

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^{†a)} 角所 考^{††} 美濃 導彦[†]

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

Ikuhisa MITSUGAMI^{†a)}, Koh KAKUSHO^{††}, and Michihiko MINOH[†]

あらまし 本論文では、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手法だが、非剛体物体の場合は、それらのパラメータ間に相関性がある場合があるにもかかわらず、それらをすべて独立に推定するために効率が悪い。そこで本論文では、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

1. ま え が き

パーティクルフィルタは、非線形・非ガウス形の状態空間に対して効率良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、その空間中でのゆう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、それぞれのゆう度を評価し、ゆう度分布を更新していく手法である。このようなモンテカルロ法に基づくアプローチであるため、一般に、状態空間の次元数に応じて必要となるパーティクル数も変化する。例えば、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、複数の関節をも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、本論文では、パーティクルフィルタによる非剛体物体トラッキングにおいて、状態空間中への不要なパーティクル生成を抑えて効率的に配置することを目標とする。上述のとおり、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効果的であると考えられる。本論文では、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）現象に着目すると、ボールが壁に衝突する際に変形が起こる。また、動物は、加速や方向転換の際に、何らかの姿勢変化を起こしていることが多い。これらの例のように、非剛体物体は、

[†] 京都大学学術情報メディアセンター，京都市
Academic Center for Computing and Media Studies, Kyoto
University, Yoshida Nihonmatsu-cho, Sakyo-ku, Kyoto-shi,
606-8501 Japan

^{††} 関西学院大学理工学部，三田市
School of Science and Technology, Kwansai Gakuin Univer-
sity, 2-1 Gakuen, Sanda-shi, 669-1337 Japan

a) E-mail: mitsugami@mm.media.kyoto-u.ac.jp

* 本論文は第 12 回画像の認識・理解シンポジウム推薦論文である。

自らの運動に変化を生じる際に外力を受けて（あるいは受けるために）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、その関係に応じて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、本論文では、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でサンプリング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とり得る形状変化・運動変化の範囲内のみをサンプリングする手法を提案する。

関連する既存研究として、非剛体物体を対象として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]、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]、更に、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。これらは、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。これらに対して、本論文の手法は、形状変化のみでなく、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相関関係に着目し、更にその相関関係が得られるような運動パラメータを陽に表現し、パーティクルフィルタに適用している点に新規性があると考えられる。

パーティクルフィルタにおける効率的なサンプリングに関する他のアプローチとして、[7] や [8] では、対象の色的一致度などの単純・低レベルな特徴量と、形状パラメータなどの複雑・高レベルな特徴量を利用してトラッキングを行う際に、低レベルの特徴量によって先に探索範囲を限定することで、計算コストのかかる高レベル特徴量に関するサンプリング数を削減するアプローチが提案されている。これらの手法は、互いに独立な低レベル・高レベル特徴量をカスケード型で処理することで効率化を図っている手法である。本論文のアプローチは、この 2 レベルの特徴量のうち高レベルの特徴量をどのように効率化するかに着目している点で、これらの手法とは異なると考えられる。また、本論文の手法を含めた上述の既存研究が、状態変数の選択方法・処理方法に着目しているのに対して、対象の運動モデルに着目した手法 [9] も提案されている。

この手法では、対象のオプティカルフローを算出して、それを運動モデルに反映させることで探索範囲の削減を図っている。ただし、この手法は、対象が有効なオプティカルフローを計算できる程度にテクスチャのある物体に限定されるという欠点をもつ。

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 [10] をもとにした手法である。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 x_t 、時刻 t までの状態ベクトルの履歴を $X_t = \{x_1, x_2, \dots, x_t\}$ とする。また、時刻 t の観測ベクトルを z_t 、その履歴を $Z_t = \{z_1, z_2, \dots, z_t\}$ 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 x_t の推定）を、観測系列 Z_t が与えられたときの状態ベクトル x_t に関する事後確率分布 $p(x_t|Z_t)$ 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 x_t の事後確率 $p(x_t|Z_t)$ を推定するのは一般に容易ではない。これに対して、状態 x_t を指定した際の観測値 z_t のゆう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 $p(x_t|Z_t)$ を次式のように変換する。

$$\begin{aligned} p(x_t|Z_t) &= p(x_t|z_t, Z_{t-1}) \\ &= \frac{p(x_t, z_t, Z_{t-1})}{p(z_t, Z_{t-1})} \\ &= \frac{p(z_t|x_t, Z_{t-1})p(x_t, Z_{t-1})}{p(z_t, Z_{t-1})} \\ &= \frac{p(z_t|x_t)p(Z_{t-1})p(x_t|Z_{t-1})}{p(z_t, Z_{t-1})} \\ &= \frac{p(z_t|x_t)p(x_t|Z_{t-1})}{p(z_t|Z_{t-1})} \end{aligned} \quad (1)$$

この式で、分母の $p(z_t|Z_{t-1})$ は、各観測 z_t が独立であることから $p(z_t)$ に等しく、また、状態 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 $p(x_t|Z_t)$ は、次式のように時刻 t における事前確率 $p(x_t|Z_{t-1})$ と観測値のゆう度 $p(z_t|x_t)$ の積で求められることが分かる。

$$p(x_t|Z_t) = k_t p(z_t|x_t)p(x_t|Z_{t-1}) \quad (2)$$

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．

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

すなわち、時刻 t における対象の状態 x_t は、直前時刻 $t-1$ の状態 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 $p(x_t|Z_{t-1})$ は次式のように計算できる．

$$\begin{aligned} p(x_t|Z_{t-1}) &= \int p(x_t, x_{t-1}|Z_{t-1}) dx_{t-1} \\ &= \int \frac{p(x_t, x_{t-1}, Z_{t-1})}{p(Z_{t-1})} dx_{t-1} \\ &= \int \frac{p(x_{t-1}, Z_{t-1})p(x_t|x_{t-1}, Z_{t-1})}{p(Z_{t-1})} dx_{t-1} \\ &= \int p(x_t|x_{t-1})p(x_{t-1}|Z_{t-1}) dx_{t-1} \quad (4) \end{aligned}$$

この式の右辺に含まれる $p(x_t|x_{t-1})$ は、隣接 2 時刻間の状態間の遷移確率を表す．例えば、状態 x が微小時間内で変化しないという運動モデルをもつ場合、遷移確率は、

$$p(x_t|x_{t-1}) = \exp\left\{-\frac{|x_t - x_{t-1}|^2}{\sigma}\right\} \quad (5)$$

などの形で表される．これは、 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 x_t が 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 $p(x_t|Z_t)$ に関する漸化式が構成されており、これを用いると $p(x_t|Z_t)$ 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 $p(x_t|x_{t-1})$ と観測値に対するゆう度 $p(z_t|x_t)$ の二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 $p(x_t|Z_t)$ 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 [11] 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 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得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

り得る．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 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、このような要求にこた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 $p(x_{t-1}|Z_{t-1})$ に応じて i 個のパーティクル 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に相当する処理を、各 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 s_{t-1}^i は、まず運動モデルに従って遷移し (drift)、更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 s_t^i とする．

• ゆう度推定

前節の式 (2) に相当するゆう度計算を、予測パーティクル s_t^i ごとに、現時刻 t における観測値 z_t に対するゆう度 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ゆう度の合計が 1 になるように正規化する．

$$\pi_t^i = \frac{p(z_t|x_t = s_t^i)}{\sum_{i=1}^N p(z_t|x_t = s_t^i)} \quad (6)$$

この π_t^i の分布が、現時刻 t における事後分布 $p(x_t|Z_t)$ 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．例えば、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.

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実装法

非剛体の移動物体の一つとして, 本論文では, 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_t, b_t) , 金魚の向きを表す楕円の長軸方向 θ_t , 金魚の体の曲がり具合を表す角度 ϕ_t という 4 パラメータで表現され, これを状態 x_t として使用する.

ゆう度計算については, 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, モデルの楕円上及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ゆう度 π_i は, この d_j のうち最大長のもの d_{\max} を使って, 以下のように与える.

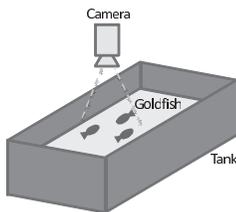


図 1 撮影環境
Fig. 1 The environment.



図 2 撮影画像例
Fig. 2 A captured image.

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

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 $s_t^i = (a_t, b_t, \theta_t, \phi_t)^T$ は以下の形で表される.

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

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, $\delta = (\delta_a, \delta_b, \delta_\theta, \delta_\phi)^T$ は, 遷移確率に応じ

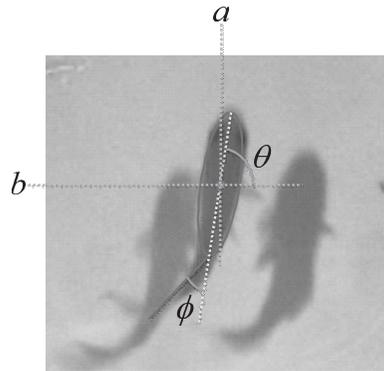


図 3 金魚モデル
Fig. 3 The model for a goldfish.

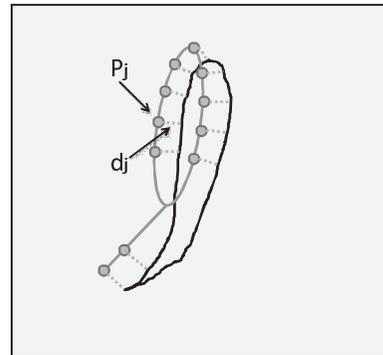


図 4 ゆう度推定
Fig. 4 Likelihood estimation.

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．

上述の静止モデルの場合，状態変数の中に速度情報をもたないため，速度の大きな対象の場合には，位置に関する diffuse δ_a, δ_b の分散をその速度に応じて大きくとらなければならない．また，diffuse の各要素は平均 0 のガウス分布に従うランダム値であるため，一定の方向に進む対象であっても全方向にパーティクルを配置することになり効率が悪くなる．このような場合には，上述の静止モデルではなく，速度情報 (a'_t, b'_t) も状態変数にもつ等速度モデルが用いられることもある．

$$\begin{pmatrix} a_t \\ b_t \\ a'_t \\ b'_t \\ \theta_t \\ \phi_t \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{t-1} \\ b_{t-1} \\ a'_{t-1} \\ b'_{t-1} \\ \theta_{t-1} \\ \phi_{t-1} \end{pmatrix} + \begin{pmatrix} \delta_a \\ \delta_b \\ \delta_{a'} \\ \delta_{b'} \\ \delta_\theta \\ \delta_\phi \end{pmatrix} \quad (9)$$

この等速度モデルでは，位置・速度に関する diffuse の分散を比較的小さくすることができるが，状態変数の次元数は増大するため，必ずしも効率的になるわけではない．

3.2 一般の実装法の問題点

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，複数の関節をも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．また，3.1 でも述べたように，運動に関する状態変数についても，パーティクル数を少なくできるように diffuse の分散を小さくしようとす

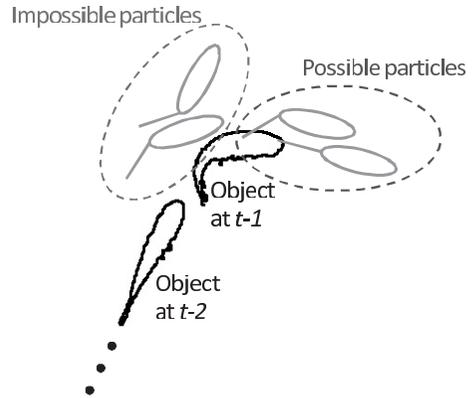


図 5 不自然なパーティクルの発生
Fig. 5 Possible/impossible particles.

ると，状態変数の次元数が増加してしまい，結果的にパーティクル数を減らすことができず，処理速度の向上にならないことが多い．

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，状態変数ベクトルの各要素は互いに独立として扱われている．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，対象の形状変化・運動変化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，反発力や摩擦力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．本論文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ており，そのような形状変化は観測画像から獲得可能である．本論文では，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まく方法を提案する。

具体的には、まず式 (8), (9) に着目する。これらの式を見ると、diffuse の各要素は、それに対応する状態変数の一次微分に相当することが分かる。つまり、 (δ_a, δ_b) , $(\delta_{a'}, \delta_{b'})$, $(\delta_\theta, \delta_\phi)$ はそれぞれ速度、加速度 (本論文で運動変化と呼んでいるものに相当する)、形状変化に対応している。そこで提案手法では、運動変化と形状変化が陽に相関関係をもつように状態変数の表現方法を変更し、その表現方法もとの運動変化と形状変化を主成分分析で学習し、その学習結果を反映した diffuse を生成させる。以上の処理について、その詳細を以下で順に示す。

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、時間がかかってもかまわない。そこで、3.1 に示した実装で、diffuse におけるパーティクルの分散である δ 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 $X = \{x_1, x_2, \dots, x_{t_0}\}$ 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 x_t の要素は、画像座標系における金魚の位置・姿勢と形状に関するパラメータであり、これらがそのまま相関関係をもつわけではない。相関関係を得るには、これらのパラメータを、画像座標系によらない金魚中心座標における運動変化・形状変化パラメータに変換する必要がある。

金魚の例では、 x_t の要素 $a_t, b_t, \theta_t, \phi_t$ 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_t'', b_t'') である。ただし、加速度については、金魚の位置・向きを基準とする座標系で $(a_t''^{(std)}, b_t''^{(std)})$ 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'_t 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。

$$\theta'_t = \theta_t - \theta_{t-1} \quad (10)$$

$$\phi'_t = \phi_t - \phi_{t-1} \quad (11)$$

同様に、運動速度も隣接フレーム間の差分で得られる。

$$\begin{pmatrix} a'_t \\ b'_t \end{pmatrix} = \begin{pmatrix} a_t \\ b_t \end{pmatrix} - \begin{pmatrix} a_{t-1} \\ b_{t-1} \end{pmatrix} \quad (12)$$

加速度は、運動速度の差分である。

$$\begin{pmatrix} a''_t \\ b''_t \end{pmatrix} = \begin{pmatrix} a'_t \\ b'_t \end{pmatrix} - \begin{pmatrix} a'_{t-1} \\ b'_{t-1} \end{pmatrix} \quad (13)$$

この加速度の向きを金魚の向きについて正規化する。

$$\begin{pmatrix} a_t''^{(std)} \\ b_t''^{(std)} \end{pmatrix} = \begin{pmatrix} \cos(-\theta_t) & -\sin(-\theta_t) \\ \sin(-\theta_t) & \cos(-\theta_t) \end{pmatrix} \begin{pmatrix} a_t'' \\ b_t'' \end{pmatrix} \quad (14)$$

以上より、金魚の形状及び運動のパラメータ組 $y_t = (a_t''^{(std)}, b_t''^{(std)}, \theta'_t, \phi'_t)$ 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 $Y = \{y_t\}$ 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 u_1, \dots, u_4 とし、特徴ベクトル $W = \{w_t\}$ とすると、以下の関係式が成り立つ。

$$\begin{pmatrix} a_t''^{(std)} \\ b_t''^{(std)} \\ \theta'_t \\ \phi'_t \end{pmatrix} = [u_1, u_2, u_3, u_4] \begin{pmatrix} w_{1t} \\ w_{2t} \\ w_{3t} \\ w_{4t} \end{pmatrix} \quad (15)$$

また、 $\lambda_1, \dots, \lambda_4$ は、それぞれ u_1, \dots, 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 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 $a_t'^{(std)}, b_t'^{(std)t}$ を追加し、 $s_t^{i(ex)} = (a_t, b_t, a_t'^{(std)}, b_t'^{(std)}, \theta_t, \phi_t)$ 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_t, b_t) をこの正規化速度パラメータ $(a_t'^{(std)}, b_t'^{(std)})$ 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。

$$\begin{pmatrix} a_t^{(std)} \\ b_t^{(std)} \\ \theta_t \\ \phi_t \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{t-1}^{(std)} \\ b_{t-1}^{(std)} \\ \theta_{t-1} \\ \phi_{t-1} \end{pmatrix} + \begin{pmatrix} \delta_a^{(std)} \\ \delta_b^{(std)} \\ \delta_\theta \\ \delta_\phi \end{pmatrix} \quad (16)$$

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。

$$\begin{pmatrix} a_t \\ b_t \end{pmatrix} = \begin{pmatrix} a_{t-1} \\ b_{t-1} \end{pmatrix} + \begin{pmatrix} \cos \theta_t & -\sin \theta_t \\ \sin \theta_t & \cos \theta_t \end{pmatrix} \begin{pmatrix} a_t^{(std)} \\ b_t^{(std)} \end{pmatrix} \quad (17)$$

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、ゆ一度計算についても、必要なパラメータがすべて $s_t^{i(ex)}$ に含まれているので、3.1 と同一の計算によって行うことが可能である。

ここで、式 (16) に着目すると、右辺の末尾のベクトル $(\delta_a^{(std)}, \delta_b^{(std)}, \delta_\theta, \delta_\phi)$ 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 $(\delta_a^{(std)}, \delta_b^{(std)}, \delta_\theta, \delta_\phi)$ 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 $\lambda_1, \dots, \lambda_4$ を分散とする正規分布に従ったランダム値 $\delta_{\lambda_1}, \dots, \delta_{\lambda_4}$ を選び、式 (15) に代入することで得られる。

$$\begin{pmatrix} \delta_a^{(std)} \\ \delta_b^{(std)} \\ \delta_\theta \\ \delta_\phi \end{pmatrix} = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4] \begin{pmatrix} \delta_{\lambda_1} \\ \delta_{\lambda_2} \\ \delta_{\lambda_3} \\ \delta_{\lambda_4} \end{pmatrix} \quad (18)$$

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

5. 評価実験

本論文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方から撮影した動画画像を用いて実験を行った。なお、画像のサイズは 640 (pixel) × 480 (pixel) (VGA) であり、使用した PC は CPU: Intel Xeon 5110(1.6 GHz)、メモリ: 2 GByte、OS: Debian GNU/Linux (etch) である。

まず、オフライン学習用に、1 匹以上の金魚が映っ

ている約 10 秒の動画画像を 10 セット用意し、各動画画像の各金魚を 3.1 に示した一般のパーティクルフィルタを実装し、パーティクル数 3000 でトラッキングした。diffuse については、 $\delta_a, \delta_b, \delta_\theta, \delta_\phi$ が、平均 0、標準偏差がそれぞれ 30 (pixel)、30 (pixel)、 $\pi/4$ (rad)、 $\pi/4$ (rad) の正規分布に従うランダム値となるように定めた。これによって合計 20 匹の追跡結果を獲得し、これらがトラッキングに成功していることを、目視によって確認した。なお、トラッキング開始から終了までの全フレームにおいて、推定された (a_t, b_t) が対象としている金魚のシルエットに含まれていた場合を、トラッキングの成功と定めた。

続いて、上述の学習用の 10 セットと異なるテスト用動画画像を 10 セット用意し、そこに現れた 24 匹の金魚について、上述と同じ一般のパーティクルフィルタと提案手法のそれぞれでトラッキングを行った。ただし、この一般のパーティクルフィルタにおける diffuse は、上述の学習ステップにおけるトラッキング結果から得られる $(a'_t, b'_t, \theta'_t, \phi'_t)$ の平均・標準偏差をもつ正規分布に従うランダム値を用いた。

パーティクル数を変えながら両手法によるトラッキングの成功回数を比較したものを図 6 に示す。両手法とも、パーティクル数が 100 程度の場合は、状態空間に対してパーティクルが疎であるために結果は同様に悪いが、パーティクル数の増加に従い、その性能が向上している。ただし、提案手法は一般のパーティクルフィルタと比較して、その性能向上が著しく、一般手法におけるパーティクル数 3000 の結果と、提案手法におけるパーティクル数 500 の結果がほぼ同等であることが分かる。このとき処理速度は、一般手法が 2.5 fps、提案手法が 15 fps であり、約 6 倍高速化され

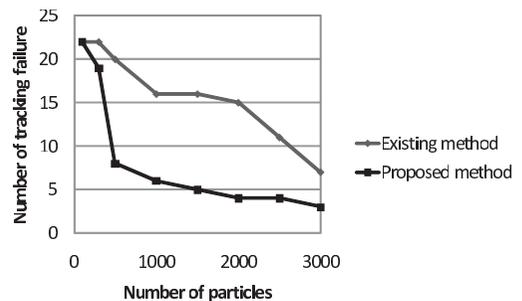


図 6 一般手法と提案手法のトラッキング成功回数の比較
Fig. 6 Comparison between the existing and proposed methods.

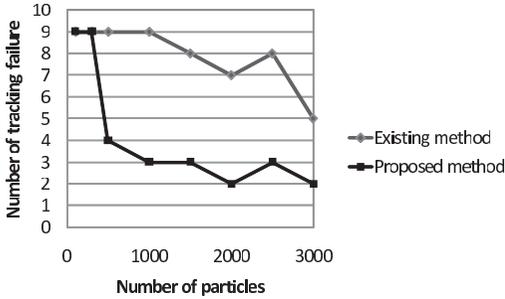


図 7 急激な運動変化・形状変化を起こす場合の性能比較
Fig.7 Comparison in the case that the shape and motion suddenly change.



図 8 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
Fig.8 Particles generated in the existing method.

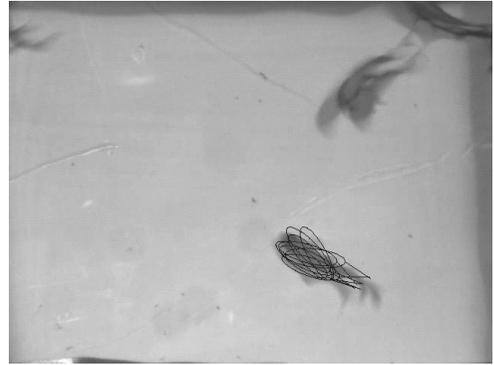


図 9 提案手法で生成されるパーティクル
Fig.9 Particles generated in the proposed method.

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。

6. む す び

本論文では、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。本論文では主に、実験によってこの高速化について確認した。しかし、この対象の運動として不自然なもの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、多数の対象が複雑に運動するような環境で実験を行う予定である。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。更に、本論文では、扱う対象ごとに適切

ていることが分かる。

また、この 24 例のうち、急激な形状変化・運動変化(4 フレーム以内で 60 度以上の運動方向変化があったもの)を起こした 9 例を抽出し、これについて上述と同様の比較を行ったものを、図 7 に示す。一般手法では、パーティクル数を増加させてもこの急激な変化に対応しにくいのにに対して、提案手法では少ないパーティクル数でもこの急激な変化に対応できていることが分かる。

以上の効果を視覚的に理解するために、両手法によって生成されているパーティクルをそれぞれ図 8、図 9 に示す。なお、図中に楕円と線分で重畳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る。ただし、見やすさのために、それぞれの図において表示しているのは、パーティクル集合の中からランダムに選んだ 10 個のパーティクルのみである。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

なモデルを与えているが、北島らの手法 [12] 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頂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、波部斉助教に深く感謝致します。

文 献

- [1] M. Isard and A. Blake, "Condensation — Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol.29, no.1, pp.5–29, 1998.
- [2] Y. Wang, J. Wu, and 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, and 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, and 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, and S. Jiang, "Monocular tracking 3D people with back constrained scaled Gaussian process latent variable models," *Asia-Pacific Workshop on Visual Information Processing*, 2006.
- [7] M. Isard and A. Blake, "ICondensation: Unifying low-level and high-level tracking in a stochastic framework," *Proc. European Conference on Computer Vision (ECCV)*, vol.1, pp.893–908, 1998.
- [8] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.30, no.10, pp.1728–1740, 2008.
- [9] K. Kawamoto, "Optical flow-driven motion model with automatic variance adjustment for adaptive tracking," *Proc. Asian Conference on Computer Vision (ACCV)*, pp.555–564, 2007.
- [10] 加藤丈和, "パーティクルフィルタとその実装法," *情処学研報*, 2007-CVIM-157, pp.161–168, 2007.
- [11] R. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME — J. Basic Engineering*, pp.35–45, 1960.
- [12] 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," *信学論 (D-II)*, vol.J88-D-II, no.2, pp.334–346, Feb. 2005.

(平成 20 年 10 月 10 日受付, 21 年 3 月 2 日再受付)



満上 育久 (正員)

2001 京大・工・電気電子卒. 2007 奈良先端科学技術大学院大学情報科学研究科博士後期課程了. 同年, 京都大学学術情報メディアセンター研究員 (科学技術振興). コンピュータビジョン, 対象抽出・追跡, プロジェクトによる実環境視覚情報提示に関する研究に従事. 博士 (工学).



角所 考 (正員)

昭 63 名大・工・電気卒. 平 5 阪大大学院工学研究科通信工学専攻博士課程了. 平 4~6 日本学術振興会特別研究員. 平 5~6 スタンフォード大学ロボティクス研究所客員研究員. 平 6 大阪大学産業科学研究科助手. 平 9 京都大学総合情報メディアセンター助教授. 平 21 関西学院大学理工学部教授, 現在に至る. 視覚メディア処理, コミュニケーション, インタラクシオンに関する研究に従事. 博士 (工学). IEEE, ACM, 情報処理学会, 人工知能学会各会員.



美濃 導彦 (正員)

昭 53 京大・工・情報工学卒. 昭 58 同大学院博士課程了. 同年工学部助手, 昭 62~63 マサチューセッツ州立大学客員研究員, 平元京都大学工学部附属高度情報開発実験施設助教授, 平 7 同教授, 平 9 京都大学総合情報メディアセンター教授, 平 14 京都大学学術情報メディアセンター教授. 画像処理, 人工知能, 知的コミュニケーション関係の研究に従事. 工博. IEEE, ACM, 情報処理学会, 画像電子学会, 日本ロボット学会各会員.