

Kalman-filter 予測を用いた特徴点マッチングと Mean-Shift 探索の統合による広域特徴点追跡

富安 史陽^{1,a)} 平山 高嗣^{1,b)} 間瀬 健二^{1,c)}

概要：Mean-Shift を利用した特徴点追跡手法では、追跡対象の大きな移動に伴い、画像上における特徴点の移動距離が大きくなると、Mean-Shift 探索の結果が局所解に陥り、探索に失敗することがある。本研究では、Kalman-filter 予測に基づいた特徴点マッチングにより連続フレーム間での大域的な特徴点の対応付けを行った後に、対応点から Mean-Shift 探索することで詳細な特徴点の追跡を目指す。これにより、Mean-Shift の欠点である局所解に陥る問題を解決する。シミュレーション実験を行い、特徴点の移動距離毎の追跡精度を評価した。提案手法は比較対象に比べ、特徴点の平均追跡誤差を小さく保ちながら追跡することができた。また、スポーツを撮影した映像に対して提案手法を適用し、スポーツ選手の追跡を行った。

1. はじめに

現在、世の中には膨大な量の映像が溢れており、私達は、これらの映像をインターネットを介して共有している。映像に含まれる有益な情報などを自動的に抽出することができれば、様々なアプリケーションに応用することができる。例えば、サッカーなどのスポーツコンテンツに対して、観客が携帯電話等で撮影し、インターネット上に投稿することで、色々な視点の映像を共有することができる。スポーツ映像の場合、視聴者の多くはスポーツ選手に注目して映像を視聴するため、映像中における選手の位置情報は、有益な情報である。また、選手の位置情報を用いて、複数カメラ間の時間同期や位置合わせを行うこともできる。

映像から得られる情報の1つに、特徴点位置の時系列変化がある。特徴量とは、画像の局所領域から得られる色情報や輝度勾配の情報を記述したものであり、特徴点とは、特徴量を算出した一定領域の中心画像座標のことである。特徴点追跡とは、時刻方向の変化を t で表現した時に、 $t-1$ フレームの特徴点と類似した特徴量を持つ t フレームの画像座標を見つけ出すことである。特徴点追跡により、画像上における物体の移動軌跡などを得ることができる。

現在までに、特徴点を追跡する多くの研究が行われてきたが、その中の1つに Mean-Shift[1] を用いた特徴点追跡手法が存在する [5][6][7][8][9][10][11][12]。Mean-Shift は極

値探索手法であるため、狭領域において詳細な特徴点追跡が可能である。しかし、特徴点が大きく移動した際に、Mean-Shift の結果が局所解に陥るという問題がある。また、時間方向に連続する画像から検出した特徴点同士を、特徴量の類似度に基づいて対応付ける特徴点マッチングを利用した手法がある。この手法では、特徴点の対応付けにより、画像上での特徴点の移動量に関係なく特徴点を追跡できる。しかし、特徴量依存であるため、類似した特徴量を持つ特徴点が複数存在する場合に、誤対応が発生しやすくなり追跡精度が低くなる。

我々は、急加速や急停止などの急激な速度変化をするスポーツ選手の追跡を最終目的とし、画像上における特徴点の移動変化が大きな場合にも、特徴点を精度良く追跡することを目指す。提案手法は、Kalman-filter[2] による特徴点の移動予測に基づいた特徴点マッチングと Mean-Shift 探索の併用である。これにより、特徴点マッチングにより大域的な特徴点追跡を行った後に、Mean-Shift 探索により狭領域において詳細な特徴点の追跡が可能となる。本手法の特徴は、特徴点マッチングと Mean-Shift 探索の利点を残しながら、2つの手法を組み合わせることで各手法の欠点を解決する点である。各手法の利点と欠点を表1にまとめる。また、特徴量には、scale invariant feature transform (SIFT) [3] と同様の性質を持ち、SIFT よりも高速に計算可能な speeded up robust features (SURF) [4] を使用する。

比較対象を、Mean-Shift 探索、Kalman-filter に基づく Mean-Shift 探索、Kalman-filter に基づく特徴点マッチングとし、シミュレーション実験を行った。結果として、提案手法が比較手法に比べ最も小さな平均追跡誤差を示した。

¹ 名古屋大学大学院 情報科学研究科 社会システム情報学専攻
〒464-8603 愛知県名古屋市千種区不老町 1

a) tomiyasu@cmc.ss.is.nagoya-u.ac.jp
b) hirayama@is.nagoya-u.ac.jp
c) mase@nagoya-u.jp

表 1 特徴点マッチングと Mean-Shift 探索と提案手法の利点・欠点
Table 1 Advantage and disadvantage of corresponding point search, Mean-Shift search and proposed method.

追跡手法	大域的移動	狭域的移動
特徴点マッチング	○	×
Mean-Shift 探索	×	○
提案手法	○	○

以降、2章で特徴点追跡手法の関連研究についてまとめ、3章で提案手法を説明する。4章でシミュレーションによる評価実験を、5章でスポーツ映像に対する追跡実験を行い、6章でまとめる。

2. 関連研究

Mean-Shift[1]は、カーネル密度推定を用いた極値探索問題にロバストな統計的データ解析手法であり、コンピュータビジョンだけでなく幅広い応用が報告されている。Mean-Shiftの特徴点追跡への応用において、一般的にMean-Shiftは、追跡している特徴点が有する特徴量と類似した特徴量を持つ画素を探し出す、最頻値探索問題の解法として利用される。具体的には、画像内に存在する追跡点との特徴量誤差が最小となる極値点を反復探索することで決定する。Mean-Shiftは、計算コストが低いため高速に計算可能であり、実時間追跡などに利用される。さらに、1つ前のフレームにおける追跡点の画像座標から探索を開始することで、特徴点の移動距離が短い場合において、サブピクセル精度で特徴点を追跡することが可能である。Mean-Shiftを用いた特徴点追跡では、追跡する特徴点が有する特徴量としてカラーヒストグラム [5][6][7][8][9][10] や局所特徴量 [11][12] がよく用いられる。

カラーヒストグラム [5][6]は、特徴点を中心とする矩形などの一定領域内に含まれる画素から算出される。特徴量の表現方法は、HSV色空間やRGB色空間など様々であるが、追跡対象を含む一定領域をテンプレートとし、画像から切り出した探索領域とのカラーヒストグラム間のBhattacharyya係数などを比較することにより特徴量の類似度が計算される。カラーヒストグラムは、追跡対象の形状変化などに頑健であるという利点を持つ一方で、追跡対象が他物体の陰に隠れてしまい見えなくなるオクルージョンや照明変動などによる画素値の変化に弱いという欠点を持つ。さらに、追跡対象のスケール方向の変化に対して、大きさを変えた探索窓を複数枚用意してそれぞれ探索しなければならない。Collins [7]は、スケール方向への変化にロバストな追跡手法として、Mean-Shiftをスケール空間に適用することでこの問題を解決している。

また、局所特徴量とは、局所領域における輝度の勾配情報に基づいた特徴量 (SIFT[3] や SURF[4] など) である。都築ら [11]は、SIFT特徴量を用いたMean-Shift探索によ

る特徴点追跡手法を提案している。画像空間とスケール空間の両方に対して、Mean-Shiftを交互に適用することで特徴点の追跡をおこなっている。また、SIFT特徴量は、画像上における被写体の回転変化、スケール変化、輝度変化に頑健な特徴量である。したがって、カラーヒストグラムを用いる手法とは異なり照明変化に伴う輝度変化に頑健な手法となっている。

しかし、このようなMean-Shiftに基づく手法は、追跡対象が画像上を大きく移動することにより特徴点が前フレームの追跡結果座標から大きく移動すると、特徴点を追跡できなくなるという問題がある。なぜなら、Mean-Shiftが大域的極値に収束することを保証しておらず、容易に追跡開始点周辺の局所解に陥るからである。

この問題の解法として、Kalman-filter [2] や Particle-filter を利用して、Mean-Shift探索の初期値を変更する手法が存在する [8][9][10]。Comaniciuら [8]は、Kalman-filterにより特徴点の移動予測を行い、その予測位置からMean-Shift探索を行うことで、局所解に陥る可能性を軽減している。しかし、追跡対象がスポーツ選手のような場合、急加速や急停止などの速度変化が激しいため予測を外れることが多く、特徴点の追跡に失敗する。また、Particle-filter[9][10]の場合、ばら撒く粒子の数に計算精度が依存する。特徴点が大きく移動する場合は、Particleの数をある程度確保しなければならず、計算量が増加する一方で、追跡対象以外の局所解に陥る可能性が増えるために追跡が失敗する。

また、パノラマ画像作成 [13] やステレオ画像を用いた3次元モデル復元 [14] などに利用される、特徴点マッチングと呼ばれる手法がある。この手法は、特徴量の類似度に基づいて、異なる画像から検出された特徴点同士を対応付けるものである。パノラマ画像生成や3次元モデル復元では、撮影者の空間的な変化に対する撮影対象の見えの変化に特徴点マッチングを適用するが、これを撮影対象の時間的な変化に適用することで、特徴点を追跡することができる。しかし、この手法は、撮影対象の見えの変化に伴い、算出される特徴点の位置や特徴量に変化する上、追跡点と類似した特徴量を持つ特徴点が複数存在することもあり、正確な特徴点の対応付けが困難になるという欠点がある。さらに、最も単純な特徴点マッチングでは、画像から検出された全特徴点と比較する必要があるため計算量も増加する。

3. 特徴点マッチングと Mean-Shift 探索の逐次処理

提案手法のフローチャートを図 1 に示す。本手法は3つのステップからなり、 t フレーム目の特徴点の追跡についてステップ毎に2.1から2.3で説明する。2.1は特徴点検出であり、2.2は特徴点マッチングに基づく大域的特徴点追跡であり、2.3はMean-Shift探索 [11] に基づく局所的特徴点追跡である。手法の説明にあたり、特徴点群を $P(t)$ 、各

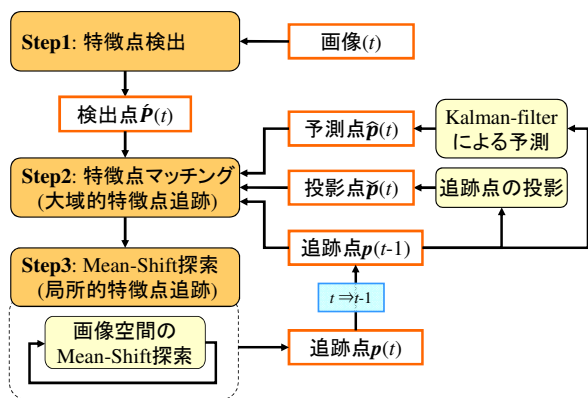


図 1 提案手法のフローチャート

Fig. 1 Flow chart of the proposed method.

表 2 各特徴点の名称と表記方法

Table 2 Each feature point name and notation system.

特徴点名	表記方法	特徴点数	補足
追跡点	$\mathbf{p}(t)$	N 点	
予測点	$\hat{\mathbf{p}}(t)$	N 点	ある追跡点に 1 つ
投影点	$\tilde{\mathbf{p}}(t)$	N 点	ある追跡点に 1 つ
検出点	$\hat{\mathbf{p}}(t)$	M 点	入力画像から検出

特徴点を $\mathbf{p}_i(t)$, 各特徴点における特徴ベクトルを $\mathbf{v}_i(t)$ とし, 下記のように表す.

$$\mathbf{P}(t) = \{\mathbf{p}_1(t), \dots, \mathbf{p}_N(t)\} \quad (1)$$

$$\mathbf{p}_i(t) = (\mathbf{x}_i(t), s_i(t)) \quad (i = 1, \dots, N) \quad (2)$$

$$\mathbf{v}_i(t) = \text{SURF}(\mathbf{x}_i(t), s_i(t)) = \{v_{i,0}, \dots, v_{i,127}\} \quad (i = 1, \dots, N) \quad (3)$$

特徴点 $\mathbf{p}_i(t)$ は, 2 つの空間的変数を持ち, 画像空間における画像座標値 $\mathbf{x}_i(t) = (x_i(t), y_i(t)) \in \mathbf{R}^2$ とスケール空間におけるスケール値 $s_i(t) \in \mathbf{R}$ である. また, $\text{SURF}(\mathbf{x}_i(t), s_i(t))$ は, 画像座標値 $\mathbf{x}_i(t)$ におけるスケール値 $s_i(t)$ での SURF 特徴ベクトル $\mathbf{v}_i(t)$ を計算する関数である.

ここで, 以降の説明で使用する特徴点をまとめる (表 2). 表 2 中の特徴点数 N と M は, 入力画像依存の値であり, 両者に大小関係は存在しない. さらに, 図 1 の Step2 では, 検出点 $\hat{\mathbf{p}}(t)$ と投影点 $\tilde{\mathbf{p}}(t)$, 予測点 $\hat{\mathbf{p}}(t)$, 追跡点 $\mathbf{p}(t-1)$ を入力として, 追跡点とマッチングする特徴点を求める. Step3 では, 特徴点マッチングの結果に基づいて, Mean-Shift 探索することで詳細な特徴点の追跡を行う.

以下では, $t-1$ フレームの特徴点追跡結果 (以下, 追跡点と表記する) が得られた状態で, t フレーム目の特徴点追跡について詳細を述べる. 追跡点の数を N 個とする.

3.1 特徴点検出

Step1 では, SURF 検出器により, 入力画像から特徴点を検出する. 検出された特徴点 (以下, 検出点と表記する) は, 128 次元の特徴ベクトルを持ち, 周囲の画素に比べて

特徴的な点である. 検出点は, Step2 の特徴点マッチングで利用する. 検出点群を $\hat{\mathbf{P}}(t)$, 各検出点を $\hat{\mathbf{p}}_j(t)$, その点における 128 次元の特徴ベクトルを $\mathbf{v}_j(t)$ と表記する. また, 入力画像から得られた検出点数を, M 個とする.

3.2 特徴点マッチングによる大域的特徴点追跡

Step2 では, 特徴点マッチングにより $t-1$ フレームの追跡点 $\mathbf{p}_i(t-1)$ と対応する特徴点を算出し, それに基づいて Mean-Shift 探索の初期値を決定する. 単純な特徴点マッチングは, 入力画像から算出された全検出点に対して追跡点とのマッチングを行う. しかし, この場合には, 計算量が増加し, 誤対応を起こす可能性が増加するという問題がある. そこで, 提案手法では, Kalman-filter[2] を用いて t フレームにおける追跡点の移動予測を行い, これに基づいて特徴点マッチング候補となる検出点の絞込みを行う.

3.2.1 Kalman-filter による追跡点の移動予測

Kalman-filter[2] を用いて追跡点の移動予測を行う. 追跡点は等加速度運動をすると仮定し, Kalman-filter の状態モデルと観測モデルを線形であると仮定すると, 式 (4) より Kalman-filter は追跡点の移動予測を行うことができる.

$$\hat{\mathbf{Q}}_i(t) = \mathbf{F}\mathbf{Q}_i(t-1) + \begin{pmatrix} \mathbf{0}_{4 \times 1} \\ w_x \\ w_y \end{pmatrix} \quad (4)$$

ここで, 状態モデル \mathbf{Q}_i は, 追跡点の画像上における画像座標値 $\mathbf{x}_i(t) = (x_i(t), y_i(t))$ と追跡点の速度 $(\dot{x}_i(t), \dot{y}_i(t))$, 加速度 $(\ddot{x}_i(t), \ddot{y}_i(t))$ を持つ. $t-1$ フレームの状態変数ベクトル $\mathbf{Q}_i(t-1)$ と t フレームの予測状態変数ベクトル $\hat{\mathbf{Q}}_i(t)$ は, 式 (5) となる.

$$\mathbf{Q}_i(t-1) = \begin{pmatrix} x_i(t-1) \\ y_i(t-1) \\ \dot{x}_i(t-1) \\ \dot{y}_i(t-1) \\ \ddot{x}_i(t-1) \\ \ddot{y}_i(t-1) \end{pmatrix}, \hat{\mathbf{Q}}_i(t) = \begin{pmatrix} \hat{x}_i(t) \\ \hat{y}_i(t) \\ \dot{\hat{x}}_i(t) \\ \dot{\hat{y}}_i(t) \\ \ddot{\hat{x}}_i(t) \\ \ddot{\hat{y}}_i(t) \end{pmatrix} \quad (5)$$

また, 状態遷移行列を式 (6) と表記する.

$$\mathbf{F} = \begin{pmatrix} 1 & 0 & \delta t & 0 & \frac{1}{2}\delta t^2 & 0 \\ 0 & 1 & 0 & \delta t & 0 & \frac{1}{2}\delta t^2 \\ 0 & 0 & 1 & 0 & \delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (6)$$

式 (6) 中の δt は, 画像間の経過時間を表している. さらに, w_x と w_y は, 期待値が 0 で標準偏差がそれぞれ σ_x と σ_y のガウスノイズである.

Kalman-filter により予測した t フレームにおける追跡点の画像座標値を $\hat{\mathbf{x}}_i(t)$ とし, $t-1$ フレームの追跡点 $\mathbf{p}_i(t-1)$

のスケール値 $s_i(t-1)$ を引き継ぐことで、予測した特徴点 (以降、予測点と表記する) とする。予測点を $\hat{p}_i(t)$ とし、この点における特徴ベクトルを $\hat{v}_i(t)$ と表記する。

$$\hat{p}_i(t) = (\hat{x}_i(t), \hat{s}_i(t)) \quad (i = 1, \dots, N) \quad (7)$$

$$\begin{aligned} \hat{v}_i(t) &= \text{SURF}(\hat{x}_i(t), \hat{s}_i(t)) \\ &= \{\hat{v}_{i,0}, \dots, \hat{v}_{i,127}\} \quad (i = 1, \dots, N) \end{aligned} \quad (8)$$

$t-1$ フレームの一位の追跡点に対し、 t フレームの予測点は 1 つである。

3.2.2 t フレームにおける追跡点の投影

$t-1$ フレームの追跡点を t フレームに投影し、追跡点 $p_i(t-1)$ の画像座標値 $x_i(t-1)$ とスケール値 $s_i(t-1)$ を引き継いだ特徴点を投影点として作成する。投影点を $\check{p}_i(t)$ とし、この点における特徴ベクトルを $\check{v}_i(t)$ と表記する。

$$\check{p}_i(t) = (\check{x}_i(t), \check{s}_i(t)) \quad (i = 1, \dots, N) \quad (9)$$

$$\begin{aligned} \check{v}_i(t) &= \text{SURF}(\check{x}_i(t), \check{s}_i(t)) \\ &= \{\check{v}_{i,0}, \dots, \check{v}_{i,127}\} \quad (i = 1, \dots, N) \end{aligned} \quad (10)$$

$t-1$ フレームの一位の追跡点に対し、 t フレームの投影点は 1 つである。

3.2.3 特徴点マッチング候補点の選定

画像上の特徴点間ユークリッド距離に応じて、式 (11) より特徴点マッチング候補となる特徴点を選定する。

$$\varepsilon \geq \frac{r_j}{r_i} \quad (\varepsilon \geq 1) \quad (11)$$

ここで、 $\varepsilon \in \mathbf{R}$ は、マッチング対象範囲調整定数である。距離 r_i は、 t フレームにおける投影点の画像座標値 $\check{x}_i(t)$ と予測点の画像座標値 $\hat{x}_i(t)$ との距離を表している (式 (12))。

$$r_i = \|\check{x}_i(t) - \hat{x}_i(t)\| \quad (12)$$

また、距離 r_j は、 t フレームの画像から算出された検出点の各画像座標値 $\hat{x}_j(t)$ と投影点の画像座標値 $\check{x}_i(t)$ または、予測点の画像座標値 $\hat{x}_i(t)$ との距離のうち小さい方の値であり、式 (13) である。

$$r_j = \min(\|\hat{x}_j(t) - \check{x}_i(t)\|, \|\hat{x}_j(t) - \hat{x}_i(t)\|) \quad (13)$$

式 (11) は、 t フレームの投影点 $\check{p}_i(t)$ と予測点 $\hat{p}_i(t)$ との距離 r_j に係数 ε をかけた値を半径として、投影点と予測点を中心にそれぞれ円を描いた場合に、その領域内に存在する検出点を特徴点マッチングの候補点にすることを意味している。さらに、投影点 $\check{p}_i(t)$ および予測点 $\hat{p}_i(t)$ 自身も特徴点マッチングの候補点とする。図 2 に、特徴点マッチングの候補点を示す。図中では、各特徴点を色違いの十字で示している。図 2 左に、投影点の説明を示す。同左上は、時刻 $t-1$ フレームの画像とその画像における追跡点 $p_i(t-1)$ (黄色の十字) を示す。この追跡点を、同左下に示す時刻 t フレームの画像に投影した点が、同図に緑

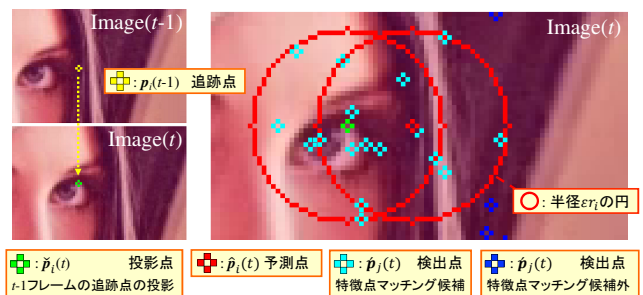


図 2 特徴点マッチング候補検出点の絞込み

Fig. 2 Filtering of detected feature point for the corresponding feature point.

表 3 特徴点マッチングによる Mean-Shift 探索の初期値

Table 3 Initial value of Mean-Shift search.

特徴点マッチング結果	検出点	予測点	投影点
画像座標値	$\hat{x}_j(t)$	$\hat{x}_i(t)$	$\check{x}_i(t)$
スケール値		$s_i(t-1)$	
特徴ベクトル		$v_i(t-1)$	

色の十字で示した投影点 $\check{p}_i(t)$ である。図 2 右では、特徴点マッチングの候補点について説明する。特徴点マッチングの候補点は、投影点と赤色の十字で示した予測点を中心とした半径 εr_i の赤色の円内に存在する特徴点である。半径 εr_i は、投影点と予測点との距離に ε を掛けた値である。この円内に存在する検出点を水色の十字で示し、円外に存在する検出点を青色の十字で示した。以降の処理で、円内に存在する全ての特徴点と $t-1$ フレームの追跡点との特徴点マッチングを行う。

3.2.4 特徴点マッチングと Mean-Shift 探索の初期値

特徴点マッチングは、特徴ベクトル間ユークリッド距離 $len(v_1, v_2)$ に基づき、候補点と追跡点 $p_i(t-1)$ に対して行う。特徴ベクトル間ユークリッド距離は、式 (14) とする。

$$len(v_1, v_2) = \sqrt{\sum_{k=0}^{127} (v_{1,k} - v_{2,k})^2} \quad (14)$$

式 (14) より求めた各候補点の特徴ベクトル間距離のうち最小の値を示す候補点をマッチング点とする。

特徴点マッチングの結果に基づき Mean-Shift 探索の初期値を決定する。Mean-Shift 探索の初期値は、マッチングした候補点の画像座標値と $t-1$ フレームの追跡点 $x_i(t-1)$ のスケール値 $s_i(t-1)$ と特徴ベクトル $v_i(t-1)$ を持つ特徴点とする。表 3 に、特徴点マッチングの結果に応じた Mean-Shift 探索の初期値をまとめる。

図 3 に、 $t-1$ フレームの追跡点 $v_i(t-1)$ が持つ特徴ベクトルと t フレームの画像の各画素における特徴ベクトルとの類似度をグレースケールで示した。図 3 の場合、投影点の位置から Mean-Shift 探索を開始すると、図 3 の中央に位置する黄色い十字で示した正解特徴点の画像座標値 $x_i(t)$ に収束したいにもかかわらず、探索領域内により類似度の小さな塊が存在するためそちらに収束し局所解に

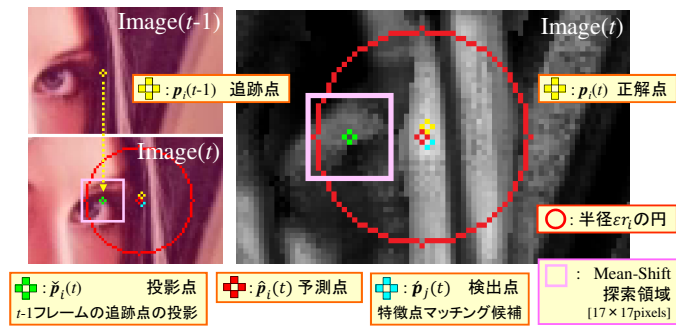


図 3 追跡点と各画素との特徴ベクトル間類似度分布

Fig. 3 Similarity distribution between the feature vector.

陥ってしまう。これに対して、対応点の画像座標値 $\hat{x}_j(t)$ や予測点の画像座標値 $\hat{x}_i(t)$ から Mean-Shift 探索を開始すれば、正解特徴点の画像座標値に収束できる。

3.3 Mean-Shift 探索による局所的特徴点追跡

都築ら [11] は、局所特徴量の 1 つである SIFT 特徴に基づき Mean-Shift を適用することで、局所領域において特徴点の詳細な追跡手法を提案している。Step3 は、彼らの手法を参考にする。都築らは、画像空間とスケール空間の両方に対して Mean-Shift を適用するが、我々は、画像空間への Mean-Shift 探索のみ行うものとする。以下、その理由について述べ、具体的な手法を説明する。

3.3.1 スポーツ映像とスケール方向の Mean-Shift 探索

SURF 検出器は、テクスチャの少ない領域において、検出する特徴点の密度が低くなる。また、検出した特徴点は大きなスケール値を持つ傾向がある。これは、テクスチャの少ない領域では、狭い範囲の画素から十分な特徴を得られないためである。このような特徴点は、追跡対象領域だけでなく背景領域を多く含む特徴点となる。したがって、スケール空間の Mean-Shift 探索を行うと、スケール値が大きくなる場合に、背景の影響が強くなり追跡に失敗する。

我々は、スポーツ映像におけるスポーツ選手の追跡を最終的な目標としている。実際に撮影されたスポーツ映像の多くは、遠方に設置したカメラで、プレーフィールドの大部分を撮影することで、試合全体を把握できるようにしている。この場合、サッカーを撮影した映像のように、撮影された選手の大きさが画像サイズに比べて小さい場合や、テクスチャの少ないユニフォームを着用している場合は、大きなスケール値を持つ特徴点が算出される傾向にある。また、連続フレーム間での選手のスケール方向に対する変化も小さくなる。

そこで、本研究では、スポーツ映像における選手の追跡において、選手のスケール変化は小さいという仮定を置き、スケール方向への Mean-Shift 探索を行わないことで、スケール値の大きな特徴点の追跡失敗を軽減する。

3.3.2 画像空間の Mean-Shift 探索

Mean-Shift 探索では、各追跡点を持つ 128 次元の特徴ベ

クトルとスケール値を参照データ $\tilde{\mathbf{v}}, \tilde{s}$ として利用し、追跡点の画像空間における変化量 $\Delta \mathbf{x} = (\Delta x, \Delta y)$ を算出する。そして、算出した変化量に基づき、追跡点の画像座標値 \mathbf{x} を更新する。

初めに、追跡点の画像座標値 $\tilde{\mathbf{x}}$ を中心とした周辺画素 $\mathbf{x}_l (l = 0, \dots, N_{loc})$ の各特徴ベクトル \mathbf{v}_l を算出する。この時、SURF の特徴ベクトルを算出するために必要なスケール値は、追跡点の参照用スケール \tilde{s} を用いる。

次に、追跡点の参照用ベクトル $\tilde{\mathbf{v}}$ と求めた各特徴ベクトルより、重み $w(\mathbf{x}_l, \tilde{s})$ を算出する。

$$w(\mathbf{x}, s) = \exp\left(-\frac{d(\mathbf{x}, s)^2}{2\sigma_d^2}\right) \quad (15)$$

ここで、 σ_d は、特徴ベクトル間の SURF の類似度を重みに変換する関数の変数である。また、参照用ベクトル $\tilde{\mathbf{v}}$ と各特徴ベクトル \mathbf{v}_l との特徴ベクトル間ユークリッド距離 $d(\mathbf{x}_l, \tilde{s})$ は、式 (16) である。

$$\begin{aligned} d(\mathbf{x}_l, \tilde{s}) &= \|\text{SURF}(\mathbf{x}_l, \tilde{s}) - \tilde{\mathbf{v}}\| \\ &= \|\mathbf{v}_l - \tilde{\mathbf{v}}\| \\ &= \sqrt{\sum_{k=0}^{127} (v_{l,k} - \tilde{v}_k)^2} \end{aligned} \quad (16)$$

現在の追跡点中心に重みを置いて探索するため、式 (18) のガウスカーネル関数と求めた重み $w(\mathbf{x}_l, \tilde{s})$ を用いて、画像空間における追跡点の移動量 $\Delta \mathbf{x}$ を式 (17) より求める。

$$\Delta \mathbf{x} = \frac{\sum_{l=0}^{N_{loc}} K_{loc}(\mathbf{x}_l - \tilde{\mathbf{x}}, \sigma_{loc}) w(\mathbf{x}_l, \tilde{s}) (\mathbf{x}_l - \tilde{\mathbf{x}})}{\sum_{l=0}^{N_{loc}} K_{loc}(\mathbf{x}_l - \tilde{\mathbf{x}}, \sigma_{loc}) w(\mathbf{x}_l, \tilde{s})} \quad (17)$$

$$K_{loc}(\mathbf{x}, \sigma_{loc}) = \exp\left(-\frac{x^2 + y^2}{2\sigma_{loc}^2}\right) \quad (18)$$

最後に、追跡点の画像座標値を $\mathbf{x}' = \mathbf{x} + \Delta \mathbf{x}$ に応じて平行移動させる。上記の処理を、 $\|\Delta \mathbf{x}\| < \epsilon_{loc}$ の収束条件を満たすまで反復する。

4. シミュレーションによる追跡精度評価実験

特徴点の移動変化が大きな場合において、提案手法が精度良く特徴点を追跡できることを確かめるために、追跡対象の移動変化をシミュレートした画像セットを作成し特徴点の追跡精度評価を行った。本実験は、都築ら [11] の評価手法に則して行った。

4.1 画像合成による実験用データセット

Lena の画像 (320 × 320 画素) を追跡対象画像とし、黒地の背景画像に追跡対象画像を重ね合わせることで連続した 50 枚の画像セットを作成した。黒地背景の利用は、特徴点追跡において背景の影響を受けないようにし、特徴点追跡手法の性能に着目するためである。重ね合わせる追跡対象画像を移動変化させることで、画像上における特徴点

表 4 追跡対象画像の変化量

Table 4 Control parameters of spatial variations.

Combination		1	2	3	4	5
A	平行移動 [画素]	3	6	9	12	15
B	回転 [°]	3	6	9	12	15
C	拡大 [画素]	1.5	3	4.5	6	7.5
D	平行移動 [画素]	3	6	9	12	15
	回転 [°]	3	6	9	12	15
E	平行移動 [画素]	3	6	9	12	15
	拡大 [画素]	1.5	3	4.5	6	7.5
F	回転 [°]	3	6	9	12	15
	拡大 [画素]	3	6	9	12	15
G	平行移動 [画素]	3	6	9	12	15
	回転 [°]	3	6	9	12	15
	拡大 [画素]	1.5	3	4.5	6	7.5
H	sin 関数による平行移動の速度変化					

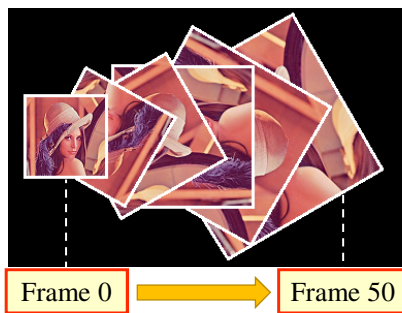


図 4 追跡対象画像の平行移動, 回転, 拡大の例

Fig. 4 Example of image sequences including translation, rotation and scale-up changes

の移動変化を表現した。具体的には, 平行移動と回転, 拡大の3種類の移動変化を組み合わせた7種類と sin 関数を用いた速度変化の全部で8種類のシミュレーションを行った。また, 各組合せの移動変化において, 追跡対象の移動速度を5段階変化させた。したがって, 全部で40種類の画像セットを用意した(表4)。表4中の見方として, 例えば, D-1は, 追跡対象画像が連続フレーム間で3画素平行移動し, 3°回転する画像セットを表している。

図4に, Lenaの画像を平行移動, 回転, 拡大変化させたG-5のシミュレーション画像セットの例を示す。この例では, 複数フレームにわたるLenaの移動変化と位置を重ね合わせて1枚の画像として表示しており, データセットに複数枚のLenaの画像が存在するわけではない。Lenaの画像が0フレーム目に左端の位置にあり, フレームが経過する(0フレームから50フレーム)につれて右端方向へ時計回りに拡大しながら平行移動していく様子を示している。

4.2 特徴点追跡精度評価方法

追跡精度の評価には, 特徴点の追跡誤差平均を用いた。誤差平均は低い値を示す方がよい。誤差の計算に必要な特

徴点の正解画像座標値は, 画像セット作成時に使用した変換行列を用いて, 初期画像から検出した特徴点の画像座標値を変換して求めた。比較対象として, Mean-Shift 探索(MS)[11], Kalman-filterの予測を用いた Mean-Shift 探索(KFMS)[8], Kalman-filterの予測を用いた特徴点マッチング(KFFPM)を採用した。また, 提案手法および比較対象において特徴点マッチングの ε を0.8, Mean-Shiftの探索幅を 17×17 画素とした。

4.3 実験結果と考察

図5に特徴点の移動量に対する各手法の特徴点追跡精度を示す。特徴点の移動量とは, ある特徴点について, $t-1$ フレームの正解画像座標値と t フレームの正解画像座標値との画像ユークリッド距離のことである。これは, 理想的な特徴点の移動距離である。平行移動の追跡精度(図5(a))では, 特徴点の移動距離が2画素以下, 5画素, 8画素, 11画素, 14画素となるシミュレーション結果がないため0の値を示している。また, 拡大の追跡精度(図5(c))でも, シミュレーションの設定上, 特徴点の移動距離が8画素よりも大きくなるシミュレーション結果が存在しない。

全手法において, 特徴点の移動距離が大きくなるにつれて, 追跡誤差平均が大きくなっており, 特徴点の追跡が困難になることがわかる。

MSは, 回転変化(図5(b))や拡大変化(図5(c))などの結果において, 特徴点の移動距離が小さいときに精度良く特徴点を追跡できていることを確認できる。しかし, 全ての結果において, 特徴点の移動距離が大きくなると特徴点を追跡することができなくなる。

KFMSは, Mean-Shift探索の初期値をKalman-filterの予測値とする手法である。これにより, MSに比べて特徴点の移動距離が大きい場合に特徴点を追跡できるようになっている。しかし, 平行移動, 回転, 拡大を組み合わせたシミュレーション(図5(g))では, Kalman-filterの予測が失敗するため追跡誤差平均が大きな値を示している。また, 特徴点の移動距離が小さな場合にも初期値を変更するため, 追跡誤差平均が大きな値となっている。

KFFPMでは, 特徴点の移動距離が大きな場合にも特徴点の追跡ができていない。しかし, 特徴点の移動距離が小さい場合において, 正しく追跡できたMSと比較すると, 回転の追跡精度(図5(b))をはじめKFFPMの方が大きな値を示している。したがって, KFFPMは, 詳細な追跡ができていないといえる。

提案手法は, 全比較手法よりも, 平行移動の追跡結果以外の全てのシミュレーションにおいて, 追跡誤差平均を小さく保ちつつ追跡できている。平行移動の追跡精度(図5(a))では, KFFPMが最もよい値を示しているが, これは, Kalman-filterの予測が正確に行われた結果である。提案手法は, 特徴点の移動が小さい場合に, Mean-Shift探

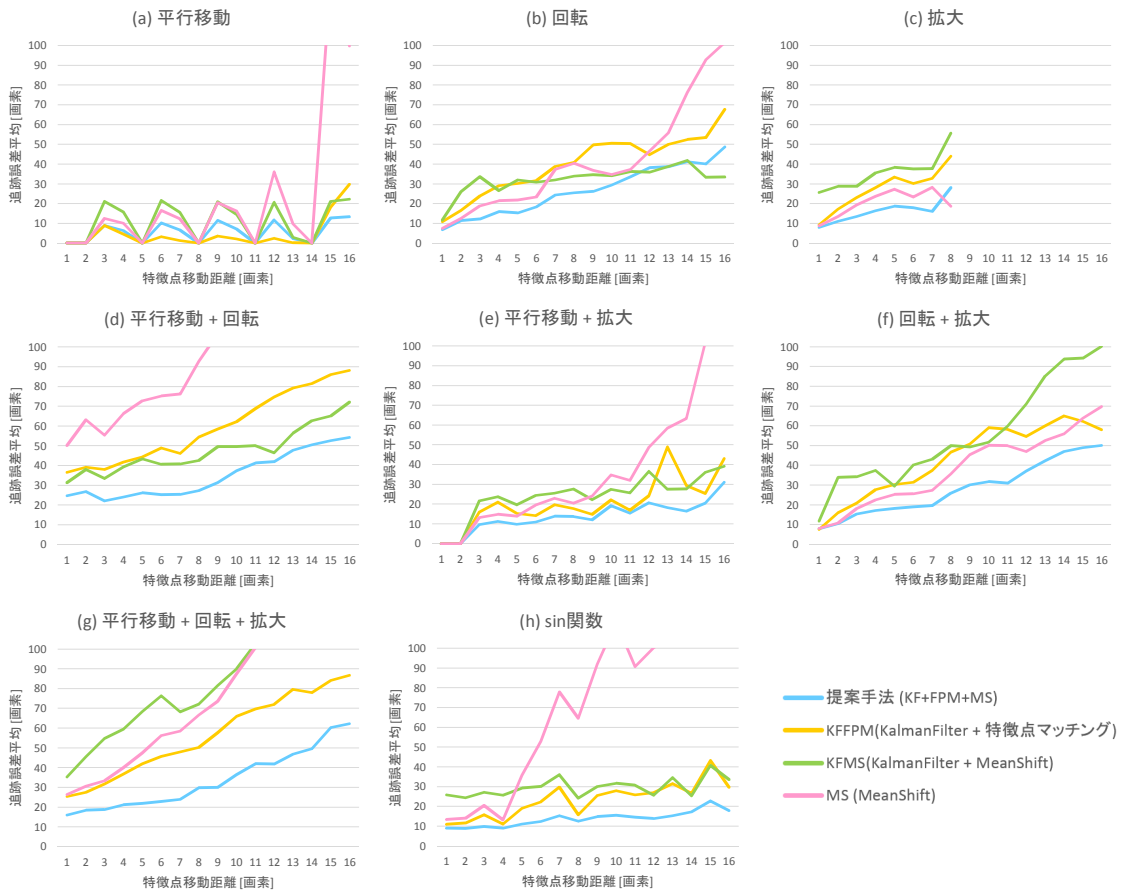


図 5 シミュレーションデータセットの各手法別特徴点追跡誤差平均
Fig. 5 Tracking accuracies of each method.

索を行うため、Mean-Shift と同様に精度のよい追跡ができています。また、特徴点の移動距離が大きな場合に、特徴点マッチングするため、追跡できるようになっています。さらに、特徴点マッチング後のマッチング点から Mean-Shift 探索を行うため、特徴点マッチングの手法よりも精度よい追跡ができています。

5. サッカー映像に対する追跡実験

サッカーを撮影した映像 (30fps) に対して、提案手法、Mean-Shift 探索 (MS)[11]、Kalman-filter の予測を用いた特徴点マッチング (KFFPM) を、それぞれ適用した。シミュレーション実験と同様に、提案手法および比較対象において特徴点マッチングの ϵ を 0.8、Mean-Shift の探索幅を 17×17 画素とした。

図 6 に特徴点追跡結果を示す。図 6 は、左列から (a) 提案手法、(b) Mean-Shift 探索 (MS)、(c) Kalman-filter の予測に基づく特徴点マッチング (KFFPM) の追跡結果である。上段から下段に向かい、0 フレーム目から 300 フレーム目まで、100 フレーム置きでの追跡結果を示した。また、(a) の追跡結果には、追跡対象である選手を矩形で囲み、選手の移動遷移を矢印で示した。

図 6 の追跡結果より考察する。シミュレーション実験では、KFFPM が提案手法について良い精度を上げていた。

しかし、黄色や赤色、紫色の矩形で示した選手は、300 フレーム目では 1 つの特徴点も追跡できていない。これは、実データには、シミュレーションに存在しない追跡対象の 3 次元的な変化があり、これにより抽出される特徴量が変化したため、追跡に失敗したと考えられる。

これに対し、提案手法と MS の結果は、300 フレームまでの間、ほぼ同程度で選手を追跡できている。連続フレーム間での追跡対象の 3 次元的な変化が微小な場合、画像から算出される局所特徴量変化も微小である。したがって、Mean-Shift が極値探索問題の解法であるので、追跡点周辺の画素から追跡点の持つ特徴量と類似した特徴量を持つ画素を検出できるためである。

6. おわりに

本稿では、画像上における特徴点の移動変化が大きな場合においても、精度良く特徴点を追跡する手法として、大域的な特徴点追跡である特徴点マッチングと局所的な詳細特徴点追跡手法である Mean-Shift 探索を組み合わせる手法を提案した。提案手法の有用性を示すために、追跡対象画像を背景画像上で移動変化させてシミュレーション画像を作成して追跡実験を行った。シミュレーション実験の結果、提案手法は 3 つの比較手法に比べて、追跡誤差を小さく保ちながら追跡できた。これは、特徴点マッチングによ

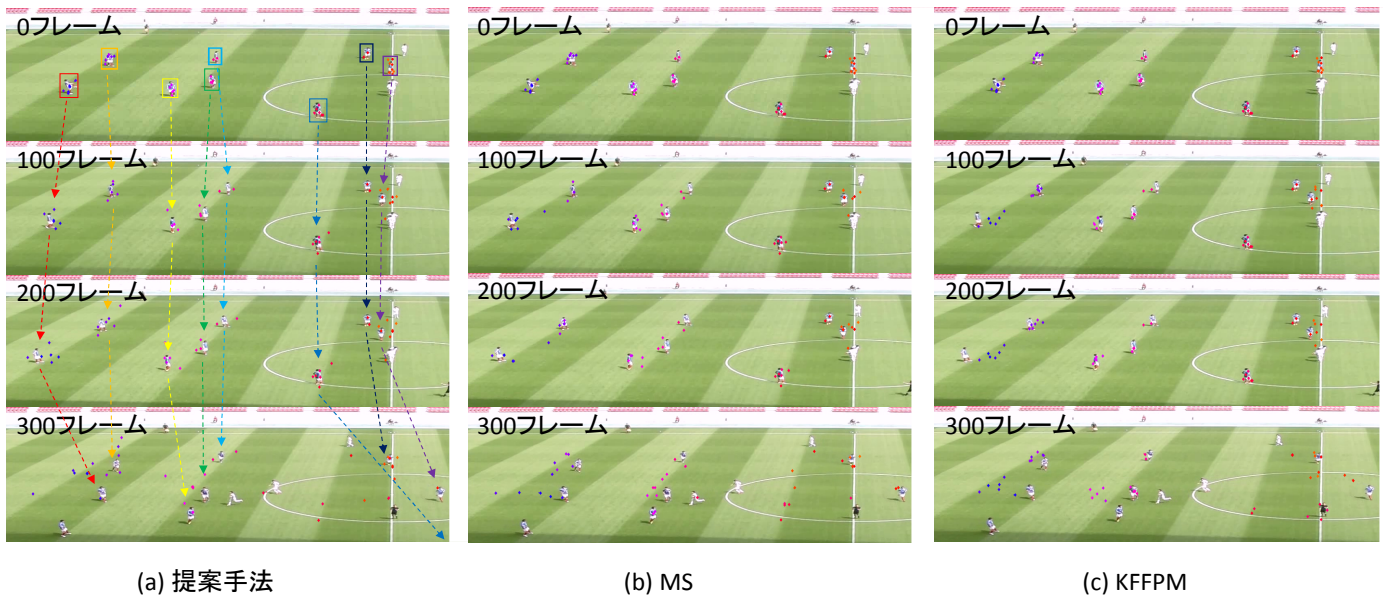


図 6 サッカー映像に対する追跡結果
Fig. 6 Tracking result of feature point for soccer movies.

り広域に特徴点を追跡した後に、マッチング点の画像座標値から Mean-Shift 探索することにより、局所解に陥ることなく詳細な特徴点追跡ができたためである。また、サッカーを撮影した映像に対して、特徴点の追跡実験を行った。その結果、提案手法と Mean-Shift 探索手法は、300 フレーム目まで追跡できた。追跡に失敗する特徴点もあり、その原因として、背景の影響を強く受ける特徴点の存在もあるが、サッカー選手の3次元的な変化により、算出される特徴量が変化することも要因であった。

今後の課題として、特徴点を群として扱うことで、人物のような非剛体物体の正確な追跡を目指す。具体的には、誤追跡した特徴点を削除する処理と新たな特徴点を追加する処理を追加する。これにより、追跡対象の3次元的な変化に対して、逐次的に追跡する特徴点を更新することができ、追跡可能になると考える。

謝辞 本研究の成果の一部は、総務省の戦略的情報通信研究開発推進制度 (SCOPE) の補助による。

参考文献

- [1] D. Comaniciu and P. Meer, "Mean Shift Analysis and Applications," Proc. IEEE International Conf. on Computer Vision, vol. 2, pp. 1197–1203, Kerkyra, Sep. 1999.
- [2] R. Kalman, "A new approach to linear filtering and prediction problems," Trans. ASME - Journal of Basic Engineering, pp. 35–45, 1960.
- [3] D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," Proc. the Seventh IEEE International Conf. on Computer Vision, vol. 2, pp. 1150–1157, Kerkyra, Sept. 1990.
- [4] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," Proc. 9th European Conf. on Computer Vision, vol. 110, no. 3, pp. 404–417, Graz, Austria, May 2006.
- [5] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time

- Tracking of Non-Rigid Objects using Mean Shift," Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR' 00), vol. 2, pp. 142–149, Hilton Head Island, South Carolina, Jun 2000.
- [6] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-Based Object Tracking," Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR' 00), vol. 2, pp. 142–149, Hilton Head Island, South Carolina, Jun 2000.
- [7] R. T. Collins, "Mean-shift Blob Tracking through Scale space," Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, no. II, pp. 234–240, June 2003.
- [8] D. Comaniciu and V. Ramesh, "Mean Shift and Optimal Prediction for Efficient Object Tracking," Proc. International Conference on Image Processing, Vancouver, vol. 3, pp. 70–73, BC, Sep. 2000.
- [9] C. Shan, T. Tan, and Y. Wei, "Real-time hand tracking using a mean shift embedded particle filter," Pattern Recognition, vol. 40, no. 7, pp. 1958–1970, 2007.
- [10] K. Deguchi, O. Kawanaka, and T. Okatani, "Object tracking by the mean-shift of regional color distribution combined with the particle-filter algorithm," International Conf. on Pattern Recognition, vol. 3, pp. 506–509, 2004.
- [11] 都築勇司, 藤吉弘巨, 金出武雄, "SIFT 特徴量に基づく Mean-Shift 探索による特徴点追跡," IPSJ Journal, vol. 49, no. SIG6(CVIM 20), pp. 35–45, 2008.
- [12] H. Zhou, Y. Yuan, C. Shi, "Object tracking using SIFT features and mean shift," Computer Vision and Image Understanding, vol. 113, no. 3, pp. 345–352, 2009.
- [13] M. Brown and D. G. Lowe, "Recognising Panoramas," Proc. International Conference on Computer Vision, pp. 1218–1225, 2003.
- [14] B. Zitova and J. Flusser, "Image registration methods: a survey," Image and Vision Computing 21, vol. 21, no. 11, pp. 977–1000, Oct. 2003.