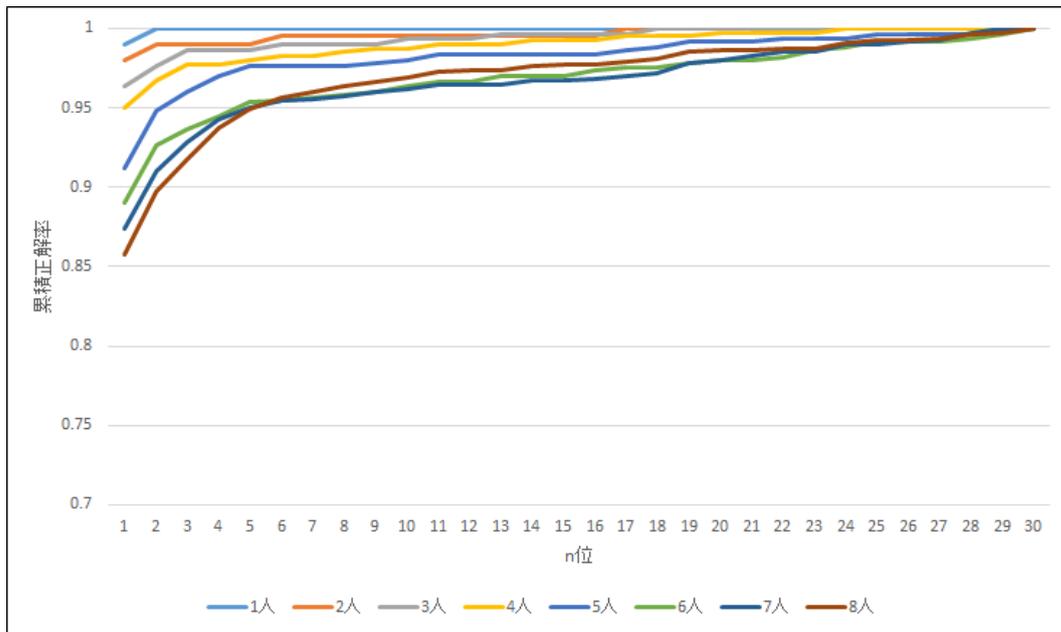


図 5.10: グループ縮小無しの場合の累積 n 位正解率

する手法を提案しており，存在確率を用いた追跡においても同様の手法を適用することが可能であると考え．つまり，一時的に追跡対象ではないとみなされた場合であっても，再度グループ内へ追跡対象とみなし追跡を継続して行えるようにすることで存在確率を用いた追跡においても追跡精度が改善できるのではと考える．

#### 5.4.4 実験 2：従来手法との比較

提案手法と従来手法の比較を行った．従来手法では2つの閾値を設けグループを拡張するが，エージェント数を制限する仕組みはない．そこで，従来手法ではエージェント数が上限に達した場合に，LRU (Least Recently Used: 人物を最後に検知してから最も時間が経過しているエージェントから削除)，あるいはLFU (Least Frequently Used: 人物を検知した回数が最も少ないエージェントから削除) 方式でエージェントを削除することとした．

提案手法によりエージェント数に上限を設けた方式と，従来手法の2つの閾値を色々変えた代表3パターン (0.9, 0.7)，(0.8, 0.4)，(0.7, 0.2) の追跡成功率を図 5.11

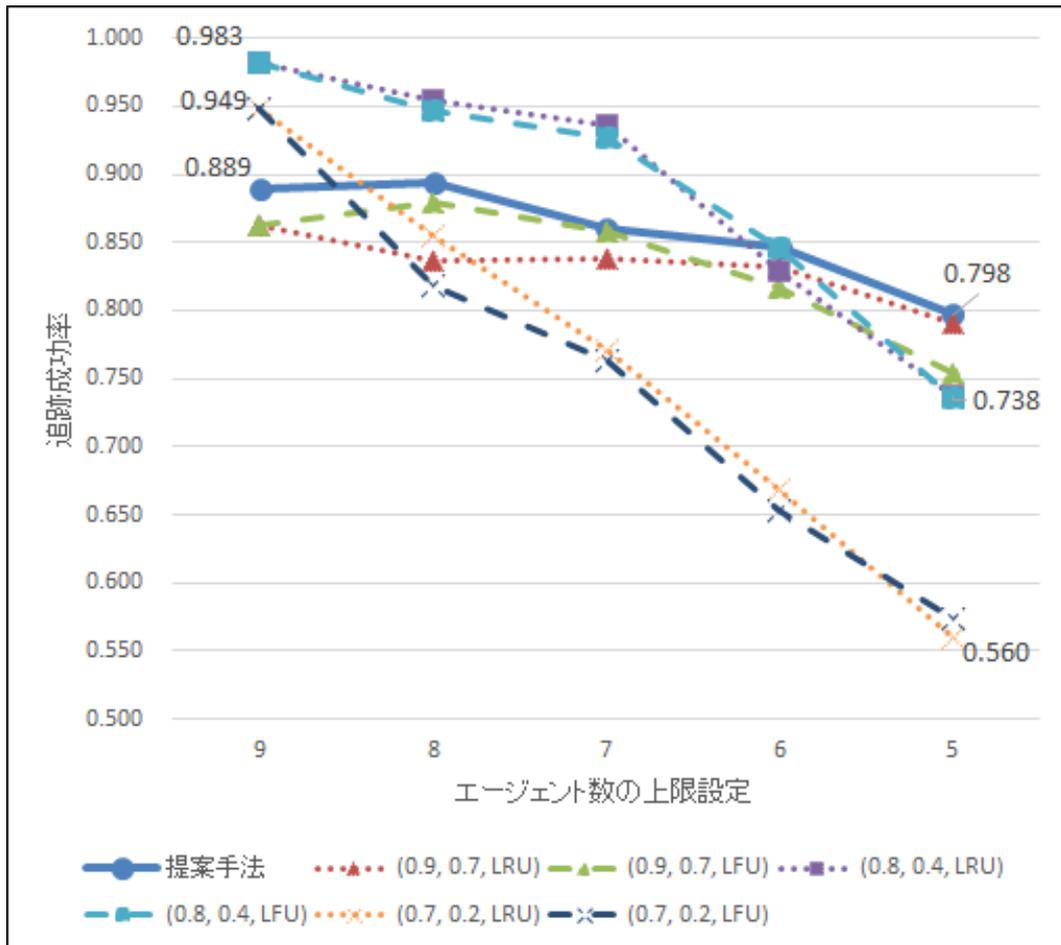


図 5.11: エージェント数固定時

に示す。ここで、(0.9, 0.7)のパターンは2つの閾値を高め設定したパターン、(0.8, 0.4)は2つの閾値を0.1刻みで変化させたとき最も追跡成功率が高かったパターン、そして(0.7, 0.2)は2つの閾値を低めに設定したパターンである。図5.11における括弧内の3つの値は(確定閾値, 再評価閾値, 削除方式 LRU or LFU)を表す。

実験結果を見るとエージェント数の上限を下げるとすべてのパターンにおいて追跡成功率が低下している。しかし、提案手法では追跡成功率の低下が少ない。従来手法においてエージェント数の上限を9に設定した場合、追跡成功率は(0.8, 0.4, LRU)で0.983、(0.7, 0.2, LRU)で0.949となっており提案手法を上回っていることが分かる。しかし、エージェント数の上限を5に設定した場合、(0.8, 0.4, LRU)では0.738、(0.7, 0.2, LRU)では0.560となった。提案手法においてエージェント数の上限を5に設定した場合の追跡成功率は0.798であり、従来手法より提案手法の方が高い追跡成功率であることがわかる。他のパターンにおいてもエージェント数の上限が5の場合では提案手法の追跡成功率が最も高くなっていることがわかる。

実験では人物がランダムに選択された2地点間を移動する。このため、LFU, LRU方式の削除方式ではエージェントが比較的ランダムに削除される。一方、提案手法では追跡対象のグループ内存在確率が計算され、存在確率が低いエージェントが削除される。また、図5.7の実験結果で見られるように提案手法の一位正解率が高い。このため、エージェント数の上限低下に伴う追跡成功率の低下が抑えられ、エージェント数の上限が5の場合では提案手法の追跡成功率が最も高くなったものだと考えられる。

### 5.4.5 考察

本節では提案手法の制限について述べる。提案手法では、図4.5に示すように追跡対象の距離が非対象の距離よりも、近くなる傾向があるという条件が必要である。つまり、提案手法ではあるノードの確率が高く計算された場合にはそのノードの隣接関係ノードの確率も高くなるように計算する。したがって、非追跡対象の確率が偶然に高い場合、そのノードの隣接関係ノードの確率も高くなるように計算する。これらの確率計算は期待していない結果である。ただし、これらの確

率は一時的なものであると考える。というのも、図 4.5 に示すように追跡対象の距離は非追跡対象の距離よりも近くなる傾向があると前提を置けるため、一時的に確率が高く計算されたとしても、いくつかのノードの評価を経て徐々に適切な値になると考える。

実際の環境では、同じルートを繰り返し歩くことや、後ろを向くことなど、追跡対象のルートをシミュレーションより遥かに長く複雑なものになると想定される。シミュレーションでも同じ道を繰り返し歩くことや、来た道に戻るといった移動経路は含まれている。頻繁に人物を検出することでグループ内の確率が頻繁に更新され、確率が高くなりすぎることや低くなりすぎる可能性がある。このような場合、画像の背景差分等の技術を利用し同じ人物が繰り返し検出された場合に、結果を集約する等の調整が必要になると考える。

### 5.5 結言

本章では人物追跡システムにおいて発生する人物の識別誤りに対応した従来の追跡手法を改良し、より効率よく追跡すべく人物の存在確率を計算する手法を提案した。提案手法では、グループを隣接関係に基づいてサブグループへ分割し、まずサブグループ単位で人物の存在確率を計算した。そして、計算した人物の存在確率を、さらに隣接するサブグループ間で集約することでグループ内における人物の存在確率を計算した。シミュレーション実験の結果、本手法により追跡効率が改善されることを確認した。従来のグループ拡張を用いた追跡より少ないエージェントの利用により、人物の追跡が可能となった。

一方、提案手法の累積  $n$  位正解率を確認したところ、累積正解率の収束の傾向、つまり何位までで正解率が 1.0 となるかについては、エージェント数を低く抑えた場合では収束は遅く下位までの確認が必要であることが分かった。監視員が確認するノードの数は少なければ少ないほど監視の手間は軽減できるため、さらなる改善が必要である。提案手法では、同一の追跡対象を追跡している 1 グループ内において、現在検出している人物の確率、および前回人物を検出していたときの確率を用いて存在確率を計算した。他者を追跡している他グループからの評価も組み込むことで、さらに確率を補正し追跡精度を改善できるのではと考えている。しかし、その場合グループ間で評価が分かれるケースが発生する可能性もあ

るため、どちらのグループの評価を正しいとみなすかグループ間の調整が必要となる。エージェント間で交渉に関する研究<sup>65)66)</sup>も為されており応用できるのではと考える。

提案手法では人物を検出したときの確率をもとにグループ内の存在確率を計算したが、グループの形状や拡張の仕方も追跡の際に活用できる情報になるのではと考える。たとえば、追跡対象が2名で、2名ともが同一のノードにいる場合、それぞれの人物が区別できなかつたとしても、それぞれのグループの追跡対象であることは間違いない。このような場合に、確率を補正することは可能である。この他にも、異なるグループ同士の重なり方や人物が存在するノードのパターンによって、確率を補正することが可能なケースは存在すると考える。

提案手法と従来のグループ拡張を用いた手法では、エージェント数の上限を全く気にしない場合では図5.11に示したように従来手法のほうが追跡成功率が高い場合があった。計算資源に余裕がある間は従来のグループ拡張による追跡をおこない、計算資源を圧迫してきたときに提案手法に切り替える等のそれぞれの手法を共存させた追跡手法も今後検討する予定である。

## 第6章 結論

本論文では，エージェントを用いたリアルタイムな人物の追跡をおこなうための人物追跡システムの実用化に焦点をあて，実用化の際に発生する人物の識別誤りに対応した追跡手法を提案した．センサによって人物の識別が適切に為されず誤検出が発生する状況下においても，エージェントを利用したシステムにおいて人物の追跡を継続するための人物追跡システムを提案した．

提案手法の誤追跡へ対応の評価に先立ち，従来提案してきた基礎となるエージェント技術による追跡を Raspberry Pi 上に構築し，期待通り人物追跡が可能であることを確認した．

つづいて，従来提案されていたエージェントによる追跡手法を拡張し，2つの閾値を用いた追跡手法を提案した．1つ目の閾値は追跡対象であると確定するための確定閾値，2つ目の閾値は人物の確定を先送りすることを目的とした再評価閾値である．2つの閾値を用いることで，確定に至らないが確定の見込みがある人物を検出した際には，一時的に確定を保留し別の機会に再評価することが可能になる．また再評価の仕組みを実現するにはエージェントの派遣先を一時的に拡大する必要がある．そこで，人物がある範囲に居るとみなすエージェントのグループによる追跡を提案した．提案手法の有効性を確認するためシミュレータを製作し人物の誤検出が発生する状況を模擬し，提案手法によって誤検出による追跡精度の低下が軽減できることを確認した．

エージェントのグループによる追跡により，人物の識別誤りによる見逃しや誤追跡を抑えつつ人物を追跡することができた．さらに，グループによる追跡の効率を改善すべくグループ内のどこに人物がいる可能性が高いかの存在確率を計算する手法を提案した．存在確率の計算により，グループ内のどのノードに追跡対象が存在する可能性が高いかを知ることができる．追跡対象が存在する可能性が低いノードをグループによる監視対象から外すことにより，グループの拡張を抑えつつ，効率的に追跡対象を追跡することができる．シミュレーション実験によ

り，従来のグループによる追跡と同等の精度を保ちつつ，かつエージェント数を抑えた追跡ができることを確認した

提案手法ではエージェントのグループを定義し，さらに隣接関係にあるノード間における存在確率を計算する手法を提案した．これらに加え，たとえば他グループ間での連携や人物が頻繁に通過する経路情報を利用することでさらなる精度向上も可能であると考え．今後，実用化に向けさらなる改善を試みていくことを計画している．

## 謝辞

本研究を進めるにあたって、鳥取大学大学院工学研究科 菅原 一孔 教授，川村 尚生 教授，高橋 健一 准教授には終始ご指導頂きました。深夜休日問わず論文をチェックして頂き心から感謝申し上げます。

人物の識別アルゴリズムや人物間の距離データの利用については 西山 正志 准教授のご協力を頂きました。誠にありがとうございました。

鳥取大学大学院工学研究科 情報エレクトロニクス専攻 人物追跡班の 宮崎 喬行 氏，牧田 岳大 氏，中谷 啓太 氏，檜垣 裕介 氏，また社会情報システム/計算機応用研究室の修士および学部生の皆様には本研究を進めるにあたって様々な力添えを頂きました。心から感謝申し上げます。

メルコ・パワー・システムズ株式会社におきましては，博士課程への進学を後押ししてくださり様々な面でアドバイスをくださった 柿内 博人 氏を初め，様々な面で支えてくださった社長の 稲岡 孝雄 氏，技術本部長の中道 巧二 氏，顧問の糟谷 武則 氏，技術部長の 淵上 政人 氏に心から感謝申し上げます。また共同研究において技術的な面でサポートを頂きました 四元 辰平 氏，谷川 浩三 氏，そして研究と業務の両立に際して様々な調整を頂きました 宗宮 正 氏，田正司 昌則 氏に心から感謝申し上げます。

英語論文の執筆や発表に際し，専門外にも関わらず熱心に内容を読みこみ様々なコメントを頂きました Jean-Yves Terreault 氏に心から感謝申し上げます。

本研究は多くの方に支えられ，まとめることができました。関係者各位には心から感謝申し上げます。今後の皆様の御健勝と御発展を祈り申し上げ，感謝の言葉とさせていただきます。

最後に，博士課程において研究を進める中で心身ともに大きな支えとなり，また励ましてくれた家族に深く感謝申し上げます。

令和2年7月 塩塚 大

## 参考文献

- [1] 四元辰平, 谷川浩三, 辻美希, 高橋健一, 川村尚生, 菅原一孔, ” 人物追跡システムにおける局所化された隣接関係の計算, ” 電子情報通信学会論文誌, Vol. J99-D, No. 5, pp. 468-477, 2016.
- [2] 四元辰平, 塩塚大, 川村尚生, 高橋健一, 菅原一孔, ” 人物追跡システムにおけるセンサの不確実性に対応するための隠れ隣接関係の導出, ” 人工知能学会論文誌, Vol. AG16-J, No. 1-10, 2017.
- [3] M. Shiozuka, T. Yotsumoto, K. Takahashi, T. Kawamura, and K. Sugahara, ” Agent-based Tracking Method Addressing Target Recognition Errors, ” IEEJ Transactions on Electronics, Information and Systems, Vol. 140, No. 4, pp. 484-491, 2020.
- [4] 村松大吾, 岩間晴之, 木村卓弘, 榎原靖, 八木康史, ” 一歩行映像から取得される複数特徴を用いた個人認証, ” 電子情報通信学会論文誌 A, Vol. J97-A, No. 12, pp.735-748
- [5] S. Xia, Y. Liu, G. Yuan, M. Zhu, and Z. Wang, “ Inddor Fingerprint Positioning Based on Wi-Fi: An Overview, ” International Journal of Geo-Information, Vol. 6, No. 5, 2017.
- [6] Y. Jiang, et al., “ ARIEL: Automatic Wi-Fi based Room Fingerprinting for Indoor Localization, ” Proceeding of the 2012 ACM Conference on Ubiquitous Computing, pp. 441-450, 2012.
- [7] T. Pulkkinen, T. Roos, and P. Myllymaki, “ Semi-supervised Learning for WLAN Positioning, ” Proceeding of 21st International Conference on Artificial Neural Networks and Machine Learning (ICANN 2011), Vol. 6791, pp. 355-362, 2011.

- [8] 谷内大祐, 前川卓也, ”位置フィンガープリントの自動更新を用いた電波環境変化に頑強な屋内位置推定,” 情報処理学会論文誌, Vol. 55, No. 1, pp. 280-288, 2014.
- [9] 久保田僚介, 田頭茂明, 荒川豊, 北須賀輝明, 福田晃, ”無線LANを用いた屋内位置推定における学習コスト削減のための高精度データ補間手法,” 情報処理学会論文誌, Vol. 54, No. 4, pp. 1609-1618, 2013.
- [10] 宮崎 喬行, 牧田 岳大, 高橋 健一, 川村 尚生, 菅原 一孔, ”見守りシステムにおける人体の影響による推定位置補正,” マルチメディア、分散、協調とモバイルシンポジウム論文集 (2019).
- [11] K. Komai, M. Fujimoto, Y. Arakawa, H. Suwa, Y. Kashimoto, and K. Yasumoto, ”Elderly Prson Monitoring in Day Care Center using Bluettoth Low Energy,” 10th International Symposium on Medical Information and Communication Technology (ISMICT 2016), Worcester, MA, USA, pp. 140-144, 2016.
- [12] 藤本まなと, 駒井清顕, 荒川豊, 諏訪博彦, 柏本幸俊, 安本慶一, ”デイケアセンターにおける高齢者の行動履歴自生成システムの開発,” マルチメディア、分散、協調とモバイルシンポジウム論文集 (2016).
- [13] Random Forests, <https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf>, Aprila, 2020.
- [14] 浅野正義, 梶原祐輔, 島川博光, ”電子タグを用いた単独歩行視覚障害者の誘導,” 情報科学技術フォーラム講演論文集, Vol. 14, No. 4, pp. 37-42, 2015.
- [15] 杉野恭兵, 丹羽佑輔, 白松俊, 大園忠親, 新谷虎松, ”Bluetoothに基づく人感センサを利用した遠隔介護支援システムの試作,” 電気学会論文誌, Vol. 136, No. 2, pp. 1-8, 2016.
- [16] S. Nirjon, J. Liu, G. DeJean, B. Priyantha, Y. Jin, and T. Hart, ”COIN-GPS: indoor localization from direct GPS receiving,” Proceedings of the 12th International Conference on Mobile Systems, Applications and Services (MobiSys 2014), pp. 301-314, 2014.
- [17] soranome, <https://soranome.net/>, June, 2020.
- [18] GPS BoT, <https://www.bsize.com/bot/gps>, June, 2020.
- [19] どこかな GPS, <https://www.dokokana-gps.jp/>, June, 2020.

- [20] 保苺佳彦, 永瀬宏, 水野舜, ” 赤外線センサ群を用いた人物位置推定システムの提案, ” 第 10 回情報科学技術フォーラム (FIT2011), pp. 397-398, 2011.
- [21] T. Shuai, M. Kudo, B.-N. Pei, H. Nonaka, and J. Toyama, “ Multiperson Locating and Their Soft Tracking in a Binary Infrared Sensor Network, ” IEEE Transactions on Human-Machine Systems, Vol. 45 , Issue 5, pp. 550-561, 2015.
- [22] S. Suzuuchi, and M. Kudo, “ Location-associated indoor behavior analysis of multiple persons, ” 2016 23rd International Conference on Pattern Recognition (ICPR), pp. 2079-2084, 2016.
- [23] 椿野駿平, 工藤 峰一, ” 赤外線センサを用いた複数人の室内行動データセットの構築, ” 研究報告デジタルコンテンツクリエーション (DCC), Vol. 32, pp. 1-4, 2017.
- [24] M.Isard and A.Blake, “ CONDENSATION – Conditional Density Propagation for Visual Tracking, ” International Journal of Computer Vision, Vol. 29, No. 1, pp. 5-28, 1998.
- [25] B. Babenko, M.-H. Yang, and S. Belongie, “ Robust Object Tracking with Online Multiple Instance Learning, ” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, Issue 8, pp. 1619-1632, 2011.
- [26] L. Zhang and L. van der Maaten, “ Preserving Structure in Model-Free Tracking, ” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 36, No. 4, pp. 756-769, 2014.
- [27] L. Wenxi, C. Antoni, L. Rynson, and M. Dinesh, ” Leveraging long-term predictions and online learning in agent-based multiple person tracking, ” IEEE Transactions on Circuits and Systems for Video Technology, 25.3, pp. 399-410, 2015.
- [28] Z. Jin and B. Bhanu, “ Multi-camera Pedestrian Tracking using Group Structure, ” International Conference on Distributed Smart Cameras Article, No. 2, 2014.
- [29] Y.J. Cho, S.A. Kim, J.H. Park, K. Lee, and K.J. Yoon, “ Joint Person Re-identification and Camera Network Topology Inference in Multiple Camera, ” arXiv:1710.00983, 2017.

- [30] K. Yoon, Y. Song, and M. Jeon, “ Multiple Hypothesis Tracking Algorithm for Multi-Target Multi-Camera Tracking with Disjoint Views, ” arXiv: 1901.08787, 2019.
- [31] J.R. -Rubio, I. Alexiou, and A. A. Bharath, “ Appearance-based indoor localization: A comparison of patch descriptor performance, ” Pattern Recognition Letters, Vol. 66, pp. 109-117, 2015.
- [32] 鄭明燮, 辻健太郎, 宮崎信浩, 松田裕司, 馬場孝之, 瀬川英吾, 上原祐介, “ 顔が判別できない低解像度映像を用いた複数カメラ間の人物同定, ” 研究報告コンピュータビジョンとイメージメディア (CVIM), 2017-CVIM-206, No. 14, pp. 1-6, 2017.
- [33] 中沢篤志, 加藤博一, 井口征志, “ 分散カメラエージェントを用いた広域人物位置検出システム, ” 情報処理学会論文誌, Vol. 41, No. 10, pp. 2895-2906, 2000.
- [34] 中沢篤志, 日浦慎作, 加藤博一, 井口征士, “ 分散視覚エージェントを用いた複数人物追跡システム, ” 情報処理学会論文誌, Vol. 42, No.11, pp. 2699-2710, 2001.
- [35] P.-F. Wu et al., “ Node Scheduling Strategies for Achieving Full-View Area Coverage in Camera Sensor Networks, ” Sensors 17 (6), 1303, 2017.
- [36] S. Hanoun et al., “ Target coverage in camera networks for manufacturing workspaces, ” Journal of Intelligent Manufacturing, Vol. 27, pp. 1221–1235, 2016.
- [37] 鈴木達也, 岩崎慎介, 小林貴訓, 佐藤洋一, 杉本晃宏, “ 環境モデルの導入による人物追跡の安定化, ” 電子情報通信学会論文誌, Vol. J88-DII, No. 8, pp. 1592-1600, 2005.
- [38] 杉村大輔, 小林貴訓, 佐藤洋一, 杉本 晃宏, “ 行動履歴に基づいた環境属性の自動構築を伴う三次元人物追跡, ” 画像の認識・理解シンポジウム (MIRU2007), pp. 405-411, 2007.
- [39] S. Alletto, R. Cucchiara, G. Del Fiore, L. Mainetti, V. Mighali, L. Patrono, and G. Serra, “ An Indoor Location-Aware System for an IoT-Based Smart Museum. ” IEEE Internet of Things Journal, pp. 244-253, 2016.

- [40] C. Alejandro, M. Antoni, B. Marc, and V. Jose, "Navigation system for elderly care applications based on wireless sensor networks," Signal Processing Conference (EUSIPCO 2012), Proceedings of the 20th European. IEEE, pp. 210-214, 2012.
- [41] 岩橋宏樹, 樋口雄大, 山口弘純, 東野輝夫, "歩行者群の移動軌跡情報を用いたモバイルカメラ画像内の人物位置推定手法," 情報処理学会論文誌, Vol. 56, No. 2, pp. 470-482, 2015.
- [42] 白山晋, 大澤理恵, "不可視マーカを利用した室内環境における人物の位置追跡法に関する研究," 可視化情報学会論文集, Vol. 37, No. 2, pp. 1-10, 2017.
- [43] 小牧周平, 相河聡, 高部魁, 中村康久 "ディープラーニングを用いた無線LAN位置推定の精度予測に関する一検討," 電子情報通信学会論文誌, Vol. J101-B, No. 10, pp. 878-882, 2018.
- [44] AI白書2017, <https://www.ipa.go.jp/files/000082711.pdf>, June, 2020.
- [45] 森駿文, 菊池浩明, "歩容データのDTW距離に基づく個人識別における複数部位のフュージョン手法," コンピュータセキュリティシンポジウム2018論文集2018(2), pp. 1-8, 2018.
- [46] Kinect, <https://developer.microsoft.com/ja-jp/windows/kinect/>, June, 2020.
- [47] 星尚志, 石塚宏紀, 小野智弘, "動的遮蔽物を考慮したBLEによる屋内位置推定手法の提案," マルチメディア, 分散協調とモバイルシンポジウム2017論文集, pp. 1540-1546, 2017.
- [48] 本位田真一, 飯島正, 大須賀昭彦, "オブジェクト指向トラックエージェント技術," ソフトウェアテクノロジーシリーズ3, 共立出版, 1999.
- [49] Raspberry Pi, <http://www.raspberrypi.org/>, April, 2020.
- [50] SIG Bluetooth. Core specification, <https://www.bluetooth.com/specifications/bluetooth-core-specification>, April, 2020.
- [51] J. Beutel, and M. Krasnyanskiy, Linux bluez howto: Bluetooth proto-col stack for linux, <http://www.tik.ee.ethz.ch/jbeutel/pub/bluezhowto.pdf>, September, 2017.
- [52] I. Fette, and A. Melnikov, The websocket protocol, <https://tools.ietf.org/html/rfc6455>, April, 2020.

- [53] OpenLayers, <https://openlayers.org/>, April, 2020.
- [54] 良永早耶佳, 大坪敦, 橋本大和, 廣重法道, 鶴田直之, ” プライバシーを考慮した安価でポータブルな入退室者カウント画像処理システムの開発, ” 第 80 回全国大会講演論文集 2018 (1), pp. 523-524, 2018-03-13.
- [55] M. Nishiyama et al., “ Person Re-identification using Co-occurrence Attributes of Physical and Adhered Human Characteristics, ” 23rd International Conference of Pattern Recognition (ICPR), pp. 2086-2091, 2016.
- [56] SARC3D, <http://www.openvisor.org/sarc3d.asp>, April, 2020.
- [57] Y. Deng, P. Luo, C. Loy, and X. Tang, “ Pedestrian attribute recognition at far distance, ” ACM Multimedia, pp. 3-7, 2014.
- [58] 松山隆司, 和田俊和, 波部齐, 棚橋和也, ” 照明変化に頑健な背景差分, ” 電子情報通信学会論文誌 D, Vol. J84-D, No. 10, pp. 2201-2211, 2001.
- [59] 関真規人, 和田俊和, 藤原秀人, 鷲見和彦, ” 背景変化の共起性に基づく背景差分, ” 情報処理学会シンポジウム論文集, Vol. 44, SIG05 (CVIM6), pp. 54-63, 2003.
- [60] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, “ Scalable Person Re-identification: A Benchmark, ” Proceedings of the 2015 IEEE Conference on Computer Vision, pp. 1116-1124, 2015.
- [61] 山内悠嗣, 山下隆義, 藤吉弘亘, ” [サーベイ論文] 統計的学習手法による人検出, ” 電子情報通信学会技術研究報告, pp. 113-126, 2012.
- [62] P. Viola, and M. Jones, “ Rapid Object Detection Using a Boosted Cascade of Simple Features, ” Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp. 511-518, 2001.
- [63] N.Dala, and B.Triggs, “ Histograms of Oriented Gradients for Human Detection, ” 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 886-893, 2005.
- [64] 杉本涼輔, 金主賢, 中島一樹, ” 歩行による床振動特徴を用いた個人識別, ” 計測自動制御学会論文集, Vol. 54, No. 6, pp. 533-537, 2018.
- [65] 大沢英一, ” マルチエージェント環境における交渉のモデル, ” 人工知能学会誌, Vol. 10, No. 5, pp. 690-696, 1995.

参考文献

- [66] 伊藤孝行, ” マルチエージェントの自動交渉モデルとその応用, ” 情報処理,  
Vol. 55, No. 6, pp. 563-571, 2014.

## 公表論文

### 学術誌論文

1. M. Shiozuka, T. Yotsumoto, K. Takahashi, T. Kawamura, and K. Sugahara, Agent-based Tracking Method Addressing Target Recognition Errors, *IEEJ Transactions on Electronics, Information and Systems*, Vol. 140, No. 4, pp. 484-491, 2020.

### 国際会議論文

1. Masaru Shiozuka, Tappei Yotsumoto, Kenichi Takahashi, Takao Kawamura, Kazunori Sugahara, Implementation Example with Ultra-Small PCs for Human Tracking System Based on Mobile Agent Technologies, *The Eleventh International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2017)*, pp. 73-78 (2017). Barcelona, Spain
2. Masaru Shiozuka, Tappei Yotsumoto, Kenichi Takahashi, Masashi Nishiyama, Takao Kawamura, Kazunori Sugahara, Countermeasure to Human Recognition Error for Agent-based Human Tracking System, *The Twelfth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2018)*, pp. 65-70 (2018). Athens, Greece
3. Masaru Shiozuka, Tappei Yotsumoto, Kenichi Takahashi, Takao Kawamura, Kazunori Sugahara, Calculation of Location Probabilities for Agent-based Target Tracking System, *The Sixteenth International Conference on Networking and Services (ICNS 2020)*, Lisbon, Portugal (Accepted)

## 学位論文の章と主論文の対応

該当章	題目	学術誌名, 国際会議名
第3章	Implementation Example with Ultra-Small PCs for Human Tracking System Based on Mobile Agent Technologies	UBICOMM 2017
第4章	Agent-based Tracking Method Addressing Target Recognition Errors	電気学会論文誌
第5章	Calculation of Location Probabilities for Agent-based Target Tracking System	ICNS 2020