

非剛体物体の形状変化と運動変化の主成分分析に基づく パーティクルフィルタの効率化

満上 育久[†] 角所 考[†] 美濃 導彦[†]

[†] 京都大学 学術情報メディアセンター 〒606-8501 京都府京都市左京区吉田二本松町

E-mail: [†]{mitsugami,kakusho,minoh}@mm.media.kyoto-u.ac.jp

あらまし 本稿では、パーティクルフィルタによる非剛体物体のトラッキングにおいて、効率的なパーティクル生成を実現する手法を提案する。パーティクルフィルタでは、物体の運動や形状をパラメトリックに記述し、そのパラメータを逐次推定するが、各パラメータを独立に推定するために効率が悪い。そこで本稿では、オフライン段階でゆっくり正確なトラッキングを行ってその物体の形状変化と運動変化の相関関係を主成分分析で学習しておき、オンライン実行時にはその学習結果に従って適切にパーティクルを生成することでパーティクル数を削減し、高速・安定なトラッキングを実現している。水槽で泳ぐ金魚を対象として実験を行い、提案手法の有効性を確認する。

キーワード トラッキング, パーティクルフィルタ, 主成分分析, 非剛体物体

Efficient Particle Filtering for a Non-rigid Object based on PCA about Changes of Its Shape and Motion

Ikuhisa MITSUGAMI[†], Koh KAKUSHO[†], and Michihiko MINOH[†]

[†] Academic Center for Computing and Media Studies, Kyoto University Yoshida Nihonmatsu-cho,
Sakyo-ku, Kyoto, 606-8501, Japan

E-mail: [†]{mitsugami,kakusho,minoh}@mm.media.kyoto-u.ac.jp

Abstract This paper proposes a novel tracking method for non-rigid objects that is based on the particle filtering and can generate particles more efficiently. The particle filtering method prepares a model of the object's motion and shape, and estimates the parameters of the model sequentially. In this estimation, each parameter is calculated independently so that the calculation is not efficient. Considering this inefficiency, the normal particle filtering method is modified; the relation between the motion and shape is measured preliminarily by slow but precise tracking, and then generate the particles efficiently based on the relation. By this modification, the tracking works well though the number of particles are reduced. Experimental results of goldfish tracking shows the effectiveness.

Key words Object tracking, particle filter, PCA, non-rigid objects

1. はじめに

パーティクルフィルタは、非線形・非ガウス型の状態空間に対して効率よく状態を推定することができる時系列フィルタリング手法として注目され、近年、コンピュータビジョンにおける対象物体トラッキングに多く利用されている[1]~[4]。このパーティクルフィルタは、対象物体の位置・姿勢・形状などをパラメータとする状態空間を考え、その空間中での尤度分布に応じて多数の仮説（パーティクル）をランダムに発生させ、それぞれの尤度を評価し、尤度分布を更新していく手法である。そのため、この手法において、トラッキングの性能はパーティクルの数や配置に強く依存する。しかし、一般のパーティクル

フィルタでは、各パラメータを独立な変数として扱うため、状態空間中のパーティクルの配置に関する制約を設けることはできない。そのため、対象の運動変化・形状変化の特徴に合わせてパーティクルを配置する領域を広くとり、パーティクル数を増やすという方策がとられる。すなわち、状態をその位置・姿勢などの運動変化に関するパラメータのみで記述できる剛体物体の場合は、状態空間の次元が低く、比較的少ないパーティクルでトラッキング可能だが、複数の関節を持つ関節物体や自由な変形が起こる非剛体物体などの場合は、運動変化パラメータに加えて自らの形状に関するパラメータをも状態として記述しなければならなくなるため、状態空間の次元数が増加し、必要なパーティクル数も増加する。パーティクルフィルタでは、

パーティクル数と処理時間が比例するため、これはトラッキングの処理速度の低下を意味する。

そこで、本稿では、パーティクルフィルタに非剛体物体トラッキングにおいてパーティクルを効率的に配置することを目標とする。上述の通り、処理速度はパーティクル数に比例するため、この配置の効率化は処理速度の向上にも効果的であると考えられる。本稿では、その実現のために非剛体物体の形状変化と運動変化の關係に着目する。例えば、ボールが壁にあたってはねかえる（速度が大きく変化する）時には、ボールが壁に衝突する際に変形が起こる。また、魚は、加速や方向転換の際に、水からの反力を得るために体をくねらせる。これらの例のように、非剛体物体は、自らの運動に変化を生じる際に外力を受けて（受けるために）形状を変化させており、この形状変化と運動変化の相関関係・因果関係を事前に観測・学習しておき、そこで獲得した運動パターンの範囲内にパーティクルを配置することで、状態空間中での無駄なパーティクルを除去できると考えられる。この考えに基づき、本稿では、まずオフライン段階において、通常のパーティクルフィルタで十分広い領域に十分な数のパーティクルを利用して低速ではあるが正確なトラッキングを行い、その形状変化・運動変化の相関関係を主成分分析で学習しておき、オンライン処理時には、その学習結果を利用してその物体が取りうる形状変化・運動変化内のみにパーティクルを生成させていく手法を提案する。

なお、非剛体物体の形状パラメータ数が大きい場合に、形状パラメータを主成分分析して次元圧縮をする方法 [1] が提案されている。また、人体を対象として Scaled Gaussian Process Latent Variable Model (SGPLVM) を用いて形状パラメータに関する事前確率を推定する方法 [5] や、それを人物トラッキングに適用した手法 [6] などが提案されている。ただし、これらの既存研究は、いずれも対象の形状パラメータのみに着目して学習・次元圧縮を行っており、対象の運動については形状と独立な位置パラメータを設定してトラッキングを行っている。これに対して、本稿の手法は、非剛体物体の運動において現れる形状変化と運動変化の間の因果關係に着目し、その因果關係を適切にパーティクルフィルタの中で利用している点に新規性があると考えられる。

2. パーティクルフィルタとその実装

2.1 時系列フィルタリング理論

パーティクルフィルタは、時系列フィルタリング [7] を基にした手法である。ここではまず、この時系列フィルタリングの一般的な理論を簡単に説明する。

時刻 t における対象の状態ベクトルを $\mathbf{x}_t$ 、時刻 t までの状態ベクトルの履歴を $X_t = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t\}$ とする。また、時刻 t の観測ベクトルを $\mathbf{z}_t$ 、その履歴を $Z_t = \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_t\}$ とする。パーティクルフィルタでは、対象のトラッキング（状態ベクトル $\mathbf{x}_t$ の推定）を、観測系列 Z_t が与えられた時の状態ベクトル $\mathbf{x}_t$ の事後確率分布 $p(\mathbf{x}_t|Z_t)$ の推定問題として定式化している。

観測系列 Z_t から状態ベクトル $\mathbf{x}_t$ の事後確率 $p(\mathbf{x}_t|Z_t)$ を推

定するのは一般に容易ではない。これに対して、状態 $\mathbf{x}_t$ を指定した際の観測値 $\mathbf{z}_t$ の尤度は比較的容易に求められることが多い。パーティクルフィルタでは、この点に着目し、ベイズの定理を用いて、事後確率 $p(\mathbf{x}_t|Z_t)$ を次式のように変換する。

$$\begin{aligned} p(\mathbf{x}_t|Z_t) &= p(\mathbf{x}_t|\mathbf{z}_t, Z_{t-1}) \\ &= \frac{p(\mathbf{x}_t, \mathbf{z}_t, Z_{t-1})}{p(\mathbf{z}_t, Z_{t-1})} \\ &= \frac{p(\mathbf{z}_t|\mathbf{x}_t, Z_{t-1})p(\mathbf{x}_t, Z_{t-1})}{p(\mathbf{z}_t, Z_{t-1})} \\ &= \frac{p(\mathbf{z}_t|\mathbf{x}_t)p(Z_{t-1})p(\mathbf{x}_t|Z_{t-1})}{p(\mathbf{z}_t, Z_{t-1})} \\ &= \frac{p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|Z_{t-1})}{p(\mathbf{z}_t|Z_{t-1})} \end{aligned} \quad (1)$$

この式で、分母の $p(\mathbf{z}_t|Z_{t-1})$ は、各観測 $\mathbf{z}_t$ は独立であることから $p(\mathbf{z}_t)$ に等しく、また、状態 $\mathbf{x}_t$ とは無関係である。よって、実際の計算時には事後確率の正規化定数 k_t として扱う。すると、事後確率 $p(\mathbf{x}_t|Z_t)$ は、次式のように時刻 t における事前確率 $p(\mathbf{x}_t|Z_{t-1})$ と観測値の尤度 $p(\mathbf{z}_t|\mathbf{x}_t)$ の積で求められることが分かる。

$$p(\mathbf{x}_t|Z_t) = k_t p(\mathbf{z}_t|\mathbf{x}_t) p(\mathbf{x}_t|Z_{t-1}) \quad (2)$$

ここで、状態の変化が次式で示すようなマルコフ性を満たすと仮定する。

$$p(\mathbf{x}_t|X_{t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}) \quad (3)$$

すなわち、時刻 t における対象の状態 $\mathbf{x}_t$ は、直前時刻 $t-1$ の状態 $\mathbf{x}_{t-1}$ にのみ依存し、それより過去の状態履歴 X_{t-2} には無関係とする。このマルコフ性を仮定すると、式 (2) の右辺に含まれる事前確率 $p(\mathbf{x}_t|Z_{t-1})$ は次式のように計算できる。

$$\begin{aligned} p(\mathbf{x}_t|Z_{t-1}) &= \int p(\mathbf{x}_t, \mathbf{x}_{t-1}|Z_{t-1}) d\mathbf{x}_{t-1} \\ &= \int \frac{p(\mathbf{x}_t, \mathbf{x}_{t-1}, Z_{t-1})}{p(Z_{t-1})} d\mathbf{x}_{t-1} \\ &= \int \frac{p(\mathbf{x}_{t-1}, Z_{t-1})p(\mathbf{x}_t|\mathbf{x}_{t-1}, Z_{t-1})}{p(Z_{t-1})} d\mathbf{x}_{t-1} \\ &= \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|Z_{t-1}) d\mathbf{x}_{t-1} \end{aligned} \quad (4)$$

この式の右辺に含まれる $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ は、隣接 2 時刻間の状態間の遷移確率を表す。例えば、状態が $\mathbf{x}$ 微小時間で変化しないという運動モデルを持つ場合、遷移確率は、

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \exp\left\{-\frac{|\mathbf{x}_t - \mathbf{x}_{t-1}|^2}{\sigma}\right\} \quad (5)$$

などの形で表わされる。これは、 $\mathbf{x}_{t-1}$ を中心としたガウス分布であり、設定した運動モデルに従い $\mathbf{x}_t$ が $\mathbf{x}_{t-1}$ と一致する確率が最も高く、そこから離れるほど次第に確率が低くなるという確率分布である。

式 (2) と式 (4) より、事後確率 $p(\mathbf{x}_t|Z_t)$ に関する漸化式が構成されており、これを用いると $p(\mathbf{x}_t|Z_t)$ を順次算出していくことが可能となることが分かる。そして、その計算の上で重要なのが、状態の遷移確率 $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ と観測値に対する尤度 $p(\mathbf{z}_t|\mathbf{x}_t)$ の 2 つであることも分かる。

2.2 パーティクルフィルタ

事後確率 $p(\mathbf{x}_t|Z_t)$ が単峰性のガウス分布であれば、前節の処理は線形計算で行うことができる。これがカルマンフィルタ [8] である。しかし、このような単峰性を仮定することは、各時刻の状態 $\mathbf{x}_t$ を一意に定めることを意味しており、一度状態推定に誤りが生じると、以後正しい状態に復帰することができない。しかし、対象物体が複雑に移動しうるトラッキングの問題においては、一時的なトラッキングの失敗が十分に起こりうる。このような一時的な状態推定誤りによってトラッキングが破綻しないようにするためには、状態 $\mathbf{x}_t$ の候補を複数保持できるように、事後確率がより柔軟な分布をとることのできる実装が望ましいと考えられる。

パーティクルフィルタは、このような要求に応えられる時系列フィルタリングの実装方法である。事後確率の確率分布を、その分布に応じた密度で離散サンプリングされた多数のパーティクルの集合によって表現しており、単峰性などといった確率分布の形状に関する制約がつかず、柔軟な分布表現が可能となる。具体的な処理の流れを以下に示す。

- パーティクル生成

まず、前時刻の事後確率分布 $p(\mathbf{x}_{t-1}|Z_{t-1})$ に応じて i 個のパーティクル $\mathbf{s}_{t-1}^i$ を生成する。

- 予測

前節の式 (4) に示す事後確率と状態遷移確率の畳み込みに相当する処理を、各 $\mathbf{s}_{t-1}^i$ を状態遷移確率に応じて遷移させる（予測する）ことによって実現する。状態遷移確率は、前節に述べたように、設定した運動モデルに従って移動した場合に確率が最大で、そこからずれるほど確率が小さくなるように設計する。その確率分布に従い、各 $\mathbf{s}_{t-1}^i$ は、まず運動モデルに従って遷移し (drift)、さらにそこから要素ごとにガウス分布に基づくランダム値を加え (diffuse) たものを、予測パーティクル $\mathbf{s}_t^i$ とする。

- 尤度推定

前節の式 (2) に相当する尤度計算を、予測パーティクル $\mathbf{s}_t^i$ ごとに、現時刻 t における観測値 $\mathbf{z}_t$ に対する尤度 π_i を求める (measure) ことによって行う。ただし、全予測パーティクルに対する尤度の合計が 1 になるよう正規化する。

$$\pi_t^i = \frac{p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}_t^i)}{\sum_{i=1}^N p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}_t^i)} \quad (6)$$

この π_t^i の分布が、現時刻 t における事後分布 $p(\mathbf{x}_t|Z_t)$ に相当する。

以上の処理を繰り返すことによって、逐次事後分布を推定していく。なお、この処理から分かるように、生成される予測パーティクルが実際の状態変化結果から完全に外れてしまうと、事後分布の推定に失敗する。したがって、実際の状態変化に対応できる運動モデルと diffuse を設定することが重要である。たとえば、対象の状態変化が設定した運動モデルにほぼ正確に従う場合には、diffuse の分散は小さくてよいが、実際の状態変化が運動モデルから大きく外れることがある場合には、diffuse の分散を大きくとらなければならない。

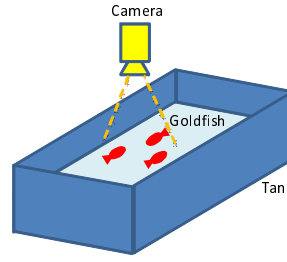


図1 撮影環境

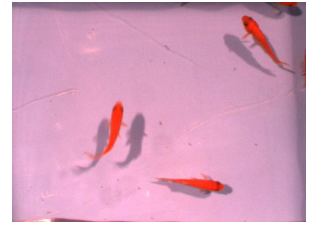


図2 撮影画像例

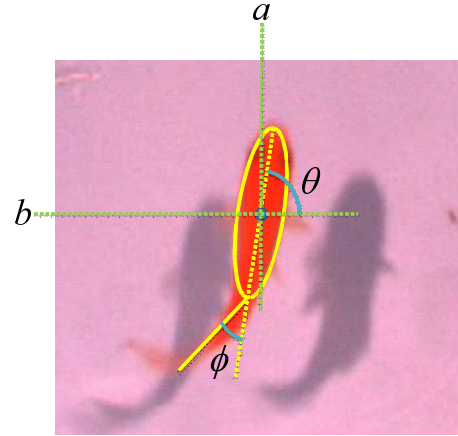


図3 金魚モデル

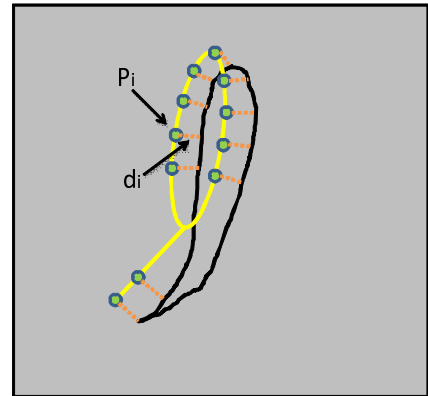


図4 尤度推定

3. 非剛体物体のトラッキングにおけるパーティクルフィルタ

3.1 金魚を対象としたパーティクルフィルタの一般的実装法

非剛体の移動物体の一つとして、本稿では、水槽を泳ぐ金魚を取り上げ、パーティクルフィルタによるトラッキングの一般的な実装例を示す。

金魚は、図1に示すように水槽の上方に設置されたカメラで撮影されており、図2のように観察される。この金魚を、図3に示すような楕円と線分によるモデルによって記述する。このモデルは、金魚の位置を表す楕円中心 (a, b) 、金魚の向きを表す楕円の長軸方向 θ 、金魚の体の曲がり具合を表す角度 ϕ という4パラメータで表現され、これを状態 $\mathbf{x}$ として使用する。

尤度計算については、金魚の周囲の境界線と上述のモデルの

一致度を用いる．具体的には，まず撮影画像から金魚色領域を抽出し，抽出結果に対して Canny オペレータを適用しエッジ画像を取得する．次に，図 4 に示すように，モデルの楕円上および線分上に複数の点 P_j を用意し，それぞれの点から画像のエッジまでの距離 d_j を算出する．各パーティクルに対する尤度 π_i は，この d_j のうち最大長のもの $d_{\max}$ を使って，以下のようを与える．

$$\pi = \exp\left\{-\frac{d_{\max}^2}{\sigma_d}\right\} \quad (7)$$

ただし， σ_d は d の分散に応じて定める定数である．

以上の準備をもとにパーティクルフィルタを実装する．状態遷移は，各パラメータが隣接 2 時刻ではほぼ変わらないことを仮定した静止モデルであるとする，生成される予測パーティクル $\mathbf{s}'_t = (a, b, \theta, \phi)^T$ は以下の形で表わされる．

$$\begin{pmatrix} a_t \\ b_t \\ \theta_t \\ \phi_t \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{t-1} \\ b_{t-1} \\ \theta_{t-1} \\ \phi_{t-1} \end{pmatrix} + \begin{pmatrix} \delta_a \\ \delta_b \\ \delta_\theta \\ \delta_\phi \end{pmatrix} \quad (8)$$

この式で，右辺の行列が drift を表しており，ここでは静止モデルを仮定しているため，単位行列となっている．また， $\boldsymbol{\delta} = (\delta_a, \delta_b, \delta_\theta, \delta_\phi)^T$ は，遷移確率に応じたパーティクルの diffuse を表す．このように，静止モデルでは，運動モデルに基づく drift がなく，ランダム値を加算する diffuse のみとなるため，このモデルをランダムウォークとも呼ぶことがある．

3.2 一般的実装法の問題点

前節に示した実装によって，金魚のトラッキングを行うことが可能だが，処理速度が非常に遅いという問題が生じる．この要因について考察する．

要因の一つは，モデルパラメータ数（状態空間の次元数）の増加である．モデルをその位置・姿勢のみで記述できる剛体物体の場合は，状態空間の次元が低く，比較的少ないパーティクルでトラッキング可能だが，複数の関節を持つ関節物体や自由な変形が起こる非剛体物体などの場合は，位置・姿勢に加えて自らの形状に関するパラメータをも状態として記述しなければならないため，状態空間の次元数が増加し，必要なパーティクル数が増加する．パーティクルフィルタでは，パーティクル数と処理時間が比例するため，このパラメータ数の増加がトラッキング処理速度の低下に直接影響する．

もう一つの要因は，状態変数のパラメータ間の相関関係を無視している点であると考えられる．時系列フィルタリングやパーティクルフィルタなどのアルゴリズムは，画像認識の分野における対象トラッキングのみに適用されるわけではないため，一般性を失わないアルゴリズムとなっている．そのため，状態変数のパラメータ間の因果関係・相関関係などは考慮されず，通常の物理現象に沿わないパーティクルを多数発生させてしまう．例えば，図 5 に示すように，前時刻 $t-1$ に体を右に曲げた金魚が現時刻 t で左に進行していることは実際の現象としてはあり得ないが，通常のパーティクルフィルタの枠組みでは，このような不自然なパーティクルを未然に排除することはでき

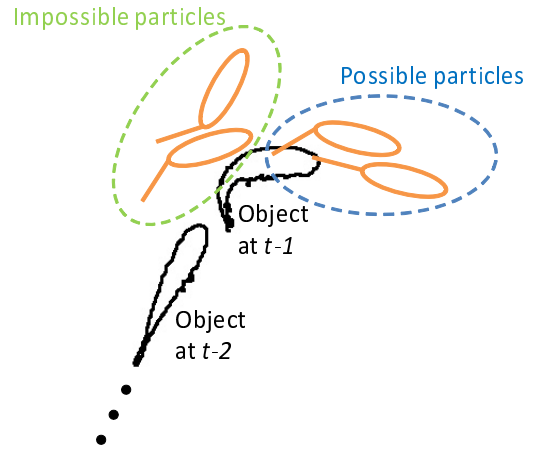


図 5 不自然なパーティクルの発生

ない．

4. 主成分分析を用いたパーティクルフィルタの高速化

物体は，反発力や摩擦などの外力が加わらない限り，静止または等速度運動をする．逆に，物体が運動を変化させるときには，ある外力が加わっているはずである．物体が非剛体である場合には，この外力がその物体の形状の変化として現れる場合がある．例えば，ボールが壁にあたってはねかえる（速度が大きく変化する）時には，ボールが壁に衝突する際に変形が起こる．また，本稿で取り上げた金魚の場合は，加速や方向転換の際に，水からの反力を得るために体をくねらせる．本稿では，この点に着目し，物体が運動する際の形状変化と運動変化の相関関係・因果関係を事前に観測・学習しておき，そこで獲得した運動パターンの範囲内にパーティクルを撒く方法を提案する．このアプローチによって，これまでのパーティクルフィルタの実装が抱える 3.2 節で述べた，状態の次元数の増大や不適切なパーティクルの存在という問題を解決することができる．

4.1 オフライン学習段階

4.1.1 データの獲得

物体の形状変化と運動変化の関係を学習するためには，正確なトラッキング結果が必要となる．ただし，この学習はオフラインで行えばよいので，時間がかかっても構わない．そこで，3.1 節に示した実装で，diffuse におけるパーティクルの分散である $\boldsymbol{\delta}$ の各要素を十分大きくとり，十分なパーティクル数を用いてトラッキングを行う．そして，その結果系列 $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{t_0}\}$ を獲得する．

4.1.2 形状変化・運動変化パラメータの抽出

獲得された $\mathbf{x}_t$ の各要素間からは，物体の形状変化や運動変化を直接的に表わすものではないので，このデータから直接的に相関関係は得られない．そこで，このデータから，形状変化・運動変化を表す情報を抽出する必要がある．

金魚の例では， $\mathbf{x}$ の要素 a, b, θ, ϕ から得られる運動変化は，金魚の向きの変化 θ' と移動速度の変化（加速度） (a'', b'') である．ただし，加速度については，金魚の向きからの相対値

$(a''^{(\text{std})}, b''^{(\text{std})})$ で表現するものとする。

各パラメータの算出方法を示す。まず、金魚の体の向き θ' と体の折れ曲がり角 ϕ_t は獲得データ X の隣接データ間の差分によって容易に得ることができる。

$$\theta'_t = \theta_t - \theta_{t-1} \quad (9)$$

$$\phi'_t = \phi_t - \phi_{t-1} \quad (10)$$

同様に、運動速度も隣接フレーム間の差分で得られる。

$$\begin{pmatrix} a'_t \\ b'_t \end{pmatrix} = \begin{pmatrix} a_t \\ b_t \end{pmatrix} - \begin{pmatrix} a_{t-1} \\ b_{t-1} \end{pmatrix} \quad (11)$$

加速度は、運動速度の差分である。

$$\begin{pmatrix} a''_t \\ b''_t \end{pmatrix} = \begin{pmatrix} a'_t \\ b'_t \end{pmatrix} - \begin{pmatrix} a'_{t-1} \\ b'_{t-1} \end{pmatrix} \quad (12)$$

この加速度の向きを金魚の向きについて正規化する。

$$\begin{pmatrix} a''^{(\text{std})}_t \\ b''^{(\text{std})}_t \end{pmatrix} = \begin{pmatrix} \cos(-\theta_t) & -\sin(-\theta_t) \\ \sin(-\theta_t) & \cos(-\theta_t) \end{pmatrix} \begin{pmatrix} a''_t \\ b''_t \end{pmatrix} \quad (13)$$

以上より、金魚の形状および運動のパラメータ組 $\mathbf{y}_t = (a''^{(\text{std})}_t, b''^{(\text{std})}_t, \theta'_t, \phi'_t)$ の系列 Y を獲得する。

4.1.3 主成分分析による学習

データセット $Y = \{\mathbf{y}_t\}$ について主成分分析を行い、分布の傾向を学習する。 Y の共分散行列の固有値を大きい順に $\lambda_1, \dots, \lambda_4$ 、それに対応する固有ベクトルを $\mathbf{u}_1, \dots, \mathbf{u}_4$ とし、特徴ベクトル $W = \{\mathbf{w}_t\}$ とすると、以下の関係式が成り立つ。

$$\begin{pmatrix} a''^{(\text{std})}_t \\ b''^{(\text{std})}_t \\ \theta'_t \\ \phi'_t \end{pmatrix} = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4] \begin{pmatrix} w_{1t} \\ w_{2t} \\ w_{3t} \\ w_{4t} \end{pmatrix} \quad (14)$$

また、 $\lambda_1, \dots, \lambda_4$ は、それぞれ $\mathbf{u}_1, \dots, \mathbf{u}_4$ 方向の分散となる。

4.2 オンライン実行段階

オンライン実行段階では、4.1 節で得られた学習結果を反映した予測パーティクル生成を行うことで、パーティクル数を削減して高速なトラッキングを実現する。ここでは、学習結果からこのような効果的な予測パーティクルを生成する方法について述べる。

まず、パーティクルの状態パラメータに、方向を正規化した速度パラメータ $a'^{(\text{std})}, b'^{(\text{std})}$ を追加し、 $\mathbf{s}_t^{i(\text{ex})} = (a, b, a'^{(\text{std})}, b'^{(\text{std})}, \theta, \phi)$ とする。そして、予測パーティクルの生成を表す式 (8) において、位置パラメータ (a, b) をこの正規化速度パラメータ $(a'^{(\text{std})}, b'^{(\text{std})})$ に変更する。そして、それらのパラメータについて静止モデルを仮定する。すなわち、drift は単位行列で表現される。

$$\begin{pmatrix} a'^{(\text{std})}_t \\ b'^{(\text{std})}_t \\ \theta_t \\ \phi_t \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} a'^{(\text{std})}_{t-1} \\ b'^{(\text{std})}_{t-1} \\ \theta_{t-1} \\ \phi_{t-1} \end{pmatrix} + \begin{pmatrix} \delta_{a'}^{(\text{std})} \\ \delta_{b'}^{(\text{std})} \\ \delta_\theta \\ \delta_\phi \end{pmatrix} \quad (15)$$

そして、式 (8) から削除された位置パラメータ a, b については、この状態遷移によって得られたパラメータを使って算出する。

$$\begin{pmatrix} a_t \\ b_t \end{pmatrix} = \begin{pmatrix} a_{t-1} \\ b_{t-1} \end{pmatrix} + \begin{pmatrix} \cos \theta_t & -\sin \theta_t \\ \sin \theta_t & \cos \theta_t \end{pmatrix} \begin{pmatrix} \delta_{a'}^{(\text{std})} \\ \delta_{b'}^{(\text{std})} \end{pmatrix} \quad (16)$$

以上の変形により、式 (8) で行われていた状態遷移メカニズムはほとんど変更されない。また、尤度計算についても、必要なパラメータがすべて $\mathbf{s}_t^{i(\text{ex})}$ に含まれているので、3.1 節と同一の計算によって行うことが可能である。

ここで、式 (15) に着目すると、右辺の末尾のベクトル $(\delta_{a'}^{(\text{std})}, \delta_{b'}^{(\text{std})}, \delta_\theta, \delta_\phi)$ の各要素は、4.1 で着目した形状変化・速度変化に対応している。そこで、通常のパーティクルフィルタのようにこれらのパラメータに独立にランダム値を与えるのではなく、データセットで学習した傾向に従った diffuse ベクトル $(\delta_{a'}^{(\text{std})}, \delta_{b'}^{(\text{std})}, \delta_\theta, \delta_\phi)$ を代入することで、効果的なパーティクルの生成を行う。これは、固有値 $\lambda_1, \dots, \lambda_4$ を分散とするランダム値 $\delta_{\lambda_1}, \dots, \delta_{\lambda_4}$ を選び、式 (14) に代入することで得られる。

$$\begin{pmatrix} \delta_{a'}^{(\text{std})} \\ \delta_{b'}^{(\text{std})} \\ \delta_\theta \\ \delta_\phi \end{pmatrix} = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4] \begin{pmatrix} \delta_{\lambda_1} \\ \delta_{\lambda_2} \\ \delta_{\lambda_3} \\ \delta_{\lambda_4} \end{pmatrix} \quad (17)$$

なお、このパーティクル予測部以外については、3.1 節で示した処理と同一である。

5. 評価実験

本稿の提案手法の有効性を確認するために、水槽で泳ぐ金魚を上から撮影した動画画像を用いて実験を行った。

まず、オフライン学習のために、10 秒前後の動画画像を 10 セット用意し、各動画画像につき 1 つの金魚を 3.1 節に示したパーティクルフィルタによってトラッキングを行った。なお、これは一般のパーティクルフィルタによる実装である。このとき、正しくトラッキングするために必要なパーティクル数は 3000 であり、処理速度は約 2.5(fps) であった。これに対して、学習結果を用いて効果的なパーティクル生成を行う提案手法では、パーティクル数 500 で同等のトラッキング能力を示し、処理速度は約 15(fps) となった。提案手法によって、処理が大幅に高速化されることが確認できる。

逆に、同数のパーティクル（すなわち処理速度が等しい）という条件下で、一般のパーティクルフィルタと提案手法の比較を行った。各手法によってトラッキングを行った様子を、それぞれ図 6、図 7 に示す。なお、図中に青線で重量表示されているのが、生成した各パーティクルに対応するモデルであり、見やすさのためにそれぞれの図において少数個（10 個）のパーティクルのみを表示している。一般のパーティクルフィルタでは、現実にはあり得ない位置・形状を含んでいるのに対して、提案手法では物理的に妥当だと思われる位置・形状になっており、効果的なパーティクル生成が行われていることが確認できる。

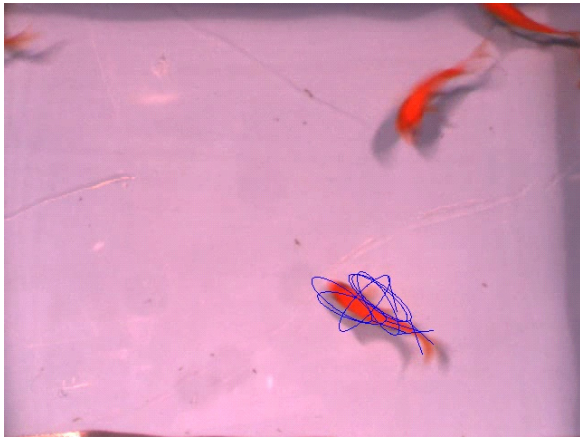


図 6 一般のパーティクルフィルタで生成されるパーティクル

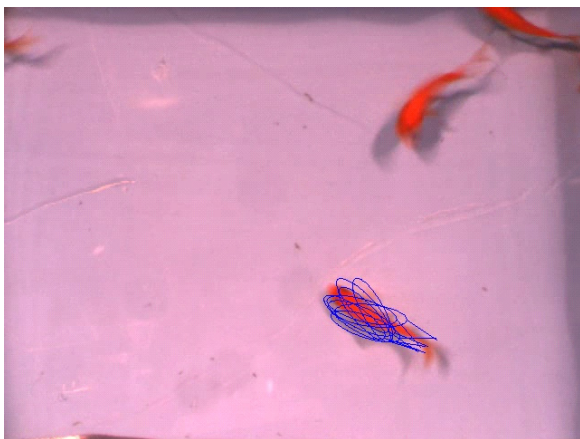


図 7 提案手法で生成されるパーティクル

6. おわりに

本稿では、パーティクルフィルタによる非剛体物体のトラッキングにおいて、効果的にパーティクルを生成することで高速なトラッキングを実現する手法を提案した。オフライン段階でゆっくり正確なトラッキングを行ってその物体の形状変化と運動変化の相関関係を主成分分析で学習しておき、オンライン実行時には、その学習結果に従ってパーティクルを生成することにより、パーティクル数を抑えながら高いトラッキング性能を実現した。水槽で泳ぐ金魚を対象として実験を行い、提案手法の有効性を確認した。

提案手法には、物理的に不自然な運動・形状変化を表すパーティクルを抑制するという特徴がある。これによって、無駄なパーティクル生成を抑えることができるため、トラッキングの処理速度の向上につながる。本稿では主に、実験によってこの高速化について確認した。しかし、この物理的に不自然な運動・形状変化が候補にならないという特長は、多数の対象が複雑に重なり合いながら運動する際に、個々の対象の時間方向における対応付けの精度を向上させるためにも有効であると考えられる。この効果について、実験を通じて確認することが今後の課題である。また、壁にあたってはね返るボールのトラッキングなど、異なる事例において提案手法の有効性を確認することも、提案手法の汎用性を確認する上で重要な課題である。さら

に、本稿では、扱う対象ごとに適切なモデルを与えているが、北島らの手法[9]のように対象モデルを自動的に獲得する手法なども検討する必要があると考えられる。

謝 辞

本研究で用いた金魚の映像を提供していただいた奈良先端科学技術大学院大学情報科学研究科の木戸出正継教授、波部斉助教、山本亮平氏に深く感謝いたします。

文 献

- [1] M.Isard and A.Blake, “CONDENSATION — Conditional Density Propagation for Visual Tracking,” International Journal of Computer Vision, Vol.29, No.1, pp.5–29, 1998.
- [2] Y.Wang, J.Wu, A.Kassim, “Particle Filter for Visual Tracking Using Multiple Cameras,” Proc. IAPR Conference on Machine Vision Applications, Vol.1, pp.298.301, 2005.
- [3] 松元郁佑, 加藤丈和, 和田俊和, “複数カメラを用いた Condensation によるオクルージョンにロバストな人物追跡,” 画像の認識・理解シンポジウム (MIRU2006), pp.291–506, 2006.
- [4] T.Bando, T.Shibata, K.Doya, S.Ishii, “Switching Particle Filters for Efficient Visual Tracking,” Robotics and Autonomous Systems, pp.873–884, 2006.
- [5] R.Urtasun, D.J.Fleet, A.Hertzmann, P.Fua, “Priors for People Tracking from Small Tracking Sets,” Proc. IEEE International Conference on Computer Vision, pp.403–410, 2005.
- [6] J.Pang, Q.Huang, S.Jiang, “Monocular Tracking 3D People with Back Constrained Scaled Gaussian Process Latent Variable Models,” Asia-Pacific Workshop on Visual Information Processing, 2006.
- [7] 加藤丈和, “パーティクルフィルタとその実装法,” 情報処理学会研究報告, 2007–CVIM–157, pp.161–168, 2007.
- [8] R.Kalman, “A new approach to linear filtering and prediction problems,” Trans. ASME — Journal of Basic Engineering, pp.35–45, 1960.
- [9] 北島利浩, 浮田宗伯, 木戸出正継, “スティックと楕円の統合モデルによる外見的特徴の少ない非剛体群の時系列位置・姿勢推定法,” 電子情報通信学会論文誌, Vol.J88-DII, No.2, pp.334–346, 2005.