社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2006－CVIM－154（13）
2006／5／18

全方位画像による例外行動の検出

青木　康洋†　岩井　儀雄†　谷内田正彦†

† 大阪大学大学院基礎工学研究科

あらまし　本研究では，入力用の全方位視覚センサを室内のほぼ中央に設置し，人物の入室から退室までの行動を追跡し，人物行動を認識して例外行動かどうかを検出するシステムを提案する．提案システムでは，追跡によって得られた人物の重心位置の時間変化を，その人物の行動軌跡として学習する．学習する行動モデルは部分的な軌跡からなる部分行動モデルの遷移によって表す．人物行動の認識は MCMC 法という確率的な手法により行い，最大事後確率を持つ行動モデルを探索する．すべての行動モデルの事後確率が，指定されたしきい値以下のときを例外発生とすることにより，システムが学習していない行動を例外行動として検出することができる．

Detection of Suspicious Actions Using Omnidirectional Images

Yasuhiro AOKI†, Yoshio IWAI†, and Masahiko YACHIDA†

† Graduate School of Engineering Science, Osaka University

Abstract　Our purpose is to detect suspicious actions from an image sequence. In this paper, we propose a system for tracking people and recognizing their actions through indoor scenes captured with an omnidirectional image sensor. The system can track people in real-time and extract their trajectories of movement. By using extracted trajectories, their actions are recognized as whether suspicious actions or not. Actions are modeled by transitions of action primitives, parts of extracted trajectories, and recognized by a stochastic approach using probabilities of the transitions.

1. はじめに

近年の不穏な社会情勢を反映して，監視カメラなどによる安心で安全な空間を提供するセキュリティシステムの開発が求められている．しかしカメラの映像を人間が 24 時間監視し続けることは，モニタの前に 24 時間いなければいけないことや，目視という単純作業による疲労や怠惰などから起こる人為的なミスといったことなど，コストや効率の面で問題がある．そこで人物の行動を認識し，未知の行動を行う人物を不審人物として自動的に検出するような監視システムが望まれる．このようなシステムによって，監視者が 24 時間モニタの前で監視し続けなくてもシステムが例外行動を検出したと通知したときのみに注意を払えば良いため，監視者の負担の軽減が期待できる．

例外行動を検出するためには人物の検出と追跡を行う追跡モジュールと行動の学習と認識を行う認識モジュールが必要である．人物追跡としては，背景モデルやカラーモデルを用いて人物検出を行い，線形ガウス型状態空間を仮定してカルマンフィルタで追跡を行う手法 [1] や，背景差分で時系列に検出を繰り返し，前後のフレームでの領域の重なりを利用して高速に追跡を行う手法 [2] が提案されている．また行動認識を行う手法についても様々な研究が行われているが，追跡結果から認識を行う手法は大きく二つに分類できる．一つは，行動をある離散的な状態に変換し，その遷移を調べることによって行動を認識する手法 [3] で，もう一つは，連続的な動きをそのまま使い，その動きを認識する手法 [4] である．前者のように，行動を離散的な状態で表現する手法としては，ジェスチャ認識や音声認識に良く用いられる隠れマルコ

フモデル (HMM) [3], [5] や，有限状態オートマトンを用いた手法 [6] が提案されている．一方，後者の連続的な動きを認識するために，近年，ベイズ推定を用いた確率的な手法 [7]～[9] が提案されている．ベイズ推定は周辺分布を求める計算が非常に困難であるが，Markov Chain Monte Carlo 法 (MCMC 法) [10] により周辺分布を近似計算することで解決できる．MCMC 法は HMM と比較して各状態のパラメータが明確であるという利点がある．そこで MCMC 法を用いて全方位視覚センサ [11] によって部屋の中での人物の行動を監視し，行動を認識する人物行動認識システム [4], [12] が提案されている．このシステムは，人物が入室してから退室するまでを監視して追跡を行い，足元位置による歩行軌跡をモデル化する．このモデル化によって得られる行動モデルは部分的な軌跡 (部分行動モデル) の遷移として表現されている．そして MCMC 法を用いて，人物が入室してから退室するまで，どの行動モデルに沿って行動を行っていたかを認識する．しかしこのシステムは足元位置を特徴量として用いるため，例えば，歩行とほふく前進では全く別の行動であるが，同じ経路をたどると，たどった足元位置は同じ軌跡となり，同じ行動とみなされてしまう．このように，従来システムでは行動中の姿勢の変化に対応できないという欠点がある．

そこで本研究では，人物の追跡に足元位置ではなく人物領域の重心位置を用いる．重心位置は人物領域の形状によって変わるため人物の移動軌跡と姿勢を同時に観測可能となる．また認識には部分行動の遷移で表現される行動モデルを用いる．そして入力軌跡が得られたときの各行動モデルが生起する確率 (事後確率に相当する) を MCMC 法により求め，最大の事後確率を持つ行動モデルをその入力軌跡の認識結果とする．しかし，このままでは例外行動の場合でも常になんらかの行動モデルとして認識してしまう．そこで，例外行動を検出するために予め事後確率に対してしきい値を設定する．そして，行動モデルの事後確率がこのしきい値を下回る場合，入力軌跡がそのモデルである可能性が無いとする．全ての行動モデルの可能性が無くなったときに例外行動として検出する．

本システムの概要を図 1 に示す．本システムは，全方位視覚センサからの入力画像より人物の検出と追跡を行う人物追跡モジュールと，人物の行動パターンを学習し，その行動を認識する認識モジュールで構成される．

2. 追跡モジュール

2.1 人物の検出

追跡モジュールには小阪らの手法 [12] を利用する．このシステムは，人物検出および追跡をオンラインで行うことができる．また，入力画像に全方位視覚センサからの画像を用いているため，広視野で人物の追跡が行えるので，人物の一連の行動を追跡するのに適している．図 2 に全方位視覚センサを用いて撮影した画像を示す．本研究では，システムの利用環境として，照明が比較的安定している室内を想定し，人物のその室内での一連の行動追跡を行う．まず，人物領域検出の前処理として，予め背景画像を取得しておく．そして，得られた背景画像と全方位視覚センサからの入力画像との差分をとり，動的しきい値法で 2 値化処理を行うことで，人物候補領域の抽出を行う．次に，人物の方位を特定するために，抽出した人物候補領域に対して方位方向の加算投影処理を行う．これによって，全方位視覚センサに対する方位角ごとの差分頻度ヒストグラムが得られ，人物候補領域が画像上のどの方位にあるのかが分かる．このヒストグラムに対し，ある一定のしきい値 T_θ 以上の差分頻度を持つ領域を人物の方位領域とする．さらに，得られた領域に対し，距離 (俯角) 方向にヒストグラムをとり，同様のしきい値処理を行うことで，人物領域を特定する．全方位画像の方位方向と距離方向は図 2 に示す通りである．

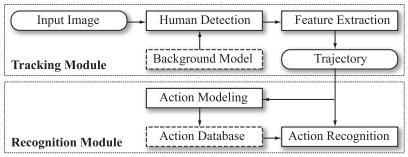


図 1　システム概要

2.2 人物の追跡

追跡処理は，人物の動きがフレーム間で連続であることを仮定して，前フレームと現在フレームの人物領域の重なりを利用して行う．

また，この追跡と同時に各人物領域での特徴量を算出し，この特徴量の移動軌跡を人物軌跡として出力する．従来のシステム [12] では，特徴量に人物の足元位置を用いていたが，これでは人物の姿勢の変化を表現できないという問題があった．例えば，歩行者でもほふく前進で進む者であっても同じ経路を進むと足元位置の軌跡は同一のものとなるのである．しかし，本研究の目的は例外行動の検出にあるため，これらは全く別の行動として認識されなければならない．そこで，本研究では特徴量として，人物の移動軌跡と姿勢を同時に表現できる人物領域の重心位置を用いる．これにより，同じ経路をたどっても姿勢が異なれば別の行動として扱うことが可能となる．

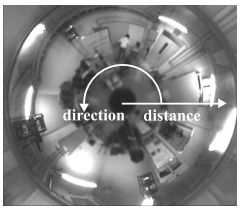


図 2　全方位画像

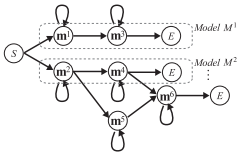


図 3　行動モデル

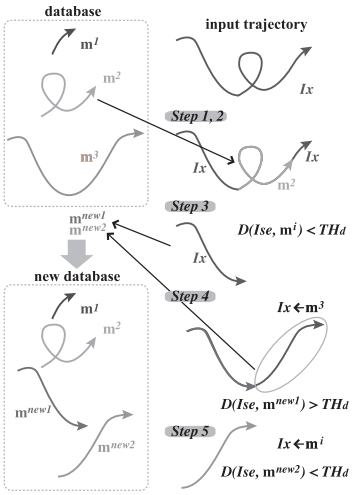


図 4　部分行動モデル自動生成アルゴリズム

3. 認識モジュール

3.1 人物の行動モデル

認識モジュールは追跡モジュールによって得られた人物の特徴量軌跡を用いて行動モデルを生成する．行動モデルの生成には松村らの手法 [4] を利用する．人物の軌跡には部分的に同じ軌跡を含む場合が多いため，軌跡をいくつかの部分的な軌跡に分割してデータベースに保存することでそのデータ量を削減できる．ここで，追跡した人物の一連の行動を行動モデル $M = \{M^{\eta} | \eta = 1, \ldots, \mathcal{M}\}$，分割された部分的な軌跡を部分行動モデル $\mathbf{m} = \{\mathbf{m}^{\mu} | \mu = 1, \ldots, m_N\}$ とする．ただし，η，μ はそれぞれ行動モデル，部分行動モデルの番号を表し，$\mathcal{M}$，m_N はそのモデルの数を表す．図 3 のように，行動モデルは部分行動モデルの遷移によって表現する．ここで，人物が行動を始めた (追跡開始) 状態を S，行動を終えた (追跡終了) 状態を E とする．また，部分行動モデル $\mathbf{m}^{\mu}$ は，τ^{μ}_{max} フレームに渡る人物特徴量の連続した点列を持つとする．なお，$\mathbf{m}^{\mu}$ のフレーム τ での特徴量として，重心位置 $(X^{\mu}_{\tau}, Y^{\mu}_{\tau})$ を用いる．さらに，モデル生成と同時に後の認識処理に必要な部分行動モデル間の遷移確率なども学習しておく．

3.2 部分行動モデルの自動生成

入力された人物の特徴量軌跡から，行動モデルと部分行動モデルを自動的に生成するアルゴリズムについて述べる．自動生成アルゴリズムは新宅の手法 [13] を用いている．s 番目から e 番目までの人物の特徴点軌跡を I_{se}，特徴量軌跡 I_{se} のフレーム t での位置を (X_t, Y_t) とし，部分行動モデル $\mathbf{m}^{\mu}$ のフレーム τ での位置を $(X^{\mu}_{\tau}, Y^{\mu}_{\tau})$ として，特徴量軌跡 I_{se} と部分行動モデル $\mathbf{m}^{\mu}$ の類似度 $D_{se}(I_{se}, \mathbf{m}^{\mu})$ を式 (1) で定義する．

$$D_{se}(I_{se}, \mathbf{m}^{\mu}) = \sum_{\tau=0}^{\tau^{\mu}_{max}} \exp\left\{-\frac{1}{2}\left(\frac{(X_{t-s+\tau} - X^{\mu}_{\tau})^2}{\sigma_X^2} + \frac{(Y_{t-s+\tau} - Y^{\mu}_{\tau})^2}{\sigma_Y^2}\right)\right\} \quad (1)$$

部分行動モデルの自動生成アルゴリズムは上で定義した $D_{se}(I_{se}, \mathbf{m}^{\mu})$ を用いて以下の 5 つのステップから構成される．この様子を図 4 に示す．

Step 1　人物軌跡のある区間の軌跡 I_{se} と，部分行動モデル $\mathbf{m}^{\mu}$ との類似度 $D_{se}(I_{se}, \mathbf{m}^{\mu})$ を測り，最大類似度を持つ I'_{se} と $\mathbf{m}^{i}$ を求める．

Step 2　類似度 $D_{se}(I'_{se}, \mathbf{m}^{i})$ がしきい値 TH_d 以上であれば，I'_{se} を $\mathbf{m}^{i}$ で表し，人物軌跡の残りの区間に対して Step1 を繰り返す．

Step 3　類似度 $D_{se}(I'_{se}, \mathbf{m}^{i})$ がしきい値 TH_d 以下であれば，I_{se} を新たな部分行動モデル $\mathbf{m}^{new1}$ として，データベースに登録する．

Step 4　新しく生成された部分行動モデル $\mathbf{m}^{new1}$ が，データベースに保存されている部分行動モデルに含まれていないか探索を行う．部分行動モデル $\mathbf{m}^{i}$ と新しいモデル $\mathbf{m}^{new1}$ の類似度が，しきい値 THd 以上の場合，$\mathbf{m}^{i}$ を $\mathbf{m}^{new1}$ で分割する．

Step 5　分割された部分行動モデル $\mathbf{m}^{i}$ の残りの部分を新しい部分行動モデルとして，Step4 を繰り返す．

以上の操作を新たな部分行動モデルが生成されなくなるまで繰り返す．

また，部分行動モデルの遷移確率 $p(\mathbf{m}^{i}|\mathbf{m}^{j})$ と，各行動モデルに対する部分行動モデルの初期生起確率 $p(\mathbf{m}^{j}|M_{t=0})$，各部分行動モデルに対する行動モデルの生起確率 $p(M_t|\mathbf{m}^{i})$ をモデルの生成と同時に学習する．それぞれの意味は 3.3 節で説明する．

3.3 確率モデルによる行動認識

3.3.1 確率モデル

学習した行動モデルを用いて，人物の行動を認識する手法を説明する．人物の行動のように非線形な動きを認識する際には，確率的な認識モデルが有効である．確率モデルには松村らのモデル [4] を用いる．このモデルは観測値を用いて行動モデルの状態を推定する．時刻 t での行動モデルを M_t，行動の状態を $\mathbf{x}_t$，観測値を $\mathbf{z}_t$ とする．ただし，状態 $\mathbf{x}_t$ の遷移にはマルコフ性を仮定する．

行動モデルの認識は，観測値に対して最大事後確率を持つ行動モデルを最適な行動モデル M_s として認識する．

$$M_s = \arg\max_{M_t} P(M_t|\mathbf{z}_t) \tag{2}$$

ここで，行動の状態 $\mathbf{x}_t$ を用いると，行動モデルの事後確率 $P(M_t|\mathbf{z}_t)$ は以下のようになる．

$$P(M_t|\mathbf{z}_t) \propto \int p(\mathbf{z}_t|\mathbf{x}_t, M_t)p(\mathbf{x}_t|M_t)d\mathbf{x}_t \tag{3}$$

部分行動モデルを用いると，状態 $\mathbf{x}$ の行動モデル M_t での条件付確率 $p(\mathbf{x}_t|M_t)$ は以下のようになる．

$$p(\mathbf{x}_t|M_t) = \int p(\mathbf{x}_t|\mathbf{m}^i)p(\mathbf{m}^i|M_t)d\mathbf{m}^i \tag{4}$$

さらに，部分行動モデルの遷移確率 $p(\mathbf{m}^i|\mathbf{m}^j)$ を用いると，部分行動モデル $\mathbf{m}^i$ の行動モデル M_t での生起確率 $p(\mathbf{m}^i|M_t)$ は

$$p(\mathbf{m}^i|M_t) \propto P(M_t|\mathbf{m}^i) \int p(\mathbf{m}^i|\mathbf{m}^j)p(\mathbf{m}^j|M_{t-1})d\mathbf{m}^j \tag{5}$$

となる． 3.2 節で学習された，予め学習した，部分行動モデルの遷移確率 $p(\mathbf{m}^i|\mathbf{m}^j)$，部分行動モデルの初期生起確率 $p(\mathbf{m}^j|M_{t=0})$，行動モデルの生起確率 $P(M_t|\mathbf{m}^i)$ を用いて，式 (5) は再帰的に求めることができる．また，状態の生起確率 $p(\mathbf{x}_t|\mathbf{m}^i)$ は一様分布とする．

式 (3) の確率を求める確率分布の計算は，分布の形状が未知であるため非常に困難である．そこで，Markov Chain Monte Carlo 法 (MCMC 法) [10] を利用して分布を近似的に計算する．状態 $\mathbf{x}_t \sim p(\mathbf{x}_t|M_t)$ で独立にサンプルできると仮定すると，サンプル $\{\mathbf{s}_t^{(n)}|n = 1, \ldots, N\}$ を用いて，式 (3) を以下のように近似できる．

$$P(M_t|\mathbf{z}_t) \propto \sum_{n=1}^{N} p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}_t^{(n)}, M_t)p(\mathbf{x}_t = \mathbf{s}_t^{(n)}|M_t) \tag{6}$$

3.3.2 サンプリング

サンプル $\{\mathbf{s}_t^{(n)}|n = 1, \ldots, N\}$ のサンプリング手法を説明する．まず，状態 $\mathbf{x}_t$ は以下のような 4 つのパラメータを持つとする [4]．

- 行動モデルの番号 η $(\eta = 1, \ldots, \mathcal{M})$
- 部分行動モデルの番号 μ $(\mu = 1, \ldots, m_N)$
- 部分行動モデル内のフレーム $\tau \in [0, \tau^{\mu}_{max}]$
- 時間方向のスケール $\kappa \in [\kappa_{min}, \kappa_{max}]$

また，状態 $\mathbf{x}_t$ が示す位置を以下のように定める．

$$\begin{pmatrix} X_{\mathbf{x}_t} \\ Y_{\mathbf{x}_t} \end{pmatrix} = \begin{pmatrix} X^{\mu}_{\tau} \\ Y^{\mu}_{\tau} \end{pmatrix} \tag{7}$$

それぞれのサンプルは重み付けられ，その重み $\pi_t^{(n)}$ は以下のように算出する．

$$\pi_t^{(n)} = p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}_t^{(n)}, M_t) \propto \exp\left\{-\frac{1}{2(D+1)} \sum_{l=-D/2}^{D/2} \left(\frac{(z^X_{t+l} - X^{\mu}_{\tau+\kappa l})^2}{{\sigma^{\mu}_X}^2} + \frac{(z^Y_{t+l} - Y^{\mu}_{\tau+\kappa l})^2}{{\sigma^{\mu}_Y}^2} \right)\right\} \tag{8}$$

ここで，${\sigma^{\mu}_X}^2$，${\sigma^{\mu}_Y}^2$ は，それぞれ部分行動モデル $\mathbf{m}^{\mu}$ に対する位置 X と Y の観測値との差のスケーリングパラメータであり，D は利用するフレーム数を表す．

時刻 t でのサンプルセット $\{(\mathbf{s}_t^{(n)}, \pi_t^{(n)})|n = 1, \ldots, N\}$ は前時刻 $t-1$ でのサンプルセット $\{(\mathbf{s}_{t-1}^{(n)}, \pi_{t-1}^{(n)})|n = 1, \ldots, N\}$ より求める．最初に，前時刻 $t-1$ のサンプルセットから，新しいサンプルセット $\{\mathbf{s}_*^{(n)}\}$ を選ぶ．この方法については，3.3.3 節で述べる．次に，選ばれた各サンプル $\mathbf{s}_*^{(n)}$ を更新する．更新の際，行動モデルの番号は，一連の行動を学習しモデルを作っているため，途中で変化することは無いと仮定する．また，部分行動モデルの番号の選択は，式 (5) で算出した部分行動モデルの生起確率 $p(\mathbf{m}^i|M_t)$ によって行い，残りのパラメータであるフレーム τ と時間方向のスケール κ は以下のように更新する．

$$\tau_t^{(n)} = \tau_*^{(n)} + \kappa_*^{(n)} + \varepsilon_{\kappa} \tag{9}$$

$$\kappa_t^{(n)} = \kappa_*^{(n)} + \varepsilon_{\kappa} \tag{10}$$

ここで，ε_{κ} は平均 0，分散 1 の白色ノイズを示す．パラメータが更新されれば，重み $\pi_t^{(n)}$ を式 (8) により求める．

部分行動モデルの生起確率 $p(\mathbf{m}^i|M_t)$ について，松村ら [4] は，部分行動の遷移確率 $p(\mathbf{m}^i|\mathbf{m}^j)$ を

$$p'(\mathbf{m}^i|\mathbf{m}^j) = \begin{cases} p(\mathbf{m}^i|\mathbf{m}^j) \exp\left\{-\frac{(\tau^j_{max} - \tau)^2}{2\sigma^2_{\tau}}\right\} & \text{if } i \neq j \\ p(\mathbf{m}^i|\mathbf{m}^j) \left(1 - \exp\left\{-\frac{(\tau^j_{max} - \tau)^2}{2\sigma^2_{\tau}}\right\}\right) & \text{otherwise} \end{cases} \tag{11}$$

と定めているが，この場合，フレームの分散 σ_{τ} を推定する必要がある．そこで本研究では，部分行動モデルの生起確率 $p(\mathbf{m}^i|M_t)$ の計算で必要な部分行動の遷移確率 $p(\mathbf{m}^i|\mathbf{m}^j)$ をモデル $\mathbf{m}^j$ からモデル $\mathbf{m}^i$ へ移る頻度 $T(\mathbf{m}^i, \mathbf{m}^j)$ を利用して，以下のように定める．

$$p(\mathbf{m}^i|\mathbf{m}^j) = \begin{cases} \frac{T(\mathbf{m}^i, \mathbf{m}^j)}{\sum_{i=1}^{m_N} T(\mathbf{m}^i, \mathbf{m}^j)} P_T & \text{if } i \neq j \\ 1 - P_T & \text{otherwise} \end{cases} \tag{12}$$

ここで，P_T は遷移を行うかどうかを表す確率で，以下のように求める．

$$P_T = \begin{cases} 1 & \text{if } \tau + \kappa > \tau^{\mu}_{max} \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

式 (13) は，ある部分行動モデルを終えてから他の部分行動モデルへ遷移する，ということを表している．また，これらの式 (12),(13) には松村らの方法 [4] と比べ，ハイパーパラメータが不要である．

3.3.3 サンプルの選択

ここでは，サンプルセット $\{\mathbf{s}^{(n)}_{t-1}\}$ から次時刻へ残すサンプルを選択し，サンプルセット $\{\mathbf{s}^{(n)}_{*}\}$ を作る方法について述べる．サンプルセットの作成方法として，重みが高いサンプルを選択する方法 [14] が提案されている．しかし，この方法は分布が多峰性を持つ場合には，一つの局所解にサンプル点が集中してしまう恐れがある．サンプル点数を増やせば，より正確に分布を近似できるため，誤認識は避けることができるが，その分計算時間が膨大になってしまう．そこで，少ないサンプル点でも局所解に集中しない方法として，定期的に再初期化を行う方法 [14] や複数のモデル間をサンプルが移行する方法 [9], [15] が提案されている．しかし，前者は状態遷移の時系列性が崩れ，後者はモデル間の遷移を記述する必要がある，といった制約がある．

そこで本研究では，遺伝的アルゴリズム (GA) のサンプルの淘汰の手法に注目し，これらの制約を用いずにサンプル点の集中を避ける手法を提案する．本研究の確率モデルは，行動モデル番号が時刻によって変わらず，$p(\mathbf{x}_t|M_t)$ の分布からサンプリングするため，一度ある行動モデルにサンプルが集中すると，それ以降他の行動モデル番号を持つサンプルを選択することは無い．したがって，可能性のある行動モデルのサンプルは次時刻にも残るようにし，可能性がないと判断されたサンプルのみを置き換えの対象とする．置き換えの対象になったサンプルは，現時刻でのサンプルセットから復元抽出で選択されたサンプルと置き換える．あるサンプルが選択される確率は，その重み $\pi^{(n)}_t$ とそのサンプルが属する局所分布の最大値との比 [16] を利用する．これによって，局所最大値に近いサンプルと置き換わる可能性が高くなる．

以下に現在のサンプルセット $\{\mathbf{s}^{(n)}_{t-1}\}$ から次時刻のサンプルセット $\{\mathbf{s}^{(n)}_{*}\}$ を選択する手順を示す．まず，各サンプルの重みと局所最大値との比 $P^{(n)}$ を次式で計算する．

$$P^{(n)} = \frac{\pi^{(n)}_{t-1}}{f(\mathbf{s}^{(n)}_{t-1})} \quad (14)$$

ただし，$f(.)$ はそれぞれのサンプルが属する局所分布の最大値を示す．また，$P^{(n)}$ は以下のように正規化しておく．

$$P^{(n)} = \frac{P^{(n)}}{\sum_{n=1}^{N} P^{(n)}} \quad (15)$$

次に各サンプルの行動モデルの事後確率 $P(M_{t-1} = \eta^{(n)}_{t-1}|\mathbf{z}_{t-1})$ とその最大値との比 $Q^{(n)}$ を次式で計算する．

$$Q^{(n)} = \frac{P(M_{t-1} = \eta^{(n)}_{t-1}|\mathbf{z}_{t-1})}{\max_{M_{t-1}} P(M_{t-1}|\mathbf{z}_{t-1})} \quad (16)$$

サンプルセットは以下のようにして決める．

$$\mathbf{s}^{(n)}_{*} = \begin{cases} \mathbf{s}^{(n)}_{t-1} & \text{if } Q^{(n)} > Th_Q \\ \mathbf{s}^{*}_{t-1} \sim P^{(n)} & \text{otherwise} \end{cases} \quad (17)$$

ただし，Th_Q はしきい値，$\mathbf{s}^{*}_{t-1} \sim P^{(n)}$ は，分布 $P^{(n)}$ にしたがって選択されたサンプル $\mathbf{s}^{*}_{t-1}$ を示す．

3.3.4 例外行動の検出

学習された行動モデルにない行動を検出したときに，それを例外行動として検出する．式 (6) にしたがって事後確率 $P(M_t|\mathbf{z}_t)$ を計算する際に，右辺値があるしきい値 $T_{P_{M_t}}$ を下回ったときは，その行動の事後確率を 0 とする．

$$P_{M_t} = \sum_{n=1}^{N} p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}^{(n)}_t, M_t) p(\mathbf{x}_t = \mathbf{s}^{(n)}_t|M_t)$$

$$P(M_t|\mathbf{z}_t) = \begin{cases} P_{M_t} & \text{if } P_{M_t} > T_{P_{M_t}} \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

各行動モデルに対して事後確率 $P(M_t|\mathbf{z}_t)$ が算出できたら，以下のように正規化する．

$$P(M_t|\mathbf{z}_t) = \frac{P(M_t|\mathbf{z}_t)}{\sum_{i=1}^{\mathcal{M}} P(M^i_t|\mathbf{z}_t)} \quad (19)$$

ここで，すべての事後確率 $P(M_t|\mathbf{z}_t)$ が 0 の時には，対応する行動モデルが学習データ中にないことを示すので，例外行動として検出する．

4. 実　　験

4.1 実験環境

人物が部屋の中を動く様子を実際に撮影し，得られた動画像を用いて処理を行うことで本システムの評価実験を行った．人物追跡処理はオンラインで行った．撮影には SONY の 3CCD カメラ XC-009 を使用し，全方位画像を得るための双曲線ミラーは山田光学工業製のものを用いた．全方位画像を 512 × 440 画素で各ピクセル 8bit の濃淡画像で取得した．撮影した部屋の大きさは約 7m × 7m で，カメラを部屋の中央に設置した．撮影した全方位画像を図 5 に示す．また，追跡処理の高速化のために日立製画像処理ボード IP5000 を使用した．そして，これらを制御し追跡結果を保存するために HP 社製ワー

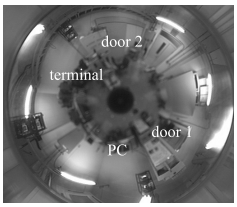
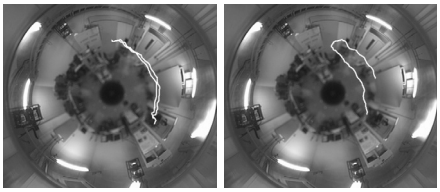


図 5　部屋の様子

表 1　人物追跡の処理時間

処理内容	処理時間 [ms/frame]
画像取得	2
差分・二値化	16
ヒストグラム取得	14
追跡・特徴量抽出	5
合計	37



(a) 部屋の往復　　(b) ほふく前進

図 6　抽 出 軌 跡

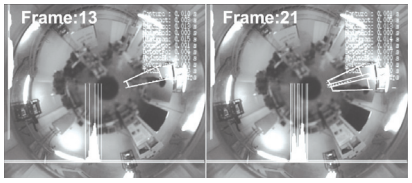


図 7　追 跡 失 敗

クステーション xw4000 を使用した．CPU は Pentium 4 2.40B GHz でメモリは 1GB，OS は Red Hat Linux 7.3 である．認識処理は，SGI 社製ワークステーションの Origin300 を用いてオフラインで行った．

4.2　追 跡 実 験

追跡モジュールにより人物を追跡する実験を行った．被験者一人を追跡したときの処理時間を表 1 に示す．また，追跡により得られた軌跡の例を図 6 に示す．表 1 では画像上の全画素に対して処理を行う必要のある，差分および二値化処理とヒストグラム取得に時間がかかっているが，全体の処理は約 27fps で行うことができた．この間隔でデータが取れれば十分に行動認識を行うことができると考えられる．図 6(a) はドア 1 から歩いて入室し，別のドア 2 の前で引き返して来たものである．図 6(b) では，経路は図 6(a) と同一であるが，ほふく前進で入室した．そしてドア 2 の前で立上り，全く同じ経路を歩いて引き返した．これらの軌跡は重心位置の軌跡であるため，経路は同一でも全く異なる軌跡となっている様子が分かる．しかし，画像は奥行き情報が欠落しているため，例えば，カメラから近いところで座ったときと遠いところに立ったときなどで，重心位置が一致することがある．狭い室内ならこれが問題になることは無いが，広い部屋で本システムを用いる場合，重心位置だけでは人物軌跡を表現できない可能性がある．

また，図 6(b) では引返す途中で軌跡がとぎれてしまっている．追跡に失敗した様子を図 7 に示す．フレーム 23 で人物の影を人物領域として検出してしまい，そこで追跡が終了してしまった様子がわかる．今後は影を検出しない方法について検討する必要がある．

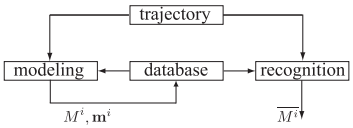


図 8　認識モジュール

表 2　認識結果のパターン

パターン 1	M と $\overline{M}$ が等しい
パターン 2	M と $\overline{M}$ が異なる
パターン 3	M が新規行動モデル

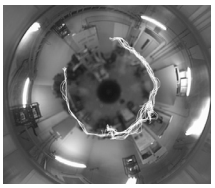


図 9　用いた人物軌跡

表 3　経　　路

経路 1	ドア 1 → ドア 2
経路 2	ドア 2 → ドア 1
経路 3	ドア 2 → PC
経路 4	PC → ドア 2
経路 5	ドア 1 → PC
経路 6	PC → ドア 1
経路 7	ドア 1 → 端末 → ドア 1

4.3　認 識 実 験

4.3.1　行 動 認 識

認識モジュールの詳細は図 8 のようになっている．行動モデルは現在のデータベースをもとにしてセルフクラスタリングにより生成されるため学習順序に依存する．3.3 節の MCMC 法による行動認識は生成されたデータベースを用いるため，行動モデルのモデリングに依存する．そこで実験に用いる軌跡集合に対して，その学習順序を乱数で適当に定めた集合を 2 つ用意し，それぞれに対して以下のようにしてテストを行った．

Step 1　集合からテスト用の軌跡を一つとる．
Step 2　残りの軌跡を先に定めた順序で学習する．
Step 3　テスト軌跡を MCMC 法で認識した結果を $\overline{M}$ とする．

表 4　集　合　1

経路 1	ドア 1 → ドア 2
経路 3	ドア 2 → PC
経路 6	PC → ドア 1

表 5　集　合　2

経路 1	ドア 1 → ドア 2
経路 4	PC → ドア 2
経路 6	PC → ドア 1

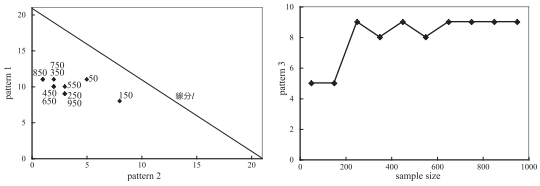
Step 4　テスト軌跡をモデリングした結果を M とする．
Step 5　$\overline{M}$ と M の関係を表 2 に示す 3 パターンに分類する．
Step 6　全ての軌跡をテストし，それぞれのパターンに分類された数を数える．
Step 7　MCMC 法のサンプル数を変えて繰り返す．

実験に用いるために取得した人物軌跡は図 9 に示したもので，経路は表 3 に示した通りである．また，各経路の軌跡はそれぞれ 3 本ずつ取得している．経路が 7 通りあるため，総軌跡数は 21 本である．実験結果を図 10, 図 11 に示す．なお，図 10(a) と図 11(a) の各点の横の数字は MCMC 法のサンプル点数を表す．表 2 のパターン 1 と 2 はそれぞれ MCMC 法による行動認識が成功した場合と失敗した場合を表す．図 10(a) と図 11(a) を見ると，サンプル点数が多くなるほどパターン 2 の MCMC 法による認識が失敗する回数が減る様子が分かる．表 2 のパターン 3 の回数は行動モデル生成のモデリング性能を表しており，回数が少ないほど性能が良いことを表す．図 10(b) と図 11(b) より，学習順序によってモデリング性能が変わることが分かる．また，モデル生成は MCMC 法とは関係が無いためサンプル数には依存せず，グラフは横ばいになっている様子も分かる．なお，サンプル点数が少ないときにパターン 3 の回数が減っているのは，MCMC 法がテスト軌跡を例外行動と誤認識したとき (パターン 2 に含まれる) を優先したためである．

次に，経路の数を減らして実験を行った．用いた経路を表 4，表 5 に示す．表 4 は経路に共通の軌跡を含まないものである．逆に，表 5 は共通の軌跡を含むものである．学習順序は乱数で適当に定めて，先と同様の実験を行った．実験結果を図 12，図 13 に示す．これらの図がほぼ同じであることから，経路の組合せには依存しないことが分かる．また，これらの図を図 10 や図 11 と比べると，モデリング，行動認識とも性能が良いことが分かる．このことから，モデリングは入力する経路の数に依存し，行動認識はサンプル数以外に，データベースの行動モデルの数にも依存すると考えられる．ある軌跡に対して，すでに同一経路の軌跡を学習済みにもかかわらず，新規行動モデルとして生成してしまうと，モデル数が増えるので，その後の行動認識の性能が下がることになる．

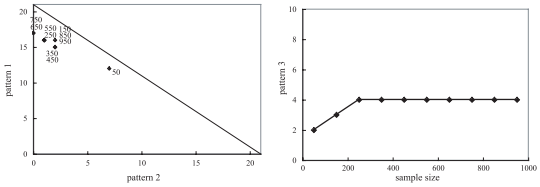
4.3.2　例 外 検 出

最後に，例外行動を検出できるかを実験した．例外行動として，図 6(a) のように，ドア 1 からドア 2 を往復したもの (例外 1) と図 6(b) のほふく前進で入室したもの (例外 2) を用いた．なお，学習には全ての軌跡を用い，学習順序は適当に定めた．また，サンプル点数は 750 点である．表 6 に検出結果を示す．この表の検出フレーム



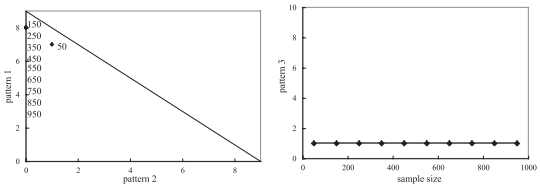
(a) パターン 1 とパターン 2　　(b) パターン 3 とサンプル数

図 10　学習順序 1



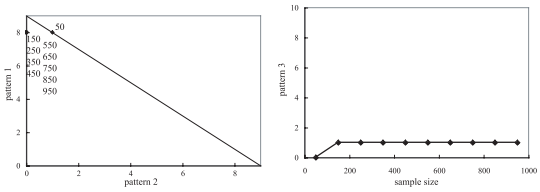
(a) パターン 1 とパターン 2　　(b) パターン 3 とサンプル数

図 11　学習順序 2



(a) パターン 1 とパターン 2　　(b) パターン 3 とサンプル数

図 12　集　合　1



(a) パターン 1 とパターン 2　　(b) パターン 3 とサンプル数

図 13　集　合　2

表 6 例 外 検 出

例外パターン	検出フレーム	理想フレーム
例外 1	153	96
例外 2	26	0

とは，軌跡を入力してから例外として検出するまでにかかったフレーム数である．また，理想フレームとは本来そのフレーム数で検出すべきフレーム数である．例外 1 は，ドアから入室し別のドアから退室する経路が 96 フレームほどであるため，このフレーム数で検出するのが理想であり，例外 2 は，ほふく前進で入室する軌跡が学習したモデルに無いため，0 フレームで検出するべきである．しかし，実際の検出には，例外 1 は約 60 フレーム，例外 2 は約 30 フレームほど余計に時間 (フレーム) が必要であった．これは，全ての行動モデルの事後確率が 0 になるまでサンプルの更新が続くためと考えられる．しかし，動画が約 30fps で取り込まれることを考えると，その間の行動は約 2 秒であるので，これで十分であると考えられる．

5. おわりに

本研究は，不審人物を自動的に検出可能な監視システムへの応用を目的として，人物の行動を認識し未知の行動を例外行動として検出するシステムを構築した．人物追跡モジュールでは人物領域の重心位置を追跡することで，人物の位置と姿勢の変化をその人物の行動軌跡とするようにした．これにより，行動中の人物の姿勢の変化を表現できるようになった．また，画像処理ボードを用いることにより，追跡処理を約 30fps で行うことができた．しかし，人物の検出が背景差分によるため，壁に映った影により，追跡が止まってしまうことがあった．周囲環境の変化に対して安定な追跡を実現する方法の検討が必要である．さらに，画像には奥行き情報がないため，重心位置だけでは不十分である可能性がある．また認識モジュールでは，実際にほふく前進で入室するような学習データにない行動を与えて，それを例外として検出できることを確認した．また，行動の認識率は MCMC 法で使うサンプル数のみならず，学習したデータベースにも依存することも確認された．これらの認識実験より，認識率を維持するためには，学習を過不足なく行う必要があることを確認した．

今後の課題は，現在はオフラインで動いている認識モジュールをオンライン化し，人物追跡から例外検出までをすべてオンラインで処理できるようにすること，行動のモデリングを学習順序に依存しないように精度を高めることである．

文　献

[1] Matsumura, A., Iwai, Y. and Yachida, M.: Tracking Algorithm for Detecting Trajectories of Human Movement from Omnidirectional Images, *Proc. of the 1st International Symposium on System & Human Science (SSR2003)*, pp. 251–256 (2003).

[2] Mituyoshi, T., Yagi, Y. and Yachida, M.: Real-time Human Feature Acquisition and Human Tracking by Omnidirectional Image Sensor, *Proc. IEEE Conf. on Multisensor Fusion and Integration for Intelligent Systems*, pp. 258–263 (2003).

[3] 青木茂樹，大西正輝，小島篤博，福永邦雄：HMM による行動パターンの認識，信学論 (D-II)， Vol. J85-D-II, No. 7, pp. 1265–1270 (2002).

[4] Matsumura, A., Iwai, Y. and Yachida, M.: Stochastic Action Recognition from Omnidirectional Images, *Proc. of Asian Conf. on Computer Vision*, Vol. 1, pp. 120–125 (2004).

[5] Oliver, N. M., Rosario, B. and Pentland, A. P.: A Bayesian Computer Vision System for Modeling Human Interactions, *IEEE Trans. PAMI*, Vol. 22, No. 8, pp. 831–843 (2000).

[6] Wada, T. and Matsuyama, T.: Multiobject Behavior Recognition by Event Driven selective Attention Method, *IEEE Trans. PAMI*, Vol. 22, No. 8, pp. 873–887 (2000).

[7] Isard, M. and Blake, A.: CONDENSATION – Conditional Density Propagation for Visual Tracking, *Int. J. of Computer Vision*, Vol. 29, No. 1, pp. 5–28 (1998).

[8] Doucet, A., Godsill, S. and Andrieu, C.: On sequential Monte carlo sampling methods for Bayesian filtering, *Statics and Computing*, Vol. 10, No. 3, pp. 197–208 (2000).

[9] Gatica-Perez, D., Lathoud, G., McCowan, I. and Odobez, J.-M.: A Mixed-State I-Particle Filter for Multi-Camera Speaker Tracking, *IEEE ICCV Workshop on Multimedia Technologies for E-Learning and Collaboration* (2003).

[10] Gilks, W. R., Richardson, S. and Spiegelhalter, D. J.: *Markov chain Monte Carlo in practice*, Chapman & Hall (1996).

[11] Yamazawa, K., Yagi, Y. and Yachida, M.: Omnidirectional Imaging with Hyperboloidal Projection, *Proc. of the Int. Conf. on Intelligent Robots and Systems(IROS-93)*, Vol. 2, pp. 1029–1034 (1993).

[12] Kosaka, Y., Hirayama, T., Iwai, Y. and Yachida, M.: People Tracking and Action Recognition System Using Omnidirectional Images, *SICE Annual Conf. 2005*, No. WP1-16-4, pp. 3584–3589 (2005).

[13] 新宅聡子：部分ジェスチャ抽出の試み，特別研究報告，大阪大学基礎工学部 (2002).

[14] Isard, M. and Blake, A.: ICONDENSATION: Unifying Low-Level and High-Level Tracking in a Stochastic Framework, *Proc. 5th European Conf. on Computer Vision*, Vol. I, pp. 893–908 (1998).

[15] Isard, M. and Blake, A.: A mixed-state CONDENSATION tracker with automatic model-switching, *Proc. 6th Int. Conf. on Computer Vision*, pp. 107–112 (1998).

[16] 姫野雅子，姫野龍太郎：多峰性問題での大域的最適解と局所最適解探索のためのニッチング法，信学論 (D-I), Vol. J85-D-I, No. 11, pp. 1015–1027 (2002).