

パーティクルフィルタによる人物行動認識と例外行動検出

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、カメラ画像上の人物を追跡し、その行動を認識することで未知の行動を行う人物を例外として検出するような、安心・安全空間を提供するための監視システムの構築を目指す。人物の行動を認識するために、連続的な状態変化である行動をいくつかの部分に分割して行動素を作り、この行動素の遷移列を行動モデルとして考える。認識は、ある観測が得られたときの行動モデルの事後確率を求めることで行う。事後確率はベイズ推定によって表され、実際の計算機上での計算はモンテカルロ法による近似計算で行える。本報告では、人物の行動として人物領域の重心軌跡を用いて認識処理を行い、未学習の行動を例外として検出できるかどうかを試みる。

Detecting Suspicious Actions Using Particle Filtering

Yasuhiro AOKI[†], Yoshio IWA[†], and Masahiko YACHIDA[†]

[†] Graduate School of Engineering Science, Osaka University

Abstract Our purpose is to detect suspicious actions from an image sequence for security and/or safety. In this paper, we propose a system for recognizing actions that are trajectories obtained by tracking people through indoor scenes captured by a camera. Actions are modeled by transitions of action primitives, or parts of obtained trajectories, and recognized by a stochastic approach using posterior distributions of the models. Monte Carlo estimation, otherwise known as Particle Filtering, is used to approximate the distributions. By this approach, the system can recognize people's actions as whether suspicious actions or not.

1. はじめに

近年、不穏な社会情勢を反映して、安心・安全空間を提供するシステムが望まれている。そのようなシステムとして、カメラによるモニタリングを利用する方法が考えられる。しかし、カメラから得た動画を人間が監視するのは、監視者は長時間モニタの前に拘束され、また、目視と言う単純作業であるため、疲労や怠惰などに起因する人為的ミスを誘発し易いなど、コストや効率の面で問題がある。そこで、動画中の人物の行動を認識し、異常行動を行う人物を例外発生として自動的に検出するようなシステムが望まれる。このようなシステムによって、長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。

例外検出手法の一つとして、クエリ画像が学習画像群と時間的・空間的にどの程度類似するかを計算して、目立った変化が表れたときを例外検出とする手法 [1], [2] がある。しかし、この手法はあくまでも変化点を検出するものであり、どのような行動を行っていたかを認識するものではない。

行動の認識を行うことで例外検出を行うためには、人物を検出しその行動を追跡するモジュールと、追跡された行動をデータベースに学習し、また認識処理を行うモジュールが必要である。検出および追跡処理としては、近年では、Boosting により検出器を学習する手法 [3] やパーティクルフィルタ [4] を利用した手法 [5]~[7] が提案されている。パーティクルフィルタは、パーティクルを

用いて内部状態を時系列に推定する手法であり、これを利用して追跡を行っている。その他、さまざまな追跡手法が提案されているが、追跡と認識のふたつのモジュールはそれぞれ独立に設計できるため、どのような追跡方法を選んでも良い。本研究ではシステムの認識処理に焦点を当てる。

認識には確率モデルを利用する方法がよく用いられており、例えば、隠れマルコフモデル (HMM) を利用した行動認識手法 [8] が提案されている。しかし、HMM を用いる場合、隠れた内部状態が何を表しているのか不明であり、また、認識の際には一連の行動の観測系列が必要であるためオンラインでの認識には向いていない、という問題がある。一方、パーティクルフィルタを行動認識に応用した手法 [9] も提案されている。この手法は追跡と認識を同時に行えるという利点があるが、行動のラベルの遷移を扱うのが難しいという問題がある。例えば、連続する時刻 $t = \{t_1, t_1 + 1\}$ で、行動のラベルが $m = \{m_1, m_1\}$ となった場合、それが行動 m_1 を 2 回繰り返したのか、それとも単に m_1 を行っている最中だったのか判断がつかない。

行動を認識する場合、先のラベルの遷移列のような遷移モデルを扱えた方がよい。従来、Matsumura ら [10] によって、パーティクルフィルタによって時系列推定を行うことで、遷移モデルで表現された行動を認識する手法が提案されている。この従来手法は、まず、特徴量の軌跡で表現された行動をいくつかの部分的な軌跡に分割する。そして、これらを行動素と呼び、各行動素にラベルを付け、もとの行動をこれらの行動素の遷移列で表現している。これによって、連続的な変化である行動軌跡を離散的な遷移モデルとして扱うことに成功している。しかし、Matsumura らが導出した式は認識処理の実装との間に食い違いあるため正確さの面で問題がある。

そこで本研究では、Matsumura ら [10] と同様の行動素の遷移列による行動モデルを時系列推定を用いて認識する手法を定式化し、パーティクルフィルタを用いて実装する。また、内部状態の遷移によって行動の軌跡を表現するが、軌跡同士は交わったり近接したりするため、ある観測から推定される内部状態は多峰性を持つ分布になる。パーティクルフィルタの性質上、多峰性を持つ分布はうまく取り扱うことができないが、Vermaak ら [11] によってこのような多峰性問題の解決策が提案されている。その方法とは、分布が多峰性を持つのであればもとの分布を局所分布の和によって表すことで解決するものであり、Vermaak らはこれによって複数の人物を同時に追跡することに成功している。本研究ではこの方法を採用することで、行動モデルの認識における多峰性問題を

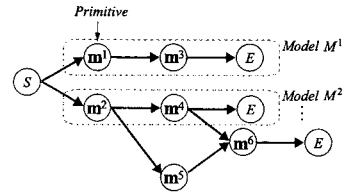


図 1 行動モデル

回避する。また、行動モデルの学習の際、全ての可能な例外を列挙することは不可能であるため例外行動そのものを学習することは非常に困難である。そこで本研究では、頻繁に観測されるような行動を通常の行動とみなして予め学習し行動モデルのデータベースを構築するものとし、データベースに無い行動が観測されたときを例外行動発生として扱う。

以下、本論文では行動モデルの表現方法と行動を表す内部状態を定義した後、まず、Matsumura らによる認識手法 [10] について説明し、続いて本研究による提案手法について述べる。次に実験とその結果について述べ、最後に結論としてまとめる。

2. 行動モデル

2.1 行動モデルと行動素

人物の行動は、例えば、部屋の移動を表現する場合は人物位置の変化であり、また、人物の姿勢を扱う場合は関節角度の連続的な変化によって表現できる。このように行動とは何らかの特徴量の連続的な変化として扱うとができるため、本研究では、行動を予め定義された特徴空間上に描かれる特徴量軌跡として表現する。

さて、経路の移動軌跡の合流地点から分岐点までのように、ある一連の行動には部分的に良く似た行動が存在することがある。これらは特徴空間上では、部分的に重なる軌跡として表現される。そこで、本研究では元の行動軌跡同士で重なる部分を共有するように軌跡を分割し、分割された部分的な軌跡を行動素と呼ぶことにする。そして人物の行動モデルを、行動素の遷移列として表現する。行動モデルの様子を図 1 に示す。ここで各 m は行動素であり、特徴空間上に軌跡 $z = m(f)$ を描く。ただし、 z は特徴量であり f は軌跡 m のパラメータである。 f を増加させ軌跡を描くことで行動素が再生されるので、本研究ではパラメータ f のことをフレームと呼ぶ。

2.2 行動の内部状態

人物の行動にはマルコフ性を持つ内部状態 x があり、現時刻 t の内部状態 x_t に従って特徴量 z_t が観測されるとする。すなわち、次式のような遷移モデルを考える。

$$x_t \sim p(x_t|x_{t-1}) \quad (1)$$

$$z_t \sim p(z_t|x_t) \quad (2)$$

本研究の場合、状態 x は行動モデルの内部状態であるので、行動素間の遷移と行動素軌跡の再生を表現できればよい。

行動素の遷移を表すためには、内部状態 x が再生中の行動素 m を含んでいければよい。状態 x_t として $x_t = (m_t, y_t)$ という形を考えると、状態遷移確率はベイズルールより

$$p(x_t|x_{t-1}) = p(y_t|m_t, x_{t-1})p(m_t|x_{t-1}) \quad (3)$$

である。これは、過去の状態から現在の行動素 m_t への遷移が $p(m_t|x_{t-1})$ で与えられ、行動素 m_t の再生が $p(y_t|m_t, x_{t-1})$ で与えられることを示している。

次に、ある行動素 m の再生について考える。現在の行動素軌跡のフレーム位置 f_t とフレームの再生速度 v_t を用いると、状態方程式

$$f_t = f_{t-1} + v_{t-1} + \mathcal{N}_f \quad (4)$$

$$v_t = v_{t-1} + \mathcal{N}_v \quad (5)$$

$$z_t = m(f_t) \quad (6)$$

によつて、行動素軌跡 $z = m(f)$ を再生できる。ただし、 $\mathcal{N}_f, \mathcal{N}_v$ はそれぞれフレーム位置 f と速度 v のノイズである。これより、ある行動素軌跡 m の内部状態 (f, v) の遷移確率は

$$p(f_t, v_t|f_{t-1}, v_{t-1}) = p(v_t|v_{t-1})p(f_t|f_{t-1}, v_{t-1}) \quad (7)$$

となる。ただし、行動素 m が遷移した場合はフレーム位置 f を初期化しなければならない。そこで、現在の行動素 m_t と一つ前の行動素 m_{t-1} を用いて、

$$\begin{aligned} p(f_t, v_t|m_t, m_{t-1}, f_{t-1}, v_{t-1}) \\ = p(v_t|v_{t-1})p(f_t|m_t, m_{t-1}, f_{t-1}, v_{t-1}) \end{aligned} \quad (8)$$

とする。

以上より、行動の内部状態として $x = (m, f, v)$ を用いると、その遷移確率は

$$p(x_t|x_{t-1}) = p(v_t|v_{t-1})p(f_t|m_t, x_{t-1})p(m_t|x_{t-1}) \quad (9)$$

とできる。

2.3 行動素の遷移確率

ここでは行動素の遷移確率 $p(m_t|x_{t-1})$ について述べる。行動モデルのデータベースには、ある行動素から別の行動素への遷移頻度が学習可能であり、したがって遷移先を決定する分布 $p(m_t|m_j)$ は既知のものとして扱え

る。しかし行動素の遷移は、ある行動素を再生し終えてから起こるものであるため、実際に遷移が起こるかどうかは現在の行動状態 x に依存する。よつて本研究では、遷移確率 $p(m_t|x_{t-1})$ を

$$\begin{aligned} p(m_t|x_{t-1}) \\ = r(x_{t-1})p(m_t|m_{t-1}) + (1 - r(x_{t-1}))\delta(m_t, m_{t-1}) \end{aligned} \quad (10)$$

と定義する。ここで $r(x_{t-1})$ は、状態 x_{t-1} が行動素の遷移を起こすかどうかを表す確率である。また、 $\delta(m_t, m_{t-1})$ は $m_t = m_{t-1}$ のときに 1 となるクロネッカーのデルタである。本定義は、右辺第一項で行動素が遷移する場合を表し、第二項では遷移せずに現在の行動素の再生を続けることを表している。

また遷移のしやすさ $r(x)$ については、ある行動素の再生が終わりそうになるほど次の行動素へと遷移しやすくなるので、状態 x のフレーム位置 f がその行動素 m の終端に近づくほど 1 に近くなるような分布にすればよい。本研究では、簡単のため $r(x) = r(m, f, v)$ を次のように定義する。

$$r(x) = \begin{cases} 1 - \varepsilon & f_m < f + v \\ \varepsilon & \text{otherwise} \end{cases} \quad (11)$$

ここで ε は小さな正の数であり、 f_m は行動素 m の終端フレームである。ただし、行動素 m に遷移先が存在しない場合は $r(x) = 0$ とする。

2.4 内部状態の尤度

観測値による尤度分布 $p(z_t|x_t)$ については、 x_t が示す特徴量 $m_t(f_t)$ との距離を利用する。まず、距離 $d(z_t, x_t)$ を次式で計算する。

$$d(z_t, x_t) = (z_t - m_t(f_t))^T \Sigma^{-1} (z_t - m_t(f_t)) \quad (12)$$

ここで、 Σ は共分散行列を表すが、本研究では、特徴量ベクトルの各要素は互いに独立であると仮定し、実装では Σ を対角行列とした。したがって距離 d は各軸に対してスケールを与えたユークリッド距離である。そして、分布 $p(z_t|x_t)$ は距離 d を用いて、

$$p(z_t|x_t) \propto \exp(-d(z_t, x_t)) \quad (13)$$

とする。

2.5 状態遷移列とモデルの事後確率

内部状態 x の時系列 $x_{0:t} = \{x_0, x_1, \dots, x_t\}$ が得られたときの行動モデル M の事後確率 $p(M|x_{0:t})$ の推定について述べる。状態 x_t は現在の行動素 m_t を含むように定義したため、状態系列 $x_{0:t}$ から行動素の遷移列

$m_{0:\tau} = \{m_0 \rightarrow m_1 \rightarrow \dots \rightarrow m_\tau\}$ を生成できる。行動モデルは行動素の遷移列で表現されているので、得られた遷移列 $m_{0:\tau}$ がモデル M に含まれているかどうかによって事後確率 $p(M|x_{0:t})$ の計算ができる。

本研究では次のような方法で事後確率 $p(M|x_{0:t})$ を求める。まず、行動素遷移列 $m_{0:\tau}$ が行動モデル M に含まれているかどうかを示す関数 $F(M, m_{0:\tau})$ を用意する。

$$F(M, m_{0:\tau}) = \begin{cases} 1 & m_{0:\tau} \subset M \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

行動モデルの事後確率 $p(M|x_{0:t})$ は、関数 F を用いて、

$$p(M|x_{0:t}) = \frac{F(M, m_{0:\tau})}{\sum_{i=1}^{N_M} F(M_i, m_{0:\tau})} \quad (15)$$

とする。ただし、 N_M はデータベース内の行動モデルの数である。

3. 従来手法と問題点

時刻 t で観測された特徴量を z_t とする。Matsumura らの手法 [10] では、行動モデル M_t の事後確率 $p(M_t|z_t)$ を計算することで行動認識を行っている。行動の内部状態を s_t として、

$$p(M_t|z_t) \propto \int p(z_t|s_t, M_t)p(s_t|M_t)ds_t \quad (16)$$

となる。 $p(s_t|M_t)$ は行動モデル M_t が起こったときの状態 s_t の生起確率で、行動素を用いて以下のように変形できる。

$$p(s_t|M_t) = \int p(s_t|m_i)p(m_i|M_t)dm_i \quad (17)$$

また、認識には状態 s_t を $s_t \sim p(s_t|M_t)$ によりサンプリングしてパーティクル $s_t^{(i)}$ を得ることににより、

$$p(M_t|z_t) \propto \sum_{n=1}^N p(z_t|s_t^{(i)}, M_t)p(s_t^{(i)}|M_t) \quad (18)$$

によって事後確率を計算するとしている。

ところで、Matsumura ら [10] はパーティクル $s_t^{(i)}$ を一時刻前のパーティクル $s_{t-1}^{(i)}$ の更新によって求めるとしている。つまり、 $s_t^{(i)} \sim p(s_t^{(i)}|s_{t-1}^{(i)})$ である。しかし、式 (16) において実際にパーティクルをサンプリングできるのは式 (17) の $p(s_t|m_i)$ である。ところが、この分布 $p(s_t|m_i)$ は一様分布と仮定されており、そもそも一時刻前の状態 s_{t-1} に依存していない。Matsumura らの論文 [10] では、認識のために導出した式と実装のためのサンプリング手法との間に矛盾が生じている。そこで本研究では、事後確率とサンプリング手法に改良を加えて、矛盾点を解決する。

4. 提案手法による認識

4.1 行動の認識

行動モデルの認識は、初期時刻 0 から現在時刻 t までの観測系列 $z_{0:t}$ が得られたときの、時刻 t での行動モデル M_t の事後確率 $p(M_t|z_{0:t})$ の分布を求めることを行う。事後確率は内部状態 x を用いることで次のように計算できる。

$$p(M_t|z_{0:t}) = \int p(M_t|x_{0:t})p(x_{0:t}|z_{0:t})dx_{0:t} \quad (19)$$

ここで、 $x_{0:t}$ は初期時刻 0 から現在時刻 t までの内部状態の系列である。また、各時刻での内部状態を観測系列のもとで条件付き独立とし、かつ、状態 x_s は未来の観測から影響を受けないとすると、

$$p(M_t|z_{0:t}) = \int p(M_t|x_{0:t}) \prod_{s=0}^t p(x_s|z_{0:s})dx_{0:t} \quad (20)$$

とできる。上式は観測系列を用いて各時刻での内部状態を推定することで内部状態の遷移列を得て、その遷移列を用いて行動モデルの事後確率を計算することを示している。

内部状態の推定 $p(x_t|z_{0:t})$ は一時刻前の分布 $p(x_{t-1}|z_{0:t-1})$ を用いて次のように逐次的に計算できる。

$$p(x_t|z_{0:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|z_{0:t-1})dx_{t-1} \quad (21)$$

$$p(x_t|z_{0:t}) = \frac{1}{p(z_t|z_{t-1})} p(z_t|x_t)p(x_t|z_{0:t-1}) \quad (22)$$

ここで、 $p(x_t|x_{t-1})$ は状態の遷移確率を、また $p(z_t|x_t)$ は尤度を表し、それぞれ 2.2、2.4 節で述べた通りである。また、 $p(z_t|z_{t-1})$ は単なる正規化定数とみなせるため、その分布の形状を知る必要は無い。計算機上での実装には、状態 x_t をパーティクル $x_t^{(i)}$ で表し、パーティクルフィルタによる近似計算を用いる。このパーティクルは分布 $p(x_t|x_{t-1})$ によってサンプリングされるため、Matsumura ら [10] の矛盾点を解決できる。

4.2 多峰性問題への対応

先述の通り、状態 x は遷移確率 $p(x_t|x_{t-1})$ によってサンプリングされるため、観測 z_t を考慮せず、状態 x_{t-1} のみによって更新される。このため、パーティクルの更新を繰り返すと、パーティクルセット内の各パーティクルの尤度 $p(z_t|x_t^{(i)})$ が偏ってしまい近似精度が悪くなる。そこで、現在のパーティクルセットからリサンプリングすることで新しくパーティクルセットを作りなおし、偏りを無くす必要がある。リサンプリングは、各パーティクルに重み付けをし、重みの大きいパーティクルほど多

く残すようにして行う。パーティクルの重みとして、観測値との尤度 $p(z_t|x_t^{(i)})$ を用いる方法 [9] が提案されているが、この場合、より尤度の高いパーティクルにパーティクル全体が集中してしまう。そのため多峰性を持つ分布では、パーティクルが局所分布に偏ってしまい近似精度が悪くなる。偏りを無くすためのリサンプリングが、かえって偏りを生む結果になってしまう。

このような多峰性問題は、例えば、複数の人物を同時に追跡する場合などに表れる。この問題に対して Vermaak ら [11] は、次のように状態の分布をいくつかの局所分布の重み付き和の形で定義する手法を提案している。

$$p(x_t|z_{0:t}) \stackrel{\text{def}}{=} \sum_{m=1}^{N_m} \pi_{m,t} p_m(x_t|z_{0:t}) \quad (23)$$

$$\sum_{m=1}^{N_m} \pi_{m,t} = 1 \quad (24)$$

ここで、 N_m は局所分布の数、 $\pi_{m,t}$ は局所分布 p_m に対する時刻 t での重みである。これを式 (21), (22) に代入することによって、次で示すように $p_m(x_{t-1}|z_{0:t-1})$ から $p_m(x_t|z_{0:t})$ 、および $\pi_{m,t-1}$ から $\pi_{m,t}$ への更新式を得る。

$$p_m(x_t|z_{0:t-1}) = \int p(x_t|x_{t-1}) p_m(x_{t-1}|z_{0:t-1}) dx_{t-1} \quad (25)$$

$$p_m(x_t|z_{0:t}) = \frac{p(z_t|x_t) p_m(x_t|z_{0:t-1})}{p_m(z_t|z_{0:t-1})} \quad (26)$$

$$\pi_{m,t} = \frac{\pi_{m,t-1} p_m(z_t|z_{0:t-1})}{\sum_{n=1}^{N_m} \pi_{n,t-1} p_n(z_t|z_{0:t-1})} \quad (27)$$

上式を用いることにより、多峰性を持つ分布をパーティクルフィルタでうまく近似することができる。

本研究では、行動素が示す特徴空間上のある一点は、別の行動素が示す一点と近いことがあり、したがって状態の分布は多峰性を持つ。そこで、局所分布 p_m として行動素別の状態の分布とすることで多峰性問題を回避することができる。

4.3 パーティクルフィルタによる近似

時刻 t でのパーティクル $x_t^{(i)}$ の重みを $w_t^{(i)}$ とすると、式 (23) は

$$p(x_t|z_{0:t}) = \sum_{i=1}^N \pi_{m_t^{(i)},t} w_t^{(i)} \delta_{x_t^{(i)}}(x_t) \quad (28)$$

と近似できる。なお、 N はパーティクルの総数であり、 $m_t^{(i)}$ はパーティクル $x_t^{(i)}$ が示す行動素である。また、ここでの $\delta_a(\cdot)$ は a で値を持つディラックのデルタ関数である。

行動モデルの事後確率 $p(M_t|z_{0:t})$ は、式 (20) をパー

ティクルで近似して

$$p(M_t|z_{0:t}) = \sum_{i=1}^N p(M_t|x_{0:t}^{(i)}) \prod_{s=0}^t p(x_s^{(i)}|z_{0:s}) \quad (29)$$

で計算できる。ここで、 $x_{0:t}^{(i)}$ は i 番目のパーティクルが時刻 0 から時刻 t までに辿ってきた状態の遷移列のことである。一見すると、時刻 t までに発生した全てのパーティクルを記録しておく必要があるように見えるが、事後確率 $p(M_t|x_{0:t})$ は 2.5 節で述べたように、状態系列 $x_{0:t}$ が辿った行動素遷移列 $m_{0:t}$ のみを必要とするため、 i 番目のパーティクルが示す状態系列 $x_{0:t}^{(i)}$ はその行動素遷移列 $m_{0:t}^{(i)}$ のみを記録しておけば良い。

パーティクルは $x_t^{(i)} \sim p(x_t^{(i)}|x_{t-1}^{(i)})$ によってサンプリングされ、その重み $w_t^{(i)}$ は、

$$\tilde{w}_t^{(i)} = w_{t-1}^{(i)} p(z_t|x_t^{(i)}) \quad (30)$$

$$w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j \in \mathcal{I}_m} \tilde{w}_t^{(j)}} \quad (31)$$

で更新する。ここで、 \mathcal{I}_m はパーティクル $x_t^{(i)}$ が示す行動素 $m_t^{(i)}$ と同じ行動素を示すパーティクルの集合である。

4.4 例外検出手法

本研究では学習したデータベースに無い行動はすべて例外行動として扱う。さて、状態遷移 $p(x_t|x_{t-1})$ による状態の更新は、学習した行動モデルのデータベースを用いて、次時刻での状態を予測していると考えられる。このとき、尤度 $p(z_t|x_t)$ は予測の正しさを与えており、あるパーティクルの尤度 $p(z_t|x_t^{(i)})$ が小さいときは、そのパーティクルによる予測が外れていることを示している。もし、全てのパーティクルの尤度が十分に小さいときはデータベースに無い例外行動を観測した可能性がある。そこで本研究では、あるパーティクルの尤度 $p(z_t|x_t^{(i)})$ が予め指定したしきい値 ε_L を下回ったときに、そのパーティクルの重み $w_t^{(i)}$ を 0 とし、全てのパーティクルの重みが 0 となったときに例外行動を検出したとする。また、例外行動が検出された場合はパーティクルセットを再び初期化することで認識処理を続けることができる。

4.5 サンプリングの問題と重みの改良

パーティクルは $p(x_t|x_{t-1})$ によって更新されるとしたが、行動モデルのデータベース D 内に

- $M_1: m_1 \rightarrow m_2 \rightarrow m_3$
- $M_2: m_4 \rightarrow m_2 \rightarrow m_5$

のように共通の行動素を含むような行動モデルが存在する場合や、次のような行動素のループを含む行動モデル

- $M_3: m_6 \rightarrow m_7 \rightarrow m_7 \rightarrow m_8$

がある場合、一時刻前のみに基づいて更新すると問題が生じることがある。例えば、最初の共通の行動素がある

例の場合、現在の行動素を m_2 とすると、遷移先は m_3 と m_5 が考えられるため、場合によってはあるパーティクルの系列 $x_{0:t}^{(i)}$ が示す行動素遷移列が $m_1 \rightarrow m_2 \rightarrow m_5$ のようにデータベースに無い遷移列を辿り得る。次のループを含む場合の例でも同様に、パーティクルの系列 $x_{0:t}^{(i)}$ が示す遷移列が $m_6 \rightarrow m_7 \rightarrow m_8$ のように、やはりデータベース内に存在しない遷移列を辿り得る。

これは、学習済みの行動素の遷移パターンから新しい行動を類推できる、と言う意味では利点となり得る。しかし、本研究のように、有限のパーティクルを用いて観測系列と行動モデルデータベース D とを比較する場合は、データベースに存在しない遷移列を辿ったパーティクル $x_t^{(i)}$ は、その系列 $x_{0:t}^{(i)}$ による事後確率が

$$\forall M \in D, p(M|x_{0:t}^{(i)}) = 0 \quad (32)$$

であるため、行動モデルの認識には全く役に立たない。さらに、観測系列 $z_{0:t}$ がデータベースに学習済みの行動モデルと同等の軌跡であった場合、先のデータベースに無い遷移列を辿ったパーティクルはいずれ尤度 $p(z_t|x_t^{(i)})$ が小さな値になり得る。しかし、行動素が遷移した直後は行動の分岐点にあたるため尤度は低くならない。このため、行動素が遷移した直後にリサンプリングを行った場合で、かつデータベースに無い遷移を辿る方が尤度分布が高くなってしまふ場合や、そもそもパーティクルの総数が少ない場合などで、全てのパーティクルがこのデータベースに無い遷移列を辿るものに偏ってしまう恐れがある。しかしながら、遷移してから暫く経つと予測が外れ出すため尤度が小さくなり、そのうちしきい値を下回り誤って例外として検出してしまふ。

この問題は、パーティクルの数を増やせば解決できるが、場合によっては大量のパーティクルが必要となるため実用的な解決法は無い。そこで本研究ではパーティクルの重みを計算する際に、式 (30) を次式で置き換えることによってこの問題を回避する。

$$\tilde{w}_t^{(i)} = w_{t-1}^{(i)} p(z_t|x_t^{(i)}) T(x_{0:t}^{(i)}) \quad (33)$$

ここで、 $T(x_{0:t}^{(i)})$ はパーティクルの辿った行動素遷移列がデータベース D 内に存在するかどうかを示すブール値であり、論理和

$$T(x_{0:t}^{(i)}) = \bigcup_{M \in D} F(M, m_{0:\tau}^{(i)}) \quad (34)$$

で計算する。なお、 $m_{0:\tau}^{(i)}$ は i 番目のパーティクルが辿った行動素遷移列であり、 $F(\cdot)$ は式 (14) のものと同一である。これは、データベース内に存在しない遷移列を辿ったパーティクルの重みを強制的に 0 へと変更する操作である。重み 0 のパーティクルはリサンプリングによって

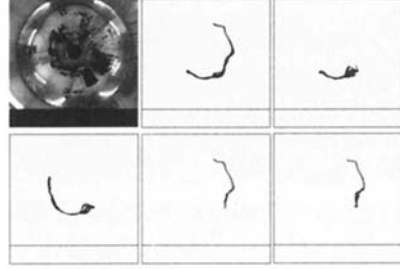


図2 全方位画像によって撮影した部屋の様子と重心軌跡

確実に他のパーティクルに置き換えられるため、データベースに無い遷移列を辿るパーティクルは存在しなくなる。ただし、この変更は現在時刻までに得られた状態を利用して尤度を変更しているため、理論的な面での正確さに関して疑問が残る。この点を踏まえた理論の改良は今後の課題とする。

5. 実験

5.1 実験環境

全方位視覚センサ [12] を用いて人物が部屋の中を動く様子を撮影し、得られた動画から人物の重心軌跡を抽出した。部屋の大きさは $7\text{m} \times 7\text{m}$ で、カメラを部屋の中央に設置した。撮影した全方位画像は 512×440 ピクセルで、8ビットのグレイスケール画像である。部屋の様子の全方位画像と人物の追跡によって得られた軌跡を図2に示す。各軌跡は5つの経路を通った軌跡であり、各経路10本、合計50本ある。なお、各軌跡は10～15秒ほどの長さである。また、実験に用いた計算機は、PowerPC G5 2.0GHz (デュアル)、メモリ 1.5GB の Mac OS X 10.4 である。なお、以下で比較する従来手法とは、Matsumura ら [10] による手法のことであり、また、行動の学習には Matsumura らが用いた手法を利用した。

5.2 検出性能

近似精度や例外の検出にはしきい値やパーティクル数の影響を受けると考えられる。そこで、これらのパラメータを変えることで検出率がどのように変わるかを調べた。ここで調べた項目は

FAR false accept ratio,

例外 (未学習) 行動を学習済みと判断する割合。

FRR false reject ratio,

学習済みの行動を例外として検出する割合。

の2種類である。実験方法は以下に示す通りである。

まず、全方位画像から得られた50本の重心軌跡を図2のように各経路ごとに手作業でクラスタリングしておく。次に、ある経路以外の軌跡を学習してデータベースを構

築する。そして、学習しなかった経路の軌跡を認識させ、例外として検出されるかどうかを試みる。また、学習した経路の軌跡についても認識処理を行い、学習済みと判断されるかどうかを試みる。

図 3(a) にしきい値を変更したときの検出率を、また、図 3(b) にパーティクル数を変更したときの検出率を示す。両図とも左側が FAR、右側が FRR であり、実線が提案手法による結果で、点線が従来手法による結果である。図 3(a) より、しきい値が大きいたまはほとんどの行動を例外として検出するために FRR が高くなり、FAR が低くなっている様子が分かる。また、しきい値を十分小さくするとそれ以上変化はない様子も見て取れる。さらに、提案手法の方が従来手法より良い結果を示している。なお、図中ではグラフは横ばいになってはいるが、しきい値を非常に小さくすると、それは例外行動検出を行わないという意味になるので、あまり小さくしすぎないほうが良いと思われる。次に図 3(b) では、パーティクル数が少ないと分布の近似精度が悪くなるために、FRR が高く、FAR が低くなっている様子が分かる。ただし、近似精度の向上のためにパーティクル数を増やすと、認識アルゴリズムの計算コストがパーティクル数に比例するため、実行速度とのトレードオフになる。

さて、両図から FAR、FRR ともにある程度以上は改善されていない様子が分かる。まず、FAR に関しては次の理由が考えられる。学習した行動について、図 2 をみると、図の中央下の行動は中央上の行動に含まれていることが分かる。また、本研究での例外検出手法は内部状態の観測特徴量に対する尤度のみを利用している。このため、図 2 の中央下の行動を学習していなくても中央上の行動を学習している場合は、中央下の軌跡を認識処理に入力したときに尤度が一向に下がらないため、データベースに含まれると判断される。これは本実験では false accept として扱われてしまうため、その結果 FAR が上昇する。これを回避するためには、学習済みの行動の一部分のみ行う行動を例外として扱う必要がある。そのためには行動の開始状態と終了状態を行動素の遷移に含めれば良いが、このような行動を本当に例外として扱うかについては今後の検討事項とする。

一方、FRR については、パーティクル数が足りず近似精度が悪いことが理由にあげられるが、他に、データベースの学習精度の問題があげられる。例えば本研究で用いた学習手法に図 2 の全ての軌跡を学習させると 9 つの行動モデルが生成されたが、図の経路は 5 つなので冗長で複雑なデータベースが生成されている。データベースが複雑になれば、分布の近似に多数のパーティクルが必要になるため、学習精度が原因で誤検出したと考えら

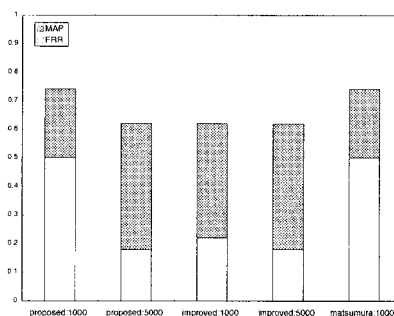


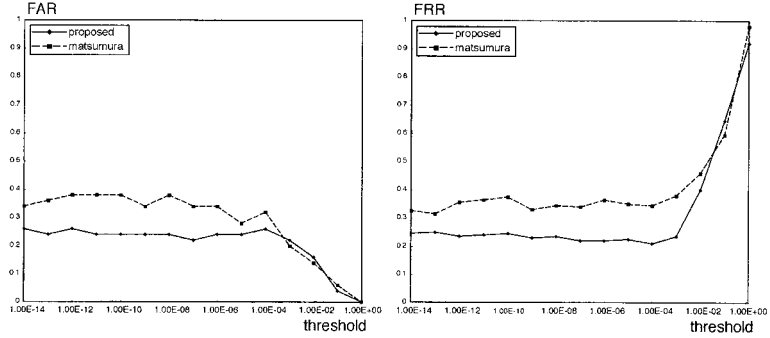
図 4 認識率の比較

れる。精度の良い学習方法については今後の課題とする。

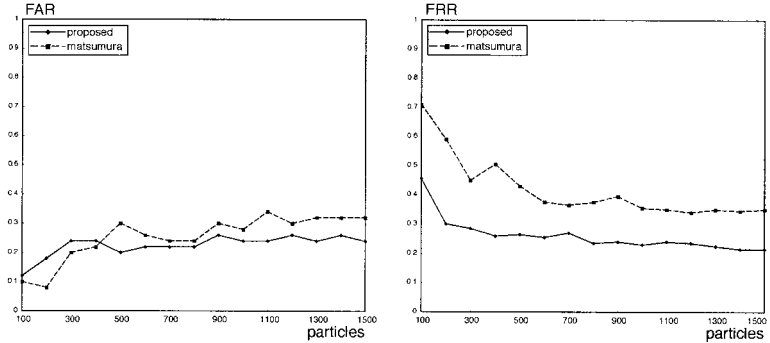
次に、認識処理によって得られた行動モデルの事後確率 $p(M|z_{0:t})$ から MAP 推定を行い、行動認識を行った。その結果を図 4 に示す。実験にはテスト用の軌跡以外を学習して、テスト用軌跡を認識するジャックナイフ法を用いた。ここで実験を行った手法は、図 4 の左から、提案手法でパーティクル数を 1000 個にしたもの、同じく 5000 個で行ったもの、続いて、提案手法を 4.5 節で述べた方法によって改良したもので、パーティクル数を 1000 個にしたものと 5000 個にしたもの、最後に従来手法でパーティクル数を 1000 個にしたものである。棒グラフの白い部分が FRR で、灰色の部分が MAP 推定で認識に成功した割合である。認識に成功したかどうかの判定は、テスト軌跡が自分と同じ経路に属する軌跡の行動モデルと同一と判断されるかどうかで行った。図 4 より、従来手法や提案手法では FRR が高いことが分かる。この原因は、4.5 節で述べた通り、直前の状態のみを利用して状態遷移を行っているために起こるものであると考えられる。FRR は図に示した通り、パーティクル数を増やすことで低く抑えることができてはいるが、実効速度を調べてみると、パーティクル数が 1000 個のときにはおよそ 32[fps] であったのに対し、5000 個のときは約 6[fps] に下がっており、認識処理の実行速度で大きな問題が生じる。そこで、認識処理に 4.5 節で述べた改良を施したところ、図 4 の通り、パーティクル数が 1000 個でも 5000 個のときの性能に近くなった。しかし、認識率自体は低いためまだ改良が必要である。

6. おわりに

本研究では、予め頻繁に観測される行動を行動モデルとして学習し行動認識を行うことで、未知の行動を観測したときに例外として検出する手法を提案した。行動モデルは行動素の遷移列によって表現し、認識手法は時系列ベイズ推定によって定式化し、パーティクルフィルタ



(a) しきい値による検出率の変化



(b) パーティクル数による検出率の変化

図3 検出率の変化

を用いて近似的に計算した。提案手法に対して実験を行った結果、未学習の行動を例外発生として検出することに成功したが、学習済みの行動も誤って例外として検出することがあった。この原因としては、行動モデルの学習精度が低いことが考えられる。なお、本研究で行った評価実験では、行動の特徴量として人物位置のみを用いたが、この場合、腕や手などの細かい動きを扱うことができないため、認識できる行動の範囲が小さいという欠点がある。今後の課題は精度の良いデータベースの構築方法の検討や人物の行動を表せるものとして最適な特徴量の検討があげられる。また、行動の種類や学習する数を増やした場合の検出率の評価も必要である。

参考文献

- [1] 南里卓也, 大津展之, “複数人物動画画像からの異常動作検出,” 信学技報 PRMU2004-77, pp.9-16, 2004.
- [2] O. Boiman, and M. Irani, “Detecting irregularities in images and in video,” *IJCV*, vol.74, no.1, pp.17-31, 2007.
- [3] P. Viola, and M. Jones, “Rapid object detection using a boosted cascade of simple features,” *CVPR*, vol.1, pp.511-518, 2001.
- [4] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking,” *IEEE*

Trans. Signal Processing, vol.50, no.2, pp.174-188, 2002.

- [5] M. Isard, and A. Blake, “CONDENSATION – conditional density propagation for visual tracking,” *IJCV*, vol.29, no.1, pp.5-28, 1998.
- [6] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, “Color-based probabilistic tracking,” *ECCV*, 2002.
- [7] K. Okuma, A. Taleghani, N. de Freitas, J.J. Little, and D.G. Lowe, “A boosted particle filter: Multitarget detection and tracking,” *ECCV*, pp.28-39, 2004.
- [8] 青木茂樹, 岩井嘉男, 大西正樹, 小島篤博, 福永邦雄, “人物の位置・姿勢に注目した行動パターンの学習・認識と非日常状態検出への応用,” 信学論 D-II, vol.87, no.5, pp.1083-1093, 2004.
- [9] M.J. Black, and A.D. Jepson, “A probabilistic framework for matching temporal trajectories: CONDENSATION-based recognition of gestures and expressions,” *ECCV*, pp.909-924, 1998.
- [10] A. Matsumura, Y. Iwai, and M. Yachida, “Tracking people and action recognition from omnidirectional images,” in *System and Human Science*, chapter 37, pp.501-514, Elsevier, 2005.
- [11] J. Vermaak, A. Doucet, and P. Pérez, “Maintaining multi-modality through mixture tracking,” *ICCV*, vol.2, pp.1110-1116, 2003.
- [12] 山澤一誠, “全方位視覚センサ hyperomni vision に関する研究-移動ロボットのナビゲーションのために-,” 博士論文, 大阪大学, 1997.