仮説検証に基づく自然環境下での複数歩行者追跡

澤田裕介† 尺長 健†

† 岡山大学大学院 自然科学研究科
700–8530 岡山市北区津島中 3–1–1
E-mail: †{sawada,shaku}@chino.cs.okayama-u.ac.jp

あらまし 動画像中の歩行者追跡は，従来，実時間処理を前提とする研究が広く行われてきた．しかし，複雑な照明条件下で多様な状況が含まれる動画像を対象とした場合，実時間処理には限界があると考えられる．そこで，本稿では，より柔軟な処理系を構成することで，複雑な状況にも対応できる歩行者追跡系の作成を目指す．具体的には，追跡処理において生じる誤追跡などの問題に対処するため，仮説検証を利用した追跡法を検討する．すなわち，背反する複数の仮説に基づく追跡結果を比較検証することで，複雑な状況に対応できる追跡系を目指す．また，通常の順方向追跡だけではなく，状況に応じて逆方向追跡を行うことで，順方向追跡だけでは盲点となる状況に対応する．最後に，実験により，仮説検証追跡の有効性を確かめる．

キーワード 複数歩行者追跡，仮説検証追跡，地平面座標，背景差分，逆方向追跡

1. はじめに

複数歩行者追跡は，監視カメラなどのセキュリティシステムにおいて重要な技術であり，広く研究が行われてきた．従来，我々の研究室では，多人物同時追跡を題材として 2 つの研究を発表している．まず，佐竹-尺長 [1], [2] は，階層的注視点制御の例題として複数人物同時追跡を取扱い，歩行者間の隠れを状態遷移させることで，数名の歩行者を対象とした追跡を実現している．また，佐竹-尺長 [3] は，疎テンプレート追跡 [4] を多人物同時追跡に適用し，部分テンプレートを協調させる方法を提案している．これらの研究では，樹木等によって移動人物に複雑な影が生じない状況で，人物間の隠れに対処する方法を論じている．

一方，セキュリティの観点からは，屋外環境で太陽の位置や天候によらず，動画像中の各人物の動きが激しい場合にも必要があり，様々な状況に対処できる柔軟な歩行者追跡を目指した研究が活発化している．例えば，隠れ処理については，手前の歩行者から順に処理する方法 [5] や，不自然な重なりを検出して追跡結果を修正する方法 [6] が提案されている．複数物体を対象とした追跡においては，隠れの取扱いは必須であり，様々な手法が対象や環境に応じて試みられてきた [7], [8], [9], [10], [11].

これらの中には，仮説検証を含むものが見受けられるが，オンライン追跡の中での利用を目指したものが多く，十分な追跡性能の改善は果たせていない．一方で，動画像による人物動作の解析においては，オフラインで性能向上を目指すことも重要であり，そのためには，仮説生成方法や検証方法に関する検討が必要と考えられる．そこで，本稿では，オンライン追跡の制約を緩めた場合を想定し，追跡が難しいフレームを繰り返し解析することにより追跡性能の向上を目指す．

2. 歩行者追跡の基本処理

本稿では，歩行者追跡を検出処理と追跡処理の組み合わせで実現する．検出処理では，入力画像から歩行者の領域を検出し，テンプレートの作成と初期位置の推定を行う．歩行者領域の検出には背景差分を用いる．一方，追跡処理では，パーティクルフィルタ [12] を用いて歩行者位置を逐次推定することにより追跡を行う．このとき，複数歩行者を同時に追跡している場合には，歩行者間の重なりを考慮して追跡を行う．以下では，これらの基本処理について説明する．

2.1 地平面座標

本稿では，歩行者は 3 次元空間中の一平面上のみを移動するものとして追跡を行う．以降ではこの平面を地平面と呼び，この平面を基準とした 3 次元空間を地平面空間と呼ぶ．画像座標を $\mathbf{m} = (u, v)^\top$，地平面座標を $\mathbf{M} = (x, y, z)^\top$ とおくと，地平面空間から画像平面への変換は次式により表される．

$$\tilde{\mathbf{m}} = \mathbf{P}\tilde{\mathbf{M}} \tag{1}$$

ここで，$\tilde{\mathbf{M}} = (x, y, z, 1)^\top$，$\tilde{\mathbf{m}} = (u, v, 1)^\top$ は $\mathbf{M}$，$\mathbf{m}$ の同次座標表現であり，$\mathbf{P}$ は地平面座標から画像座標への変換行列である．変換行列 $\mathbf{P}$ は，地平面座標と画像座標の対応点の集合から線形解法によって予め求める．

逆に，画像座標から地平面座標へ変換する場合，地平面に接する 2 次元位置 $\mathbf{M}_G = (x, z)^\top$ へ変換することになる．これは次式によって表される．

$$\tilde{\mathbf{M}}_G = \mathbf{H}\tilde{\mathbf{m}} \tag{2}$$

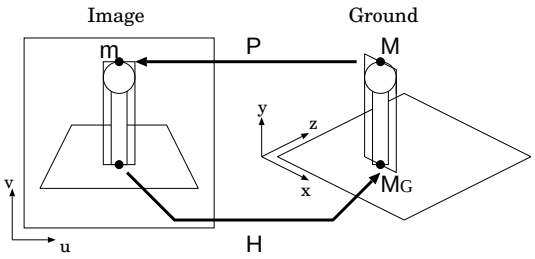


図 1　画像-地平面間の変換

ここで，$\tilde{\mathbf{M}}_G = (x, z, 1)^\top$ は $\mathbf{M}_G$ の同次座標表現であり，$\mathbf{H}$ は画像平面から地平面への変換行列である．変換行列 $\mathbf{H}$ は $\mathbf{P}$ と同様に線形解法によって求められる．地平面空間と画像平面との関係を図 1 に示す．

以下では，歩行者の位置を足元が存在する地平面上の 2 次元座標 $\mathbf{l} = [x, z]$ として表す．

歩行者の追跡を行うため，2 種類のテンプレートを作成する．一方は歩行者の画素値を表すテンプレートであり，これを $\mathbf{X}_T$ とする．もう一方は $\mathbf{X}_T$ の中で有効である領域を 1 で表す 2 値のテンプレートであり，これを $\mathbf{B}_T$ とする．これらのテンプレートは，背景差分によって得られた画像平面上の歩行者の領域を，地平面空間上へ変換することで求める．

2.2 検出処理

歩行者の領域を検出するため，まず入力画像と背景画像との差分を求め，差分の大きさにより 2 値画像を作成する．すなわち，入力画像の各画素値が背景か背景でないかを表す 2 値画像を作成する．この画像を背景差分 2 値画像と呼ぶ．そして，背景差分 2 値画像の中の各連続領域で歩行者の大きさに近いものがあれば歩行者領域とする．ここで，背景画像は，入力画像の各画素の最頻値から作成する．

n 画素からなる画像の集合 (フレーム数 F) が入力画像であるとき，f 番目のフレームの入力画像を $\mathbf{X}_f (1 \leqq f \leqq F)$，その i 番目の画素値を $X_{fi} (1 \leqq i \leqq n)$ とする．このとき，背景画像 $\mathbf{X}_b$ の各画素値 X_{bi} を次式により求める．

$$X_{bi} = \arg\max_{l \leqq c \leqq C-l} \sum_{f=1}^{F} N(X_{fi}, c) \tag{3}$$

$$N(X, c) = \begin{cases} 1 & c - l \leqq X \leqq c + l \\ 0 & otherwise \end{cases} \tag{4}$$

ここで，C は画素値の最大値，l は画素値を同値とみなす範囲を示すパラメータであり，本稿では $l = 2$ とする．

このようにして求められる背景画像 $\mathbf{X}_b$ を用いて，各入力画像 $\mathbf{X}_f$ の背景差分 2 値画像を作成する．f 番目のフレームの背景差分 2 値画像を $\mathbf{B}_f$ としたとき，$\mathbf{B}_f$ の各要素 B_{fi} は入力画像の画素値 X_{fi} と背景画像の画素値 X_{bi} との残差を 2 値化することにより求める．



入力画像 $\mathbf{X}_f$　　背景差分 2 値画像 $\mathbf{B}_f$

図 2　背景差分の例

$$B_{fi} = \begin{cases} 1 & |X_{fi} - X_{bi}| \geq \theta \\ 0 & otherwise \end{cases} \tag{5}$$

ここで，θ は 2 値化の閾値を表し，本稿では $\theta = 50$ とした．背景差分の例を図 2 に示す．

$\mathbf{B}_f$ 内の連続する領域を有効領域と呼ぶ．有効領域の内，歩行者の大きさに近いものを歩行者領域として検出する (図中，白で表示)．このため，有効領域を画像平面から地平面座標に変換し，変換後の高さと歩行者の基準身長とを比較することで判定する．これにより，画像中の位置の違いによる歩行者のスケール変化に対応した歩行者検出を行うことができる．有効領域が歩行者の領域として検出されたら，テンプレートの作成と初期位置の決定を行う．

2.3 追跡処理

追跡処理では，パーティクルフィルタ [12] とテンプレートマッチングを組み合わせることで位置の推定を逐次行う [4]．パーティクルフィルタ追跡では姿勢の推定を行うが，本稿では姿勢空間は地平面空間であるため，2 次元位置と等価である．また，テンプレートは地平面座標を基準として作成されているので，画像中のスケール変化に対応したマッチングを行うことができる．このとき，歩行者間に重なりが生じている場合，重なりを考慮しテンプレートマッチングを行う必要がある [5]．

2.3.1 パーティクルフィルタ追跡

本稿では，松原-尺長 [4] に従い，姿勢空間でパーティクル集合を伝播させていくことにより，姿勢追跡を行う．即ち，フレーム毎に L 個の解候補である位置 $\mathbf{l}_l (1 \leqq l \leqq L)$ を生成する．これらの位置は，前のフレームで求まった解候補から評価値の高かった位置を複数選び，それらにガウスノイズを付加して拡散することで得られる．生成された各位置で，そのフレームの入力画像を評価することにより，評価値を重みとした位置の重み付き平均として推定位置が求められる．この処理を逐次行うことにより，追跡処理を実現する．

2.3.2 マッチングの評価

パーティクルフィルタ追跡における各位置の評価方法を述べる．従来，疎テンプレートマッチングを用いた追跡系を松原-尺長 [4] が提案しているが，照明条件が場所によって異なる野外で変形の大きい歩行者を追跡するには疎テンプレートは適していないと思われる．そこで，

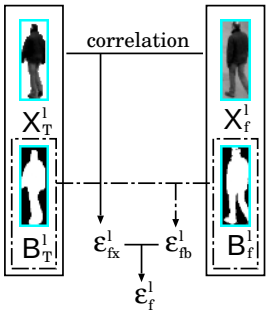


図 3　位置 l の評価方法

本稿では，背景差分によって得られる $\mathbf{X}_T$，$\mathbf{B}_T$ を用いた照明変動や歩行動作を考慮したテンプレートマッチングを行う．

解候補の位置 $\mathbf{l}_l$ を用いて歩行者テンプレート $\mathbf{X}_T, \mathbf{B}_T$ を画像平面上へ変換した結果を $\mathbf{X}^l_T, \mathbf{B}^l_T$ と表す．このとき，$\mathbf{X}^l_T, \mathbf{B}^l_T$ と入力画像との相関を算出し，その結果から位置の評価値 ϵ^l_f を求める．

$\mathbf{X}^l_T, \mathbf{B}^l_T$ に対応する入力画像の領域を $\mathbf{X}^l_f (\subset \mathbf{X}_f)$，その有効領域を $\mathbf{B}^l_f (\subset \mathbf{B}_f)$ とする．まず，画素値の相関 ϵ^l_{fx} をテンプレートの有効領域と入力画像の有効領域との共通部分のみを用いて求める．これは次式により表される．

$$\epsilon^l_{fx} = \frac{(\mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_f)^\top (\mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_T)}{\|\mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_f\| \|\mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_T\|} \tag{6}$$

ここで，$\mathbf{Q}^l_T$ は $\mathbf{B}^l_T$ の各要素を対角要素とする対角行列であり，同様に $\mathbf{Q}^l_f$ は $\mathbf{B}^l_f$ の各要素を対角要素とする対角行列を表す．

しかし，画素値の相関は有効領域の共通部分が少ない場合にも高い値が得られてしまうことがある．そのため，有効領域自体の相関を次式によって求める．

$$\epsilon^l_{fb} = \frac{\mathbf{B}^{l\top}_f \mathbf{B}^l_T}{\|\mathbf{B}^l_f\| \|\mathbf{B}^l_T\|} \tag{7}$$

位置が合っていれば $\epsilon^l_{fx}, \epsilon^l_{fb}$ は共に大きくなると考えられるため，これらの値の積を位置の評価値 ϵ^l_f として用いる．

$$\epsilon^l_f = \epsilon^l_{fx} \epsilon^l_{fb} \tag{8}$$

図 3 に，評価値の計算方法を示す．

2.3.3　隠れ領域の取扱い

複数の歩行者を同時に追跡するとき，人物間に重なりが生じることがある．そのような場合，隠された側の歩行者については，このままでは評価値を正確に求めることができない．正確な評価値を求めるためには，隠れが生じた領域を取り除くことが必要である．Bazzani ら [5] は手前にいる歩行者から順に追跡を行うことによって歩行者の領域を隠れ領域に追加する手法を用いており，これはテンプレートマッチングにおいても有効であると考えられる．

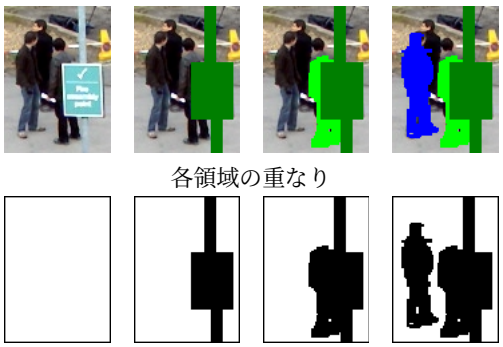


図 4　隠れ領域の更新

隠れ領域を表す 2 値の分布を $\mathbf{B}_O$ で表す．これは要素が 0 であれば隠れ領域を，1 であれば隠れがない領域であることを示す．各フレームの追跡処理では，隠れが存在しない状態を $\mathbf{B}_O$ の初期状態とし，カメラに近い (地平面上において手前にいる) 歩行者から順次追跡を行う．そして，位置が推定された歩行者の領域を適宜 $\mathbf{B}_O$ の隠れ領域に追加していくことで，手前の歩行者が原因となる隠れに対応することができる．また，隠れを起こす遮蔽物体が画像上にある場合には，予めその物体の地平面上における位置と領域を求め，初期状態に加えておくことで対応する．隠れ領域 $\mathbf{B}_O$ が更新される様子を図 4 に示す．

隠れ領域をマッチングの評価領域から除くように式 (6) (7) に変形を加えると，相関の算出は以下の式によって行われることになる．

$$\epsilon^l_{fx} = \frac{(\mathbf{Q}_O \mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_f)^\top (\mathbf{Q}_O \mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_T)}{\|\mathbf{Q}_O \mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_f\| \|\mathbf{Q}_O \mathbf{Q}^l_f \mathbf{Q}^l_T \mathbf{X}^l_T\|} \tag{9}$$

$$\epsilon^l_{fb} = \frac{(\mathbf{Q}_O \mathbf{B}^l_f)^\top (\mathbf{Q}_O \mathbf{B}^l_T)}{\|\mathbf{Q}_O \mathbf{B}^l_f\| \|\mathbf{Q}_O \mathbf{B}^l_T\|} \tag{10}$$

ここで，$\mathbf{Q}_O$ は $\mathbf{B}_O$ の各要素を対角要素とする対角行列である．これにより，隠れ領域によるマッチング性能の低下を防ぐことができる．

3.　仮説検証追跡と逆方向追跡

実時間処理で追跡が難しい事例として，複数の歩行者が接近や交差した場合など，単一のフレームのみでは解が一意に定まらない状況が挙げられる．これらの状況に対応するため，複数の仮説を用いた仮説検証追跡を提案する．これは複数の解候補が得られたときにそれぞれを仮説として保持しておき，仮説毎の追跡を行ったあとに信頼性の高い仮説を追跡結果とする手法である．また，追跡結果から過去のフレームの修正が必要であると判断された場合に，解析方向を順方向から逆方向へと変更する逆方向追跡を導入する．本稿では，仮説検証追跡と逆方向追跡を組み合わせた追跡系を構成することで，追跡精度の向上を目指す．

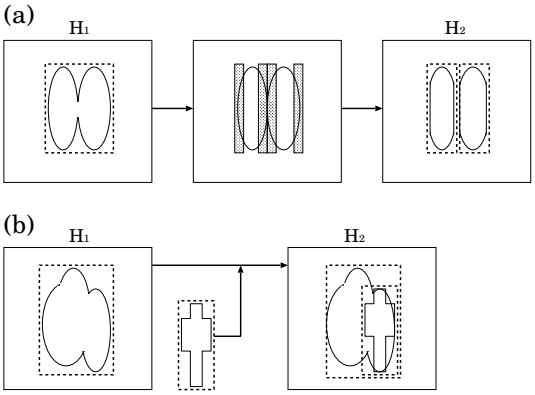


図 5　検出仮説の生成

3.1　仮説の生成

入力画像を解析した結果を一意に定めることが難しいと判断された場合には，それぞれを解析結果とした場合の仮説を生成する．このとき，各仮説を H_i (i は何番目に生成された仮説かを表す番号) と表す．本稿では，仮説として検出仮説と追跡仮説の 2 種類を扱う．検出仮説は，大きな有効領域が検出されたときに，その中から歩行者領域を切り出した場合と切り出さなかった場合を仮説として生成する．また，追跡仮説は，各歩行者の推定位置が一意に定まらなかった場合に，それぞれを解とした仮説として生成する．

3.1.1　検出仮説の生成

2.2 で示した検出処理によって，単独歩行者は比較的容易に検出できる．一方，重なりのある状態で登場した各歩行者は，それぞれが離れた状態になるまで検出することができない．そのため，有効領域が歩行者より大きいときには，その領域から歩行者の領域を切り出す処理が必要になる．本稿では，有効領域の分割は以下の処理によって行う．

- 輪郭分割

人が隣り合っている場合や影領域が強い場合には，それぞれの有効領域が結び付き，u 方向に大きくなってしまう場合が多い．しかし，このような有効領域内では，v 方向に歩行者の輪郭がかすかに出ている場合がある．そこで，有効領域内において v 方向の輪郭を求め，その位置から領域を左右に分割することで，歩行者の領域を切り出す．すなわち，有効領域の輪郭線の v 方向の連続性を解析することにより，v 方向の境界候補を検出する (詳細は略す)．隣り合っている人物の領域を輪郭分割によって切り分ける様子を図 5(a) に示す．ただし，輪郭分割では領域の切り分けによって，歩行者の形状が崩れる可能性がある．

- 標準テンプレートマッチング

複数の歩行者が重なっている領域では，歩行者に近い形状を持つ領域があると考えられる．そこで，予め歩行者の形状に近い標準テンプレートを用意し，有効領域とのテンプレートマッチングを行うことで，歩行者の領域を求める．標準テンプレートを用いて，有効領域から歩行者の領域を求める様子を図 5(b) に示す．求まった歩行者の領域を有効領域から取り除き，再び標準テンプレートマッチングを繰り返すことで，有効領域内の各歩行者の領域を解析することができる．

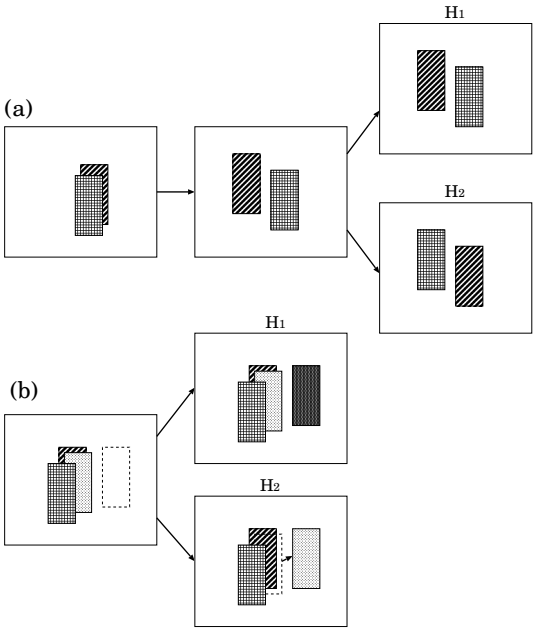


図 6　追跡仮説の生成

これらの切り出し処理は，重なりのない歩行者の検出と比べると精度が低くなるため，誤検出の原因となる可能性がある．そのため，一定のフレーム毎に切り出しを行った場合と切り出しを行わなかった場合の 2 つを検出仮説を生成し，何フレームか後に，どちらが妥当かの検証を行う．これらの処理は実時間で行うことは難しいが，実時間性を重視しない仮説検証追跡には有効である．

3.1.2　追跡仮説の生成

追跡中に複数の解候補が得られたとき，どちらが正しいかをそのフレームのみからでは判断できないような場合に追跡仮説を生成する．本稿では，以下のような条件において仮説の生成を行う．

- 前のフレームでは互いに重なりがあった 2 人の歩行者が，現フレームでは重なりがなくなった状態で推定された場合．
  - 仮説 1：2 人の歩行者の追跡結果をそのまま用いる．
  - 仮説 2：2 人の歩行者の追跡結果を入れ替える．
- 追跡が行われていない歩行者領域が存在し，その近くに複数の歩行者が重なりがある状態で追跡されている場合．
  - 仮説 1：歩行者領域を新たな歩行者として検出する．
  - 仮説 2：歩行者領域と近傍の各歩行者のテンプレートとの相関を求め，その中から相関の最も高い歩行者の位置を有効領域に割り当てる．

これらの条件によって仮説が生成される様子を図 6 に示す．図 6(a) では，それまで重なりのあった 2 人の歩行

者が離れることにより仮説が生成される．このとき，仮説 H_1 は追跡結果に基づいた仮説であり，仮説 H_2 は二人の追跡結果を入れ替えたものである．これらの仮説を検証することで，歩行者が交差したときに追跡結果が入れ替わる失敗を軽減できると考えられる．

また図 6(b) では，新たに検出された歩行者領域の近くに複数歩行者の重なりがあるため，仮説が生成される．このとき，仮説 H_1 では歩行者領域を新しい歩行者として検出する．一方，H_2 では重なりのある複数の歩行者の中から，有効領域との相関が最も高い歩行者を歩行者領域へ移動させて追跡を行う．これらの仮説を検証することで，歩行者の重複した追跡や検出による失敗を軽減できると考えられる．

3.2 仮説の検証

仮説が生成されると，各仮説毎に追跡を進めていく必要がある．このとき，各仮説について生成されてから追跡を行ったフレーム数を保持しておき，仮説検証を行う時期の判定に用いる．

具体的には，仮説検証追跡は以下のように行われる．図 7 に処理の進行例を示す．

（1） 仮説の生成：複数の解候補が得られたとき，それぞれを仮説 H_i として生成する (図 7(a))．

（2） 仮説の解析：仮説毎に追跡を行い，結果を時系列情報として保存する．この場合，無制限に追跡するのではなく，仮説が生成されてから一定フレーム数追跡を行った時点で追跡を終了する (図 7(b))．また，仮説に基づく追跡の際に新たな仮説が生成された場合には，追跡フレーム数が継承される．

（3） 仮説の検証：生成された全ての仮説に関する追跡が終了したとき，各仮説 H_i の評価値 E_i を解析結果を用いて求める．ここで，最初の分岐について，最も高い評価値に至る仮説が選択される (図 7(c))．

（4） 仮説の更新：この時点で，仮説検証によって選択された分岐は，最初の分岐のみである．そのため，選択された仮説についても，その後に分岐が発生している場合がある．この場合，その分岐を仮説生成位置として再度仮説の解析を行う (図 7(d))．

これらの仮説検証を繰り返し，全ての仮説を検証することにより追跡結果を得る (図 7(e))．

仮説の検証を行うためには，それまでのフレームの追跡結果を総合した評価が必要となる．本稿では，背景差分の信頼性が高いものであると仮定して，以下の値を仮説の評価値 E として用いる．

$$E = \frac{1}{F_E}\sum_{f=1}^{F_E}(w_1 E_x + w_2 E_d + w_3 E_o) \tag{11}$$

ただし，$w1$，w_2, w_3 は重みであり，$w_1 + w_2 + w_3 = 1$ である．また，E_x，E_d，E_o はそれぞれ，歩行者の画素値の相関の平均，有効領域の中で追跡結果と重なっている割合，歩行者間の重なりの少なさを表し，次のように定義される．

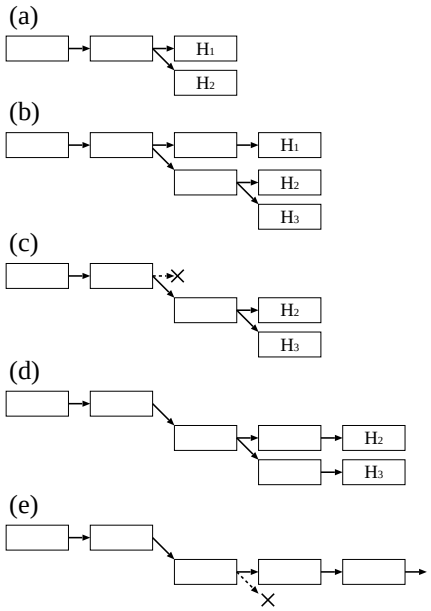


図 7　仮説検証の流れ

$$E_x = \frac{1}{N}\sum_{n=1}^{N}\epsilon_{fx}^{(n)} \tag{12}$$

$$E_d = \frac{\|\mathbf{B}_f \cap \mathbf{X}_T^1 \cap \ldots \cap \mathbf{X}_T^N\|}{\|\mathbf{B}_f\|} \tag{13}$$

$$E_o = \frac{1}{N}\sum_{n=1}^{N}\frac{\|\mathbf{B}_O^n\|}{\|\mathbf{X}_T^n\|} \tag{14}$$

$$\tag{15}$$

ここで，N は各フレームにおける追跡歩行者の人数，$\epsilon_{fx}^{(n)}$ は n 番目の歩行者の推定位置における評価値 (式 (6)) を示すものとし，$\mathbf{X}_T^n$，$\mathbf{B}_O^n$ はそれぞれ n 番目の歩行者の推定位置のテンプレート，およびそれと同じ範囲の領域を $\mathbf{B}_O$ から切り出したものである．また，F_E はその時点までに追跡が行われたフレームの総数を表す．なお，本稿では $w_1 = 0.6, w_2 = 0.2$ としている．

3.3 逆方向追跡処理

複雑な状況で追跡処理を進めると，不自然な位置から歩行者が検出されるなど，過去の追跡結果の修正が必要となる場合が出てくる．そのような場合，順方向の追跡処理を逆方向へと変更し，過去のフレームの結果を修正することで対応する．これを逆方向追跡と呼び，本稿では，いずれかの歩行者の位置推定結果が以下のような状態にである場合に，逆方向追跡処理を行う．

- 新たな歩行者を検出したとき

背景差分により新たな歩行者が検出されたときには，その歩行者の登場位置を求めるために逆方向追跡を行う．これにより，本来は画面に登場したときには歩行者間の重なりなどによって検出が難しい歩行者の位置を推定す

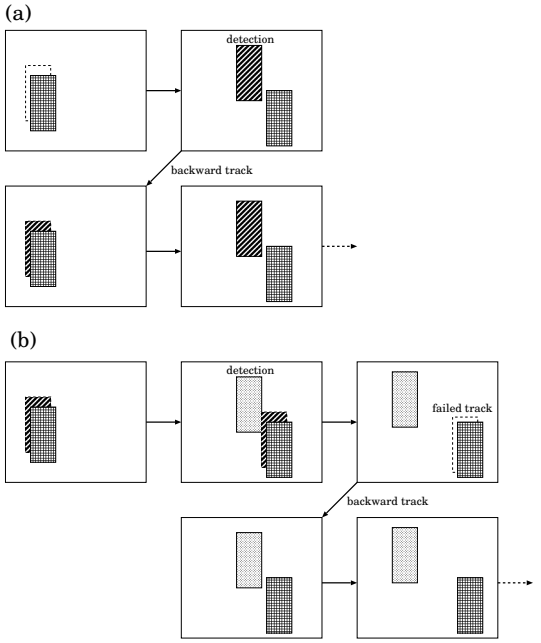


図 8　逆方向追跡の例

ることが可能となる．この処理では，検出された歩行者はパーティクルフィルタを用いて逆方向に追跡を行い，他の歩行者は過去に求まっている追跡結果をそのまま追跡結果とする．その人物の逆方向追跡が終了した時点で，追跡方向を順方向へと戻し，本来の追跡処理を行う．

検出による逆方向追跡の例を図 8(a) に示す．人物間の重なりが強いときには検出仮説を用いても歩行者検出を行うことは難しい．そのような場合には互いに離れたときに検出を行い，逆方向追跡を行うことで重なりが起こっていた過去のフレームの位置も求まる．

- 追跡に失敗して終了したとき

歩行者追跡では，位置推定結果と有効領域との重なりが極端に少なくなった場合などには，追跡に失敗したと判断して追跡処理を終了する．このような場合には，失敗が発生したと考えられるフレームからの追跡結果を除去するために，逆方向追跡を行う．この場合，追跡に失敗して終了した歩行者の追跡結果を分析し，有効領域との重なりが少ない場合や，他の歩行者との重なりが多い場合など，追跡結果が不安定であると判定されたら追跡結果の除去を行い，他の歩行者は過去に求まっている追跡結果をそのまま追跡結果とする．追跡結果が安定していると判断されたフレームまで戻したら，追跡方向を順方向へと戻し，本来の追跡処理を行う．失敗による逆方向追跡が発生した例を図 8(b) に示す．追跡が失敗する状況として，一人の歩行者に複数の追跡が行われる場合がある．この追跡を続けると，追跡結果間の重なりが大きくなり，一方は追跡に失敗したと判定される．ここで逆方向追跡を行い，過去のフレームにおいて失敗した追跡結果と他の追跡結果との重なりが強い場合には失敗した追跡結果を取り除くことで，重複した追跡結果を修正することができる．

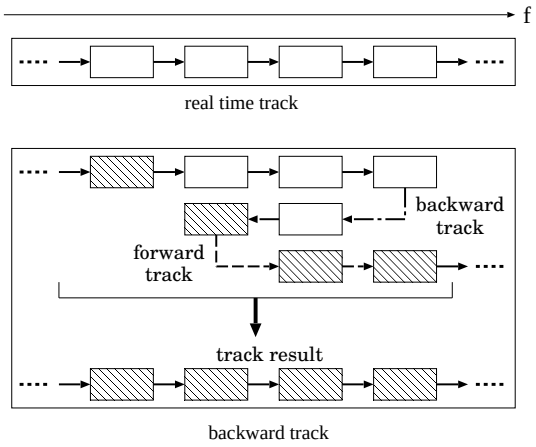


図 9　逆方向追跡

逆方向追跡においても，順方向処理と同様に仮説検証を行うことも考えられるが，無限ループが発生する危険性がある．そのため，本稿では，逆方向追跡においては仮説検証を行わないことにする．

逆方向追跡を用いた追跡の典型例を図 9 に示す．逆方向追跡を用いた場合，順方向と逆方向の追跡を繰り返し，追跡結果を更新する．つまり，各フレームにおいて最終的に得られる追跡結果は，最後に追跡が行われたときの結果となる．

4. 実　　験

4.1 実 験 条 件

複数人物追跡において実時間追跡と，仮説検証追跡と逆方向追跡を組み合わせた追跡系の追跡結果を比較し，その有効性を検証した．

入力画像には，PETS2009 [13] のテストデータである S2.L1 と S2.L2 の二つの動画像を用いた．これらは 768×576 のカラー画像であり，S2.L1 は 795 フレーム，S2.L2 は 436 フレームである．ここで，S2.L1 は複数の歩行者が行き交う動画像であり，歩行者がすれ違うときなどに人物間の重なりが発生する．一方，S2.L2 は多くの歩行者が入り乱れる動画像であり，人物間の重なりが激しいために S2.L1 より追跡が難しい．

なお，歩行者一人の追跡に用いるパーティクルフィルタのサンプル数は 100，試行回数は 10 回とした．

4.2 安定追跡率の比較

S2.L1 および S2.L2 において，予め各フレームの各歩行者の正解位置を手動で求め，追跡結果の位置と比較することで評価を行った．ここでは，画像上において正解位置の 10 ピクセル以内の位置に追跡結果が存在していれば追跡成功とし，追跡に成功している歩行者の割合を全フレームの平均値として算出したものを安定追跡率とした．ここで，隠れが生じている歩行者については正確な位置を求めることが難しいため，隠れを考慮せずに求

表 1　各追跡手法の安定追跡率 [%]

データ	追跡結果	隠れた位置を含む	隠れ 30 %以下
S2.L1	実時間追跡	91.43	92.06
	仮説検証追跡	95.94	96.70
S2.L2	実時間追跡	32.74	35.87
	仮説検証追跡	61.62	64.30

めた安定追跡率と，隠れが 30 %以上生じている歩行者を除いた場合の安定追跡率の 2 つを算出した．

安定追跡率の比較結果を表 1 に示す．この結果より，S2.L1 および S2.L2 において，隠れを含む，含まないに関わらず，実時間追跡より仮説検証を用いた追跡の方が安定追跡率が上がっており，仮説検証が有効であることが確認された．一方，現在の仮説検証では，追跡に失敗したことを誤って判断した場合など失敗する例が存在することが判った．

4. 3　具体的な追跡例

S2.L1 についての具体的な追跡例を図 10 に示す．図 10 の結果より，歩行者間の重なりがない場合にはどちらの手法でも追跡できることがわかる．これは，検出処理と追跡処理が複数歩行者追跡に有効に働いていることを示している．しかし，frame 348-359 の結果を見ると，実時間追跡では標識の後ろで歩行者が交差したとき，それぞれの追跡対象が入れ替わってしまっている．一方，仮説検証追跡では追跡対象の入れ替わりは発生しなかった．(添付資料に仮説検証追跡の結果を示す．)

これは，歩行者が交差したときに追跡対象を入れ替えた場合と入れ替えなかった場合の追跡仮説を生成し，それぞれ検証することによって正解の追跡結果が得られたからである．また，frame 519-541 では 2 人の歩行者が重なりがある状態で登場しているが，有効領域が歩行者一人の場合よりも大きくなってしまうため，実時間追跡では検出されるまでに時間がかかってしまった．一方，仮説検証追跡では重なりがある状態でも追跡を行うことができた．これは，検出仮説により有効領域の分割が行われ，逆方向追跡により重なりのある歩行者でも登場位置が求められたためである．

次に，S2.L2 の追跡例を図 11 に示す．この動画像では人物間の重なりが多いため，実時間追跡では初期フレームの歩行者を検出することが難しかった．また，重なりが無くなった歩行者はある程度の検出，追跡を行うことができたが，影領域の影響があるため，高い精度の結果は得られなかった．一方，仮説検証追跡では，有効領域の分割と逆方向追跡が行われたため，歩行者間の重なりがある登場人物の領域をかなり検出できた．また，強い影領域と重なりがある歩行者も検出，追跡することができた．これらの結果から，検出仮説の生成，および逆方向追跡は検出処理が難しい事例であるほど有効であると言える．(添付資料に仮説検証追跡の結果を示す．)



図 10　S2.L1 の追跡結果の比較

S2.L2 については，歩行者の領域を正しい範囲で検出できない場合や，過剰に検出している場合が存在した．また，各歩行者の追跡では追跡対象の入れ替わりが多くあったため，十分な追跡性能は得られていない．検出性能，追跡性能を向上させるための仮説の生成，検証方法の改良が今後の課題である．

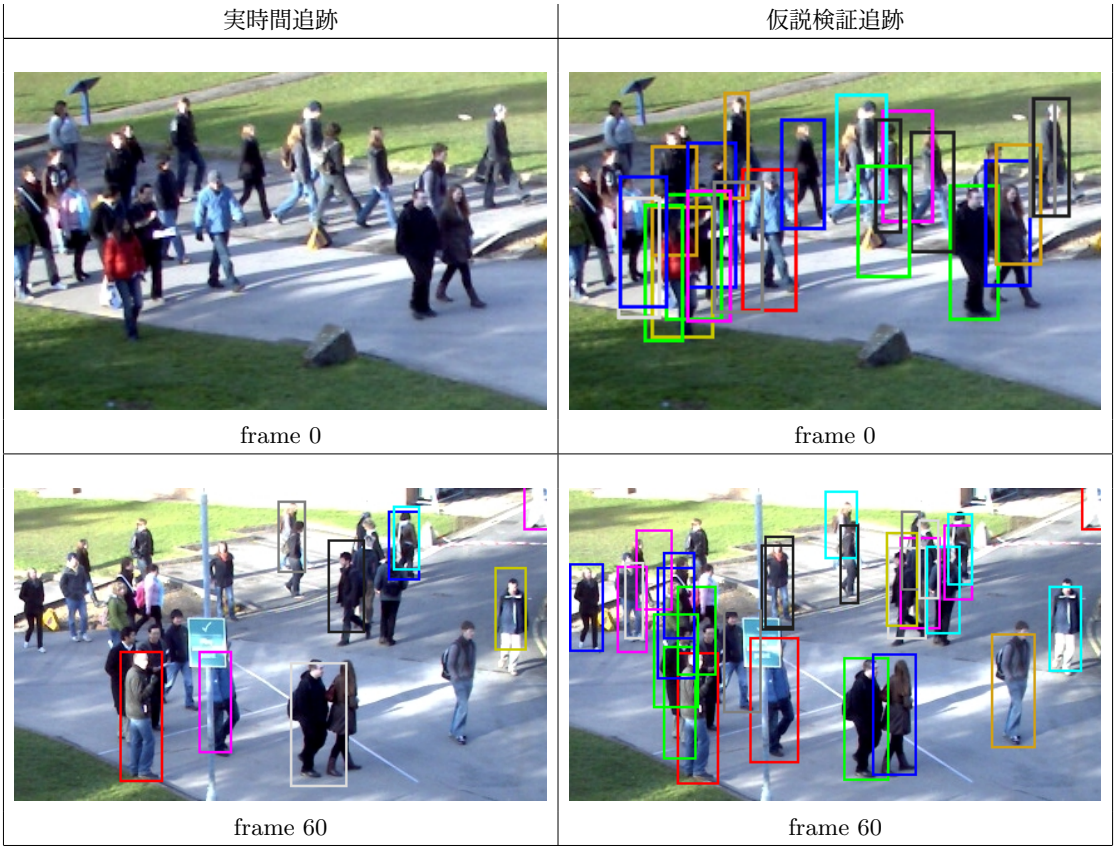


図 11　S2.L2 の追跡結果の比較

5. ま と め

本稿では，複数歩行者追跡において実時間追跡の性能に限界があることを示し，性能の向上を目指す手法として仮説検証追跡と逆方向追跡を導入することにより追跡性能の向上を目指した．動画像を用いた追跡実験結果より，仮説検証追跡の有効性を確認できた．また，特に，検出性能の向上は複雑な事例であるほど効果が大きくなることがわかった．今後の課題としては，より複雑な問題に対応する仮説の生成方法や，検証方法の改良などが挙げられる．

本研究の一部は科学研究費補助金基盤研究 (B)(課題番号 20300067) によった．

文　献

[1] 佐竹純二，尺長健，“階層的注視点制御による動画像上での複数人物追跡，” 電子情報通信学会論文誌 D-II, vol.J86-D-II, no.8, pp.1212-1221, 2003.

[2] J. Satake, and T. Shakunaga, “Human tracking based on hierarchical attention control,” Proc.MVA2000, pp.517-520, Nov. 2000.

[3] J. Satake, and T. Shakunaga, “Multiple target tracking by appearance-based condensation tracker using structure information,” Proc.International Conference on Pattern Recognition (ICPR2004), vol.3, pp.294-297, 2004.

[4] 松原康晴，尺長健，“疎テンプレートマッチングとその実時間物体追跡への応用，” 情報処理学会論文誌, vol.46, no.SIG 9(CVIM 11), pp.60-71, Jan. 2005.

[5] L. Bazzani, D. Bloisi, and V. Murino, “A comparison of multi hypothesis kalman filter and particle filter for multi-target tracking,” Proc.Eleventh IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS2009), pp.47-54, 2009.

[6] J. Yang, P. A.Vela, Z. Shi, and J. Teizer, “Probabilistic multiple people tracking through complex situations,” Proc.Eleventh IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS2009), pp.79-86, 2009.

[7] P. Smith, and G. Buechler, “A branching algorithm for discriminating and tracking multiple objects,” IEEE Trans. Automatic Control, pp.101-104, 1975.

[8] Y. Bar-Sharon, and T. Formanm, “Tracking and data association,” Academic Press, 1988.

[9] T. Quach, and M. Farooq, “Maximum likelihood track formation with the viterbi algorithm,” Proc. 33rd IEEE Conf. Decision and Contril(CDC ’94), pp.271-276, 1994.

[10] F. Yan, W. Crismas, and J. Kittler, “Layered data association using graphtheoretic formulation with applications to tennis ball tracking in monocular sequence,” IEEE Trans. pattern analysis and machine intelligence, vol.30, no.10, pp.1814-1830, Oct. 2008.

[11] Z. Kalai, K. Mikolajczyk, and J. Matas, “Forward-backward error: Automatic detection of tracking failure,” Proc. ICPR2010, pp.2756-2759, 2010.

[12] M. Isard, and A. Blake, “Condensation-conditional density propagation for visual tracking,” International Journal of Computer Vision, vol.29, pp.5-28, 1998.

[13] Proceedings of Eleventh IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS2009), June 2009.