DEIM 2022 C41-5

人物再同定に基づく段階的クラスタリングを用いた複数人物追跡

山崎　智史† 　劉　 健全†

† NEC バイオメトリクス研究所　〒 211-8666 神奈川県川崎市中原区下沼部 1753
E-mail: †{s-yamazaki31,jqliu}@nec.com

あらまし　映像中の人物追跡において，外観容姿の類似性を考慮した人物再同定は長い時間範囲での追跡に有用である．特に視界障害物による数秒程度の短時間遮蔽があった場合，人物再同定により高精度に人物追跡ができる．しかし人物の出現時間差が長くなるほど同一人物の候補数が増えるため，別人物を誤って追跡する可能性が高くなる．出現時間差が長い場合には，誤追跡を抑制しつつ人物再出現を捕捉できる追跡手段が求められる．そこで本研究では人物の出現時間差を考慮した段階的クラスタリング手法を提案し，複数人物の追跡を高精度で実現する．本論文では提案手法である段階的クラスタリングの概要を述べ，実験評価により提案手法の性能を示す．

キーワード　人物追跡，人物再同定

1　はじめに

複数人物追跡はコンピュータビジョン研究の中でも重要なトピックの一つである．例えば公共エリアにおける長時間滞留者などの不審人物の発見や病院内の患者の所在確認などの用途での複数人物追跡の利用が期待される．しかしながら公共エリアでの長時間滞留者はカメラ画角から一度退出した後に再出現する可能性がある．再出現の前後では図 1 のように異なる体の向きで出現する場合もあり，人物の再追跡は難しい．現実のアプリケーションでは人物の再出現があっても追跡漏れおよび他人受入の少ない高精度の追跡手法が求められる．

昨今の複数人物追跡研究では tracking by detection と呼ばれる検出に基づく人物追跡の手法が主流となっている．検出に基づく人物追跡手法では人物検出器により映像の各フレームにおける人物を含む矩形を検出し，検出した人物と前フレームまでで追跡した人物（追跡体）とのデータ関連付けにより人物追跡を実施する．高精度かつ高速処理が実現できる深層学習ベースの人物検出技術の進展 [1–5] により，検出に基づく人物追跡手法では映像中に現れた人物を漏れなく検出できるため，高い追跡性能を発揮できる．データ関連付けでは検出した人物と追跡体との間の類似度を算出し，類似度に基づいてそれらを対応付ける．データ関連付けのための有用な手がかりとしては外観容姿の特徴がある．外観容姿の特徴は視覚障害物による一時的な遮蔽があっても人物再同定ができるため，多くの人物追跡技術で用いられている．外観容姿の類似度は人物再同定モデル（ReID モデル）から算出される ReID 特徴量のユークリッド距離またはコサイン類似度によって計算できる．DeepSORT [6] は追跡用に学習を行った ReID モデルを用いて，外観容姿の類似性を人物追跡に活用している．他にも人物検出と ReID 特徴量抽出の両方の処理を同時に行う結合モデル [7–11] による追跡の堅牢化がなされている．

上記の追跡技術の結果として算出された追跡体を入力とした追加分析を行うことで，より追跡漏れの少ない人物追跡を試みる提案がいくつかなされている．TrackletNet [12] では追跡体に含まれる各人物の位置および外観特徴の入力として追跡体間の類似性を算出するモデルを学習する．算出された追跡体間の類似性に基づくクラスタリングにより堅牢な人物追跡が実現される．また Recurrent Neural Network (RNN) に基づく人物追跡でも追跡体を入力とする提案がなされている [13]．しかし RNN ブロックの伝搬に基づくと，時間的に離れた検出人物との類似性を低く見積もってしまう恐れがあり，数分間の空きがあってからの再出現を捉えるのは困難である．

図 1: 映像中での出現人物のサンプル．公共エリアではカメラ画角から退出して 1 分以上経過後に同一人物が体の向きを変えて再出現する可能性がある．

多くの人物追跡手法では出現時間差が数秒以内の人物矩形間を対象としてデータ関連付けを行っており，同一人物が数分経過後にカメラ画角に再出現した際には別人物として追跡されてしまう．そこで DeepCC [14] は時間経過に堅牢な ReID 類似度で関連付けられるグラフを構築し，グラフ分割によるクラスタリング手法を適用することで出現時間差がより長い場合および別カメラで再出現した場合での人物追跡手法を提案している．しかしながら DeepCC はグラフ構造の頂点としては人物矩形位置および対応する ReID 特徴量を想定しており，TrackletNet [12] のような追跡体をデータ単位として長い出現時間差および別カメラでの出現を想定した人物追跡手法の提案はなされていない．

図 2: 提案手法の全体アーキテクチャ．出現時間差が 5 秒以内の人物矩形間は人物クラスタリングによってデータ関連付けを行う．出現時間差が 5 秒以上の人物矩形間のデータ関連付けは，人物クラスタリングで算出した追跡体をデータ単位とした追跡体クラスタリングによって実施する．

以上の背景から，本論文では長い出現時間差および別カメラでの出現を想定した段階的クラスタリングによる複数人物追跡手法を提案する．先行研究では，数分以上の長時間経過後の再出現を加味した人物追跡や複数カメラを跨った人物追跡の高精度化に追跡体を入力とする手法は使われてこなかった．そこで本論文では追跡体を入力とした人物追跡に適した追跡体の生成手法と，生成した追跡体を入力とした長時間範囲および複数カメラを対象としても適用可能な追跡手法を提案する．提案手法では 1 段階目として出現時間差が短い人物間の外観容姿の類似性に基づく人物クラスタリングを実行することで，誤追跡がほとんどない追跡体を生成する．次に 2 段階目では 1 段階目で生成した追跡体間の類似性に基づく追跡体クラスタリングを行うことで長時間の人物追跡を実現する．以下では提案手法の詳細を述べ，撮影映像を用いた提案手法の評価結果を示し，最後に結論を述べる．

2 手　　法

本研究の提案手法の全体図を図 2 に示す．提案手法では映像の画像フレームおよび画像フレーム内の人物矩形位置を入力として，出現時間差が短い人物間の類似性に基づく人物クラスタリングおよびその結果を入力とした追跡体クラスタリングを行い，人物の追跡を行う．入力の人物矩形位置は既存の人物検出器 [1–5] で算出し，人物矩形画像から外観容姿の類似度計算のための ReID 特徴量の抽出処理を実施する．人物矩形画像から抽出した ReID 特徴量に基づいて，提案手法では人物の出現時間差を考慮した 2 段階のクラスタリングを実施する．

2.1 出現時間差が短い人物間の類似性に基づく人物クラスタリング

提案手法の人物クラスタリングは密度ベースのクラスタリング手法である DBSCAN [15] を採用し，外観容姿の類似性に基づいたクラスタリングを行う．DBSCAN は入力された全ての人物矩形データに対して追跡体に対応するクラスタ ID を割り当てる．クラスタ数を指定しないノンパラメトリックなクラスタリング手法を用いているため，映像中のユニークな人物数が何人でも追跡ができる．ただし DBSCAN では時空間的な制約を考慮していないため，同時刻の別位置にある人物矩形が同一人物として認識されうる．そこで本手法では Intersection over Union (IoU) に基づく追跡手法である SORT [16] を利用したクラスタの精緻化を行い，時空間的な制約から不適切なクラスタ ID 割当を修正する．以下では DBSCAN および精緻化の詳細を述べる．

2.1.1 外観容姿の類似性に基づく DBSCAN

提案手法で利用する DBSCAN のアルゴリズムを Algorithm 1 に示す．Algorithm 1 は DBSCAN [15] に沿って，類似度閾値 ϵ と類似データの最小数 $minPts$ をパラメタとして同一クラスタ（同一人物）の判定を行う．Algorithm 1 の入力は人物矩形の ReID 特徴量および時刻情報を含むデータとする．まずはじめに Algorithm 1 では検出時間差 t_{diff} の間で検出された人物の中から類似した人物の検索（SimilaritySearchOverDB）を行う．この検索では ReID 特徴量間の特徴量距離が ϵ よりも低い，入力データと類似度の高い人物を列挙する．列挙数が $minPts$ 未満の場合，入力データと類似人物はないと認識し，入力データに新規のクラスタ ID を割り当てる．列挙数が $minPts$ 以上の場合は列挙した類似人物が属しているクラスタ ID の中から主要クラスタ ID である C_{pri} を 1 つ選択（selectPrimaryClusterId）する．列挙した類似人物が属しているクラスタ ID は複数存在する場合がある．その場合は属している人物が最も多いクラスタを主要クラスタとし，主要クラス

Algorithm 1 DBSCAN with short-term relationships.

Input: 入力データ D, 類似度閾値 ϵ, 最大フレーム時間差 t_{diff}, 最小点数 $minPts$

$C = 0$
for d in D **do**
  N = SimilaritySearchOverDB(d, ϵ, t_{diff})
  if $|N| < minPts$ **then**
    $C_{pri} = C + 1$
  else
    C_{pri} = selectPrimaryClusterId(N)
    updateClusterId(N, C_{pri})
  end if
  label(d) = C_{pri}
  insertDataIntoDB(d)
end for
return D

Algorithm 2 SORT for each cluster.

Input: 同一クラスタの人物 $D = \{d_{1,1}, ..., d_{M,t}\}$, IoU 閾値 IoU_{min}

$T = \{\}$ /* 追跡体の初期化 */
$R = \{\}$ /* 最終結果の初期化 */
for $i = 1 \ldots t$ **do**
  D_{cur} = selectCurrentDetections(D, i)
  T_{new} = predict(T)
  C = ComputeCostMatrix(T_{new}, D_{cur})
  T_{match},D_{match},$D_{unmatch}$ = MinCostMatching(C,IoU_{min})
  update(T_{match},D_{match})
  createNewTracklets(T, $D_{unmatch}$)
  deleteExpiredTracklets(T)
  R.append(D_{match}, $D_{unmatch}$)
end for
return R

タ以外の人物データのクラスタ ID を主要クラスタ ID である C_{pri} に変更する（updateClusterId）．Algorithm 1 ではこの updateClusterId の処理によりクラスタの統合が行われる．入力データに C_{pri} を割り当てた後，入力データをデータベースへ格納（insertDataIntoDB），以後の SimilaritySearchOverDB での検索対象データとして用いる．入力データのすべてにクラスタ ID を割り当てるまで以上の手順を続ける．

Algorithm 1 ではオリジナルの DBSCAN [15] と入力データの処理順序が異なる．オリジナルの DBSCAN [15] の場合，起点のデータから類似したデータに対してクラスタ ID 割当を実施する．そして類似データがなくなったら，クラスタ ID が未割当の別データを起点として類似検索とクラスタ割当を実施する．一方で Algorithm 1 では時々刻々と検出される人物を入力としたオンライン処理も実施可能なように設計している．SimilaritySearchOverDB では以前の入力となったクラスタ ID が割当済みの結果が検索の対象となる点もオリジナルの DBSCAN と異なる．しかしながら updateClusterId によりクラスタの ID の統合が行われるため，$minPts = 0$ の場合のクラスタリング結果はオリジナルの DBSCAN と一致する．なお DeepSORT [6] などの多くのオンライン追跡アルゴリズムでは，Algorithm 1 の updateClusterId のようなクラスタ ID の統合処理は行われず，入力順序によって追跡結果が変化してしまう．

2.1.2 SORT を利用した追跡の精緻化

提案手法の DBSCAN では外観容姿を考慮した人物クラスタリングにより追跡体に対応するクラスタが生成される．しかしながら DBSCAN では人物の位置および動きが考慮されていないため，不適切な人物が同一人物クラスタとして認識されうる．そこで本提案手法では同一人物クラスタに SORT [16] を適用して追跡結果の精緻化を行う．

SORT [16] は追跡体の位置，動きを Kalman filter [17] の線形速度状態モデルで近似する．追跡体の状態 x は以下のようにモデル化される．

$$x = [u, v, s, r, \dot{u}, \dot{v}, \dot{s}]^T. \quad (1)$$

u と v はそれぞれ追跡人物の水平，垂直方向のピクセル位置を表し，矩形のスケールおよび縦横の長さ比は s, r で表す．$\dot{u}, \dot{v}, \dot{s}$ はそれぞれ u, v, s の速度を表す．u, v, s, r は Kalman filter [17] の枠組みで観測量として扱われ，r に関しては速度を状態量として保持しない．Kalman filter [17] では u, v, s, r で表現されるピクセル位置およびスケールの時間変化を推論する．もし現在フレームで線形割当で対応する人物がいなかった場合は観測による補正は無く，単に線形速度モデルでの状態予測が次フレームでも用いられる．

Kalman filter [17] による状態予測を用いた追跡体・検出人物の対応付けアルゴリズムを Algorithm 2 に示す．Algorithm 2 は同一クラスタに属するすべての人物データを入力とし，入力の各人物データに追跡 ID を割り当てたデータを最終的な追跡結果として出力とする．各人物データは人物矩形のピクセル位置と検出された時間の情報を持つことを想定する．Algorithm 2 では同一クラスタに属する人物が最初に検出された時間を 1，最後に検出された時間を t として，時系列順序のループ処理で人物追跡を実行する．ループ処理でははじめに現在時間の人物データを取得する（selectCurrentDetections）．次に Kalman filter [17] を用いて既存の追跡体の現フレームでの位置予測（predict）を行う．追跡体と検出した人物の対応付けは追跡体の現フレームでの予測位置の矩形と検出した人物の矩形の Intersection over Union (IoU) に基づいて実施する．Algorithm 2 では ComputeCostMatrix が算出するコスト行列は同フレームでのすべての追跡体と検出した人物間での IoU 距離によって計算する．ここで IoU 距離は $1 - IoU$ で定義され，値が小さいほど類似していることを表現する．このコスト関数に線形割当アルゴリズムであるハンガリアン法 [18] を適用して，最小コストとなる追跡体・検出人物の対応を算出する（MinCostMatching）．最終的な対応付けでは IoU 閾値 IoU_{min} による対応棄却が実装されている．これにより IoU が低い追跡体と検出した人物の対は別人物として判定する．そして Algorithm 2 は対応する追跡体が存在する検出人物のピクセル位置情報を Kalman filter [17] の枠組みで観測量として扱い，対応する追跡体の状態更新を行う（update）．また対応する追跡体が存在しない検出人物から新規の追跡体の生成が行われる（createNewTracklets）．新規の追跡体は対象人物の短冊情報から状態が初期化され，状態に含まれる速度はゼロ，速度の共分散も大きな値を初期値とする．最後の観測から一定フレーム以上経過した追跡体は Kalman filter [17] の予測が不安定となるため削除する（deleteExpiredTracklets）．オリジナルの SORT では不正確な予測の抑制と冗長な追跡体の削除のために経過フレーム数を 1 としている．提案手法では DBSCAN で利用した検出時間差 t_{diff} を利用する．

なおオリジナルの SORT [16] では誤追跡抑制のために追跡体生成後の数フレームは連続的に観測されないと追跡体を削除するように設定されている．しかしながら提案手法では既に DBSCAN で誤追跡抑制が実施できているため，生成後の数フレームでの連続的な観測は強制しない．

2.2 追跡体クラスタリング

一般的に出現時間差が長くなるほど同一人物の候補となる人物矩形の数が増えるため，別人を誤追跡してしまう可能性が増えてしまう．そこで提案手法では長い出現時間差の人物追跡のために，追跡体を入力とした密度ベースクラスタリングを行う．追跡体クラスタリングでは出現時間差が短い人物間の類似性に基づく人物クラスタリングで算出した追跡体を入力とする．追跡体には異なる時刻で出現した同一人物矩形に対応する ReID 特徴量が複数含まれる．追跡体に含まれる人物矩形画像の中には体の一部の遮蔽や被写体ぶれにより人物再同定に適していない画像も含まれ得る．そこで提案手法では追跡体間の距離を各々の追跡体に含まれる ReID 特徴量の総当たり組合せで算出された距離の平均値として定義する．距離の平均値を利用することで，人物再同定に適していない人物矩形の影響を抑制した堅牢な同一人物判定が期待される．

追跡体クラスタリングは追跡体をデータ単位として Algorithm 1 と同様のクラスタ ID 割り当てを実行する．同一のクラスタ ID が割り当てられた追跡体は，追跡体に含まれるすべての人物矩形を同一人物として認識する．また追跡体に含まれる人物矩形の数が少ない（追跡体のサイズが小さい）場合は堅牢な同一人物判定が実施できない可能性がある．そのためクラスタリングの入力とする追跡体の最小サイズを設定し，最小サイズ未満の追跡体は独立したクラスタとして扱うことで最終的な追跡結果を得る．

3 評　　価

本研究では提案手法を屋外設置カメラで撮影した映像に適用し，提案手法の有用性を確認した．評価の映像入力としては屋外に設置したカメラ 1 台での撮影映像を利用する（図 3）．図 1 に示した人物画像はこの撮影映像から抜粋した画像サンプルである．この映像はフル HD 解像度，5 FPS，約 3 分間の映像長で，50 名の人が行き来するシーンが撮影されている．映像に現れる各人物はカメラ画角から消えて 30 秒以上経過後の再出現を最低でも 1 回行っている．図 3 の赤枠で囲われた人物のように再出現の前後で服装に変化はないが，体の向きが前後逆になって出現する．提案手法の追跡結果は映像の各画像フレームに対して作成した出現人物の矩形位置および正解人物 ID と突き合わせることで追跡精度を算出する．追跡精度はデータ関連付けに重きを置いた評価指標であるアソシエーション精度（AssA, AssPr，AssRe）を用いる [19].

$$\mathrm{AssA} = \frac{1}{|\mathrm{TP}|}\sum_{c\in\{\mathrm{TP}\}}\frac{\mathrm{TPA}(c)}{\mathrm{TPA}(c)+\mathrm{FPA}(c)+\mathrm{FNA}(c)}, \quad (2)$$

$$\mathrm{AssPr} = \frac{1}{|\mathrm{TP}|}\sum_{c\in\{\mathrm{TP}\}}\frac{\mathrm{TPA}(c)}{\mathrm{TPA}(c)+\mathrm{FPA}(c)}, \quad (3)$$

$$\mathrm{AssRe} = \frac{1}{|\mathrm{TP}|}\sum_{c\in\{\mathrm{TP}\}}\frac{\mathrm{TPA}(c)}{\mathrm{TPA}(c)+\mathrm{FNA}(c)}, \quad (4)$$

表 1: 追跡精度の評価結果.

	AssA	AssPr	AssRe
DeepSORT	0.2736	0.8367	0.2864
提案手法（1 段階目のみ）	0.3424	**1.0**	0.3424
提案手法	**0.7711**	0.9318	**0.8123**

$$\mathrm{TPA}(c) = \{k\},\ k \in \{\mathrm{TP}|\mathrm{prID}(k) = \mathrm{prID}(c) \wedge \mathrm{gtID}(k) = \mathrm{gtID}(c)\}, \quad (5)$$

$$\mathrm{FPA}(c) = \{k\},\ k \in \{\mathrm{TP}|\mathrm{prID}(k) = \mathrm{prID}(c) \wedge \mathrm{gtID}(k) \neq \mathrm{gtID}(c)\}, \quad (6)$$

$$\mathrm{FNA}(c) = \{k\},\ k \in \{\mathrm{TP}|\mathrm{prID}(k) \neq \mathrm{prID}(c) \wedge \mathrm{gtID}(k) = \mathrm{gtID}(c)\}, \quad (7)$$

TP は 1 つの人物矩形データ，{TP} は評価映像で検出されるすべての人物矩形データを表す．各人物矩形データに対応する正解人物 ID(gtID) と追跡結果で割り当てた追跡 ID(prID) に基づき，正しいデータ関連付け (TPA) および誤ったデータ関連付け (FPA および FNA) の数が算出される．TPA, FPA, FNA によりデータ関連付けの総合評価を AssA，特にデータ関連付けの誤追跡の少なさを評価したものが AssPr，追跡漏れの少なさを評価したものが AssRe になる．なお本評価における AssA では各画像フレームの人物矩形は既に人手でアノテーションされたものを利用するため，人物検出器の誤検出の影響は含まれない．MOT ベンチマークなどでも利用される一般的な AssA の定義では人物検出器の誤検出の寄与も加味される [19].

提案手法のオンライン DBSCAN では外観容姿の類似性の計算に ReID 特徴量が必要となる．本評価では OSNet [20, 21] の学習済みモデル [22] を利用する．評価映像中の各人物矩形画像から ReID 特徴量抽出を行い，ReID 特徴量間の距離をユークリッド距離として算出，距離に基づいて追跡 ID 割当を実施した．また比較評価では利用する ReID 特徴量を OSNet の学習済みモデルに置き換えた DeepSORT [6] を用いている．提案手法における DBSCAN の類似度閾値 ϵ および比較評価で用いた DeepSORT での類似距離閾値は 15.0 に統一することで，ReID 特徴量距離の上ではどちらの手法でも同等に同一人物判定が可能になるように設定した．評価用プログラムは python で実装，評価実験は Intel Xeon 3.0 GHz の CPU と Ubuntu 18.04 を OS とする計算機上で行った．

表 1 に本評価で得られた提案手法の追跡精度を示す．表 1 では既存手法である DeepSORT，提案手法におけるクラスタリングの 1 段階目のみを適用した場合，および 2 段階目も適用した場合の AssA,AssPr,AssRe を示している．なお 2 段階目適用時には追跡体間の距離の閾値および入力追跡体の最小サイズを変えて最も AssA が高かった結果を記載している．表 1 によれば，提案手法である段階的クラスタリングの結果は 1 段階目の

図 3: 評価映像のサンプル．赤枠で囲われた人物はカメラ画角から一度退出して 1 分以上経過後に再出現する．

みの結果に比べて 0.42 ポイント程度の AssA 向上があることが分かる．今回の評価映像では同一人物の再出現があるため，DeepSORT や 1 段階目のみの提案手法による人物追跡では追跡漏れが発生して AssRe が大きくならない．提案手法は出現時間差に関係なく，人物再同定を行うため AssA および AssRe の向上が実現できている．ただし AssA の向上は追跡漏れの改善によるもので，AssPr が 0.07 ポイント程度低下していることから追跡体クラスタリングによって誤追跡の数は若干増えている．

追跡体クラスタリングでの距離の閾値を変えた場合の AssA，AssPr および AssRe を図 4 に示す．図 4 では追跡体クラスタリングの入力の追跡体を提案手法の人物クラスタリング（clustering），DeepSORT で生成した場合の結果をプロットしている．また図 4 には入力の追跡体の最小サイズを 5 とした場合の結果もプロットしている．距離の閾値が 0 の場合は追跡体の拡大が行われないため，図 4 の clustering と DeepSORT の追跡精度は表 1 の提案手法（1 段階目のみ）と DeepSORT の結果にそれぞれ一致する．

図 4a において提案手法は追跡体の最小サイズが 1，距離の閾値が 22 で最大の AssA となる．一方で図 4c では提案手法の AssPr は距離の閾値この AssA 最大値を与える 21 以上では減少傾向になる．入力の追跡体生成に DeepSORT を使った場合でも，AssA の最大値を取る閾値以上で AssPr が減少傾向を示す振る舞いが現れる．距離の閾値が高くなり，誤追跡が増加すると追跡精度の総合評価である AssA が劇的に低下する．そのため高い AssPr が維持できる最大の閾値付近で AssA の最大値が与えていると考えられる．

また追跡体の最小サイズが 5 の場合，clustering，DeepSORT の手法に関わらず，最小サイズが 1 の場合に比べて距離の閾値の増加による AssPr の低下が少なくなる傾向がある．そのため高い AssPr を維持した上で最大の AssA を取るためには追跡体の最小サイズは重要な役割を担う．実際，図 4a では提案手法でも追跡体最小サイズが 5 で AssPr=1.0 の場合の AssA 最大値は閾値 23.6 で 0.6910 となり，追跡体最小サイズが 1 で AssPr=1.0 の場合での 0.4910（閾値 20.1）と比べて高い．

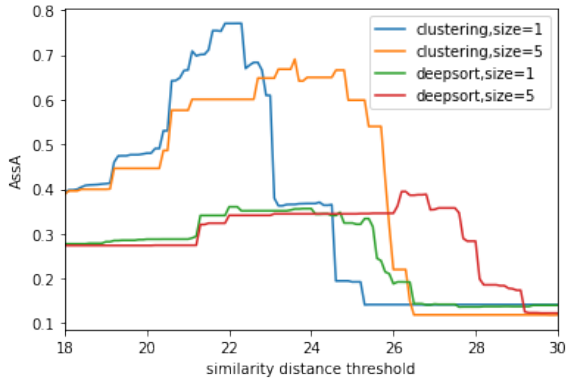
追跡体間の距離のヒストグラムを図 5 に示す．図**??**では同一人物間および別人物間の追跡体の距離ヒストグラムをそれぞれプロットしている．なおヒストグラムのビン幅は 1，縦軸を相対度数密度としてプロットしている．

ヒストグラムの振る舞いは追跡体生成の方法によって変化する．図 5a によると，clustering および DeepSORT でヒストグラムが最大となる距離区間はそれぞれ (31,32) および (33,34) となる．図 5a において距離が 30 以下となる数は clustering の方が DeepSORT に比べて多く，clustering での同一人物の追跡体間距離は平均として DeepSORT に比べて低くなっていると言える．DeepSORT で生成した追跡体では AssPr が clustering で生成した追跡体に比べて低く，誤追跡が多い．誤追跡を含む追跡体での距離では，別人との距離が追跡体に含まれる人物間での距離平均に加味されてしまう．別人との距離は高い値が出る傾向があるため，追跡体に含まれる人物間での距離平均は誤追跡を含まない追跡体の場合と比べて高くなると考えられる．つまり提案手法の 1 段階目の人物クラスタリングは誤追跡を含まない追跡体を生成できるため，2 段階目の追跡体クラスタリングに適した方法であると言える．

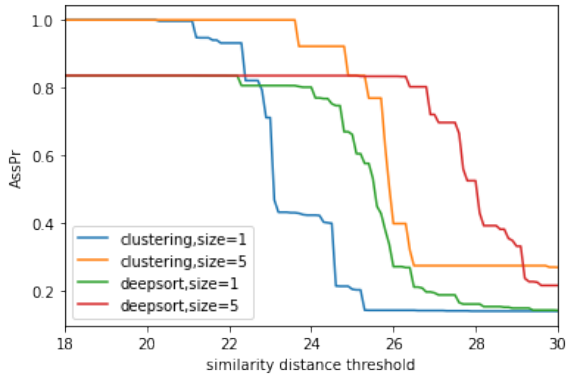
一方で図 5b によると clustering および DeepSORT でのヒストグラム最大値はそれぞれ (34,35) と (33,34) となり，図 5a と比較すると手法間で振る舞いの差が小さい．また clustering，DeepSORT ともに距離が 25 以下となる数は 1% 未満となる．追跡体クラスタリングでは誤追跡がすぐに同一クラスタに伝搬されてしまうため，極めて少数でも誤りが入ると AssPr が大きく低下してしまう．そのため図 4 では距離閾値が 25 以上では AssA および AssPr が追跡体クラスタリングを実施する前よりも低い値を取るようになっていると考えられる．

4 結　　論

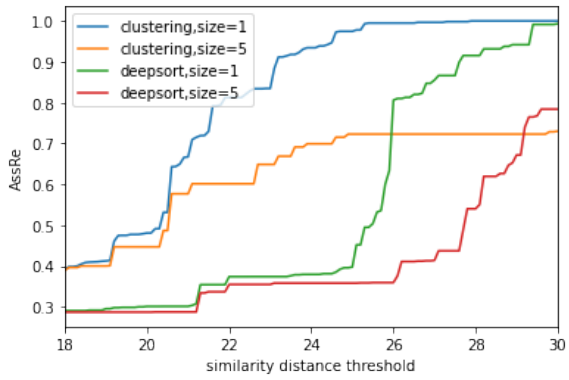
本論文では，人物の出現時間差を考慮した 2 段階の人物クラスタリングを用いて複数人物の追跡を行う手法を提案した．評価実験の結果，段階的クラスタリングを行うことで，人物の再出現のある映像でも高い追跡精度を達成できることが分かった．提案手法の 2 段階目処理である追跡体クラスタリングでは，入力となる追跡体の最小サイズを設定することで誤追跡を少ない追跡ができると期待される．本論文での評価では約 3 分間の映像を入力としたが，映像長をさらに長くすると誤追跡が発生する可能性が高くなる．そのため今後はより長い映像での追跡精



(a) AssA



(b) AssPr



(c) AssRe

図 4: 追跡体クラスタリングでの距離の閾値を変えた場合の追跡精度．横軸を追跡体間の距離として (a) データ関連付けの総合評価スコアである AssA と (b) 誤追跡の少なさを評価した AssPr，(c) 追跡漏れの少なさを評価した AssRe を示している．

度の評価が必要である．

文　　献

[1] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[2] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[3] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[4] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[5] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, volume 2015-January, pages 91–99. Neural information processing systems foundation, 2015.

[6] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017.

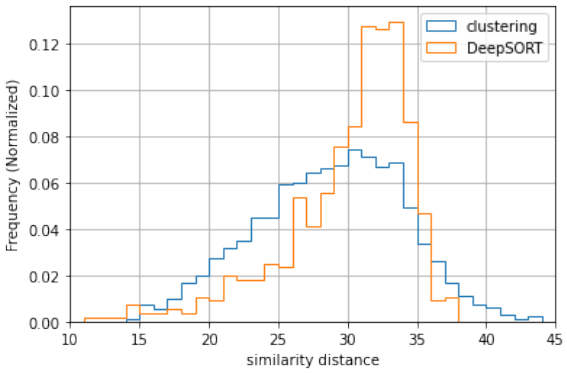
[7] Chao Liang, Zhipeng Zhang, Yi Lu, Xue Zhou, Bing Li, Xiyong Ye, and Jianxiao Zou. Rethinking the competition between detection and reid in multi-object tracking. *arXiv preprint arXiv:2010.12138*, 2020.

[8] Zhichao Lu, Vivek Rathod, Ronny Votel, and Jonathan Huang. Retinatrack: Online single stage joint detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14668–14678, 2020.

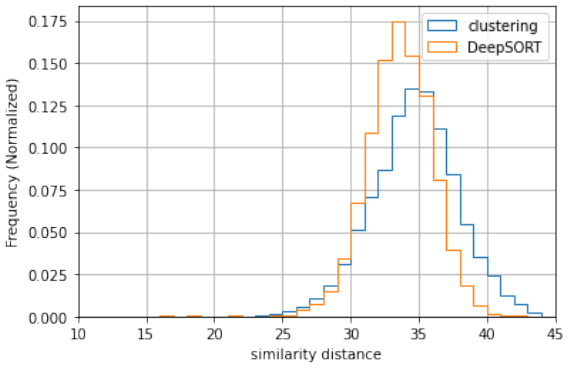
[9] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 164–173, 2021.

[10] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, pages 1–19, 2021.

[11] Jialian Wu, Jiale Cao, Liangchen Song, Yu Wang, Ming Yang, and Junsong Yuan. Track to detect and segment:



(a) 同一人物の追跡体間の距離



(b) 別人物の追跡体間の距離

図 5: 追跡体間の距離のヒストグラム．ビン幅を 1，縦軸を相対度数密度としてプロットしている．

An online multi-object tracker. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12352–12361, 2021.

[12] Gaoang Wang, Yizhou Wang, Haotian Zhang, Renshu Gu, and Jenq-Neng Hwang. Exploit the connectivity: Multi-object tracking with trackletnet. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 482–490, 2019.

[13] Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 300–311, 2017.

[14] Ergys Ristani and Carlo Tomasi. Features for multi-target multi-camera tracking and re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6036–6046, 2018.

[15] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press, August 1996.

[16] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016.

[17] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 1960.

[18] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[19] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision*, pages 1–31, 2020.

[20] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Omni-scale feature learning for person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3702–3712, 2019.

[21] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Learning generalisable omni-scale representations for person re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[22] Kaiyang Zhou and Tao Xiang. Torchreid: A library for deep learning person re-identification in pytorch. *arXiv preprint arXiv:1910.10093*, 2019.