

令和7年度卒業論文

Mask IoU を加えた nasuSORT による
重なり・非線形的な移動に頑健な小動物追跡

2026年2月6日提出

学籍番号：60223700

提出者名：QIAN YIJIE

プログラム名：宮崎大学 工学部 工学科
情報通信工学プログラム

指導教員：椋木 雅之

概要

本研究では、重なりや非線形的な移動が頻発する環境下における複数物体追跡の精度向上を目的として、セグメンテーション情報を活用した追跡手法を提案する。従来の SORT 系追跡手法は、バウンディングボックス (BB_{ox}) に基づく IoU (Intersection over Union) 計算を主とするため、物体同士が接近・重なり合う場面において ID スイッチが発生しやすいという問題があった。また、Soft-NMS を導入した場合、検出数の増加により誤ったトラック生成が生じる可能性も指摘されている。

本研究では、YOLO11 のセグメンテーションによって得られるマスク情報に着目し、マスクに基づく IoU を用いた第一段階マッチングと、BB_{ox} に基づく第二段階マッチングを組み合わせた二段階マッチング手法を構築した。さらに、各トラックに対して検出を一意に対応付ける排他的処理を導入することで、Soft-NMS 使用時に発生しやすい追跡の誤対応を抑制した。

複数の動画データを用いた追跡実験の結果、提案手法は既存手法と比較して ID スイッチ数および生成される追跡 ID の数を削減できることを確認した。これにより、本手法は遮蔽や動的カメラ環境においてもロバストな複数物体追跡を実現できることを示した。

目次

1. はじめに.....	1
2. 複数小動物追跡の従来研究.....	3
2.1. 複数物体追跡.....	3
2.2. SORT.....	4
2.3. nasuSORT.....	4
2.3.1. 複数小動物用 SORT による追跡の安定化.....	5
2.3.2. Soft-NMS による検出漏れの抑制.....	7
2.3.3. 複数小動物用 SORT と Soft-NMS の組み合わせ.....	8
2.4. YOLO.....	9
2.5. 従来の手法の問題点.....	10
3. 重なり・非線形的な移動に強い小動物追跡.....	12
3.1. Mask IoU を加えた nasuSORT による小動物追跡精度の改善.....	12
3.2. 提案手法の流れ.....	13
3.3. Mask IoU の計算.....	14
3.3.1. BBox IoU の定義.....	15
3.3.2. Mask IoU の定義.....	15
3.3.3. 遮蔽・重なり環境における利点.....	17
3.4. Mask の予測.....	18
3.5. Mask による第一段階マッチング.....	19
3.5.1. 第一段階マッチングの目的.....	19
3.5.2. Mask IoU に基づく類似度計算.....	19
3.5.3. コスト行列の構築.....	20
3.5.4. 第一段階マッチングのアルゴリズム.....	20
3.5.5. 本研究における意義.....	20
3.6. 排他的処理.....	21
4. 評価実験.....	23
4.1. 実験データ.....	23
4.2. 検出器の学習.....	25
4.3. 実験設定.....	25
4.4. 実験結果.....	27

4.5. 考察.....	30
5. おわりに.....	31
謝辞.....	32
参考文献.....	33

1. はじめに

複数物体追跡 (Multiple Object Tracking: MOT) [1]は、映像中に存在する複数の物体に対して一貫した識別子 (個体ごとに割り当てられる固有の番号: ID) を付与しながら、時間的に連続して追跡する技術である。MOT は、自動運転における周辺環境認識や監視システム、スポーツ解析、さらには野生動物の生態や行動の解析など、幅広い分野で重要な役割を果たしている。

近年、深層学習に基づく物体検出器の高精度化に伴い、画像から物体を見つける「検出 (Detection)」の結果を時間軸でつなぎ合わせる「追跡 (Track)」へと統合する Tracking-by-Detection 型の手法が主流となっている。その代表的な手法として SORT[2]といった追跡アルゴリズムがあり、計算効率と実装の容易さから多くの応用で利用されている。しかし、従来の Tracking-by-Detection 型の手法は主に人や車両といった比較的大きく、移動が滑らかな対象を想定して設計されている。

一方で、小動物を対象とした追跡には、従来の MOT 手法では十分に対処できない特有の課題が存在する。小動物の追跡では、(1)個体間の外見的差異が小さく互いに類似していること、(2)群れ行動により個体同士が接近・重なりによる遮蔽によって一時的に検出が失われやすいこと、(3)急激な方向転換や速度変化を伴うルールのない移動という非線形的な移動が多く見られることが問題となる。このような状況では、検出の一時的な欠落や追跡データ (Track) と検出結果 (Detection) の紐付けミスが発生しやすく、追跡対象の ID が途中で入れ替わってしまう「ID Switch」や追跡そのものが途切れてしまう「Track の消失」といった追跡性能の低下が顕著となる。

SORT[2]とその系列の追跡手法では、物体を長方形の枠である「Bounding Box (BBBox)」で表現し、BBBox 同士がどれだけ重なっているかを示す指標である「Intersection over Union (IoU)」を用いて「前後のフレームで同一物体である」と判定する「マッチング (対応付け)」を行っている。しかし、遮蔽や重なりが生じた場合、BBBox は物体の実際の可視領域や形状を正確に表現できず、IoU に基づく類似度評価の信頼性が低下する。さらに、非線形的な移動によって予測位置と検出位置の乖離が大きくなると、BBBox IoU は急激に低下し、本来対応付けるべき Track と Detection がマッチングされない問題が生じる。

これらの課題に対し、本研究では、小動物追跡に特化した追跡手法である nasuSORT[3]に着目し、セグメンテーションにより得られる「Mask」という物体の形状情報をより忠実に反映できる情報を導入する。Maskによって、物体の可視領域を画素単位で表現できるため、遮蔽や重なりが発生した場合でも、残存する物体領域の情報を保持できる。本研究では、このMaskに基づくMask IoUを nasuSORT のマッチング処理に組み込み、従来の BBox IoU に依存した対応付けの問題を緩和することを目的とする。

本研究の目的は、Mask IoU を加えた nasuSORT により、重なりや遮蔽、急激な方向転換や速度変化になる非線形的な移動が頻発する環境においても頑健に小動物を追跡できる手法を構築し、ID Switch や Track 消失を低減することである。

以下、2章では従来の複数小動物追跡手法とその問題点について整理する。3章に提案手法の詳細を述べる。4章では実験を通じて提案手法の有効性を検証し、小動物追跡におけるMask IoU 導入の効果について考察する。5章では本研究の結果を要約かつ分析し、今後の改善点について説明する。

2. 複数小動物追跡の従来研究

2.1. 複数物体追跡

複数物体追跡[1]とは、動画像中に存在する複数の対象物体を同時に追跡し、各物体に対して時間的に一貫した識別子を付与する技術である。追跡対象は人や車両、動物など多岐にわたり、これまでに多くの研究が行われ、様々な追跡アルゴリズムが提案されてきた。

複数物体追跡手法は、処理方法の違いから大きくバッチ型とオンライン型の二つに分類される。バッチ型手法は、動画全体あるいは複数フレームをまとめて処理することで、高い追跡精度を実現することを目的としている。一方、オンライン型手法は、現在のフレームと過去のフレーム情報のみを用いて処理を行うため、高速でリアルタイム性に優れるという特徴を持つ。しかし、バッチ型手法はリアルタイム処理が困難であり、オンライン型手法は追跡精度が低下しやすいという課題が存在していた。

これらの課題に対する解決策として、近年広く用いられるようになった手法が Tracking-by-Detection（検出に基づく追跡）である。この手法では、フレームごとに物体検出を行い、得られた検出結果に対してフレーム間で同一物体であるかを判定し、一貫した ID を割り当てることで追跡を実現する。Track と Detection の対応付けには、物体の位置情報や運動情報に加え、外観特徴量などが利用される。

Tracking-by-Detection を採用することで、オンライン型手法でありながら、高い追跡精度と処理速度の両立が可能となった。代表的な手法として、カルマンフィルタとハンガリアンアルゴリズムを用いたシンプルな位置推測と対応付けを行う SORT[2]や、外観特徴を深層学習で抽出し再識別能力を高めた DeepSORT[4]、DeepSORT の各構成要素を最新技術で最適化しさらに精度を向上させた StrongSORT[5]などの追跡手法が提案されている。他にも椋木研究室で開発した複数小動物用 SORT[6]や nasuSORT[3]がある。

Tracking-by-Detection においては、追跡性能が検出精度に大きく依存するため、高性能な物体検出器の利用が重要である。

近年では YOLO[7]などの深層学習に基づく検出器が広く用いられている。

2.2. SORT

SORT (Simple Online and Realtime Tracking) [2]は、前述の Tracking-by-Detection の枠組みにおいて、計算効率と精度のバランスに優れた代表的なアルゴリズムである。SORT は主に「カルマンフィルタによる位置予測」と「ハンガリアンアルゴリズムによる対応付け」の2つのステップで構成される。

まず、カルマンフィルタという物体の現在の位置や速度から、次の瞬間にどこにいるかを予測する数理モデルを用いて、物体の次フレームにおける位置を予測する。次に、実際に検出器 (YOLO[7]等) から得られた Detection (現在のフレームでの検出結果) と、カルマンフィルタによる予測位置を比較する。この比較の際、予測した枠と実際の検出枠がどれだけ重なっているかを示す BBox IoU を計算し、その値を類似度として利用する。

そして、ハンガリアンアルゴリズムという複数の予測と複数の検出結果を、全体の重なりが最大になるように最適に組み合わせる計算手法を用いることで、個々の物体に対して一貫した ID を割り当てる。

SORT の最大の特徴は、物体の外見の情報 (色や形など) を一切使わず、動きの滑らかさ (位置関係) のみを利用して追跡を行う点にある。これにより、非常に高速な処理が可能となっている。一方で、物体が互いに重なり合ったり (遮蔽)、小動物のように不規則で激しい動き (非線形的な移動) をしたりする場合には、予測と検出の重なりが小さくなり、追跡が失敗しやすいという性質も持っている。

2.3. nasuSORT

複数物体を同時に追跡する環境においては、物体同士の重なりや遮蔽によって検出結果が不安定となり、追跡 ID の切り替わり (ID Switch) が発生しやすいという問題がある。特に、小動物のように密集して行動する対象では、この影響が顕著であり、追跡の継続性が大きく損なわれる。こうした課題に対し、本研究の基盤となる nasuSORT[3]は、複数小動物用 SORT[6]のアルゴリズムと、物体検出の後処理技術である Soft-NMS を組み合わせることで、遮蔽に対する頑健性を向上させている。

2.3.1.複数小動物用 SORT による追跡の安定化

まず、nasuSORT の追跡機構のベースとなっている複数小動物用 SORT[6]について説明する。小動物は群れを形成して行動することが多く、個体同士が接近・接触する状況が頻繁に発生する。そのため、遮蔽による一時的な検出欠落が生じやすく、従来の SORT では Track の消失や ID の誤割り当てが起りやすかった。

複数小動物用 SORT では、遮蔽や検出漏れが発生する状況に対応するため、検出が一時的に得られなくなった場合でも追跡を継続できるように、追跡対象がその場に留まっていると仮定する「滞留」状態を導入している。滞留状態においては、カルマンフィルタによる位置予測を停止することで、不確実な予測値による状態更新を抑制し、追跡の安定性を確保している。さらに、誤検出に起因する不要な追跡 ID の新規生成を防ぐために、「二段階マッチング」が採用されている。

複数小動物用 SORT の処理の流れを図 1 に示す。

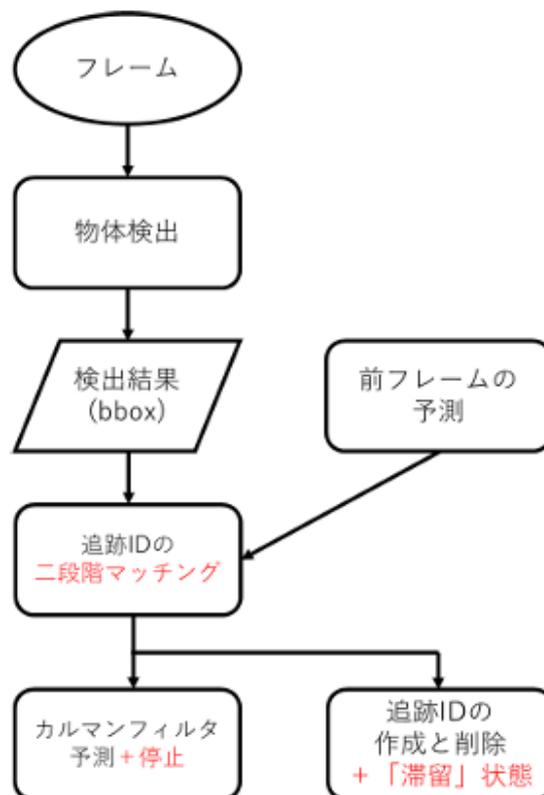


図 1 複数小動物用 SORT の流れ[3]

入力された動画像の各フレームに対して、物体検出器（例えば YOLO など）を用いて物体検出を行う。検出結果は、物体の位置および大きさを表す BBox と、検出の信頼性を示す信頼度スコアによって構成される。これらの検出結果は、あらかじめ設定した信頼度スコアの閾値に基づいて、高信頼度検出と低信頼度検出に分類される。

追跡中の各 Track に対しては、過去の追跡履歴に基づいて予測された位置と、各検出結果との重なり度合いを算出し、最も重なりの大い BBox を対応付ける。この重なり度合いの評価には IoU（Intersection over Union）が用いられる。IoU は、二つの領域の共通部分の面積をそれらの和集合の面積で割った指標であり、領域同士の重なりを定量的に表現する。

その後、二段階マッチング処理が行われる。第一段階では、高信頼度の検出結果を対象として Track との対応付けを行い、信頼性の高い対応関係を優先的に確定させる。次に、第一段階で対応付けられなかった Track に対して、低信頼度の検出結果とのマッチングを第二段階として実施し、検出漏れに対する追跡の継続性を確保している。

第二段階のマッチング処理が終了した後、対応付けが成立した追跡情報（Track）と検出（Detection）は、新たに割り当てられた BBox の位置を用いて軌跡を更新する。一方、いずれの Track とも対応付けられなかった BBox については、新規追跡 ID が割り当てられるが、その対象は信頼度スコアの高い検出に限定される。また、BBox と対応付けられなかった Track は直ちに削除されるのではなく、「滞留」状態として保持される。

次フレームに向けては、追跡対象が等速度で移動していると仮定し、これまでの軌跡情報に基づいてカルマンフィルタを用い、次フレームにおける BBox の位置および大きさを予測する。

2.3.2. Soft-NMS による検出漏れの抑制

nasuSORT のもう一つの重要な要素が、検出結果の重複を排除する後処理アルゴリズムの改良である。

従来の物体検出では、NMS (Non-Maximum Suppression : 非極大値抑制) が広く用いられている。NMS の処理の流れを図 2 に示す。

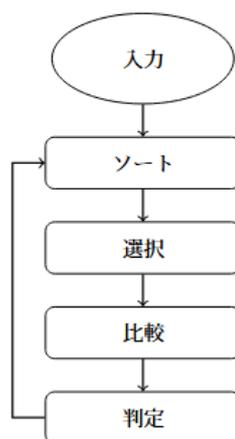


図 2 NMS

NMS では、検出器から出力されたすべての BBox とその信頼度スコア (入力) を、クラスごとに信頼度スコアの高い順に並べ替える (ソート)。次に、最も信頼度スコアの高い BBox を確定結果として選択し (選択)、その BBox と残りのすべての BBox との IoU (重なり度合い) を計算する (比較)。IoU があらかじめ設定した閾値 (例 : 0.5) 以上となる BBox は、重複検出とみなされ、強制的に削除される (判定)。この処理を、未処理の BBox がなくなるまで繰り返す (ループ)。

しかし、従来の NMS では、高い重なりを持つ複数の検出結果のうち、信頼度スコアが低いものが完全に削除される。そのため、オクルージョンが発生した場面では、本来追跡すべき物体に対応する検出が除去され、追跡が途切れる要因となっていた。

これに対し、Soft-NMS は、図 3 に示すように、NMS の処理フローにおいて「比較」と「判定」の間に信頼度スコアの減衰処理を導入した手法である。Soft-NMS では、重なり大きい検出結果を単純に削除するのではなく、IoU に応じて信頼度スコアを減衰させることで、検出結果として残す。この処理に

より、オクルージョンが発生した場合でも、物体に対応する検出候補を保持しやすくなり、検出漏れの抑制が可能となる。

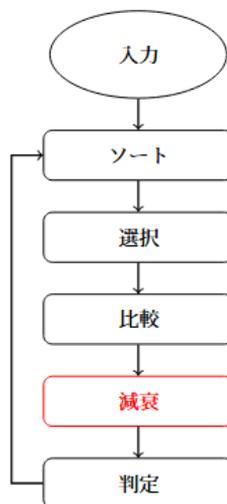


図3 Soft-NMS

2.3.3.複数小動物用 SORT と Soft-NMS の組み合わせ

Soft-NMS を導入すると、一つの物体に対して複数の BBox が出力されるリスクが生じるが、nasuSORT では前述の「二段階マッチング」を活用することでこの問題を解決している。

Soft-NMS によってスコアが低下した余分な BBox は、二段階マッチングの「第二段階」に回されるため、既存の Track との対応付けには使われるが、新規 ID の生成には使われない。その結果、誤検出を増やさずに、NMS による正しい検出の削除を防ぐことが可能となる。

以上のように、nasuSORT は、Soft-NMS による検出漏れの抑制と、複数小動物用 SORT の二段階マッチングによる誤追跡の抑制を組み合わせることで、重なりが頻発する環境においても ID Switch を削減し、追跡の継続性を向上させている。

2.4. YOLO

YOLO (You Only Look Once) [7]は、動画像や静止画像中から特定の物体を検出するための代表的な物体検出アルゴリズムである。YOLO は、画像全体を一度の推論で処理し、物体の位置とクラスを同時に推定する End-to-End 型の物体検出手法であり、高速かつ高精度な検出が可能である点に特徴がある。

従来の物体検出手法では、まず画像中から物体が存在しそうな領域を抽出し、その後、抽出された各領域を Deep Neural Network (DNN) に入力して識別を行うという二段階の処理が一般的であった。しかしこの方法では、候補領域ごとに DNN による推論を行う必要があるため、処理時間が長くなるという問題があった。

これに対し、YOLO では、物体候補領域の生成を DNN の一部として組み込み、位置推定とクラス識別を一貫して行う構成を採用している。この End-to-End な処理により、画像全体を一度だけ処理することで物体検出を行うことが可能となり、リアルタイム処理に適した高速な検出が実現されている。

YOLO による物体検出では、各 BBox に対して信頼度スコアが算出される。信頼度スコア (Confidence Score) は、「その BBox に物体が存在している確からしさ」と「BBox が物体を正確に囲んでいる度合い」を反映した値であり、さらに各クラスに属する確率と組み合わせることで物体の検出結果が表現される[8]。この信頼度スコアに基づいて、検出結果の信頼性が評価される。

YOLO の推論後には、重複した検出結果を除去するために Non-Maximum Suppression (NMS) が適用される。NMS によって、高い信頼度スコアを持つ BBox のみが最終的な検出結果として残されることで、検出の冗長性が抑制される。

本研究では、YOLO の改良版である YOLO11[9]のセグメンテーションモデル (YOLO11-seg) を検出器として用いる。YOLO11-seg は、BBox による位置情報に加え、物体の画素単位の形状を表す Mask 情報を出力できるため、遮蔽や重なりが発生する状況においても、物体の可視領域をより詳細に表現できる。一方、従来手法である nasuSORT では、検出器として YOLOv5[10]が用いられており、出力は BBox に基づく情報に限定されている。

このように、YOLO は Tracking-by-Detection 型の追跡手法において基盤となる検出器であり、その検出精度や出力形式は、後段の追跡性能に大きな影響

を与える。本研究では、セグメンテーション情報を利用可能な YOLO11-seg を採用することで、従来の BBox に基づく検出・追跡の限界を克服することを目指す。

2.5. 従来の手法の問題点

従来の多くの Tracking-by-Detection 型複数物体追跡手法では、物体の位置や大きさを BBox によって表現し、Track と Detection の対応付けを BBox 間の IoU に基づいて行っている。しかし、小動物追跡のように遮蔽や重なりが頻発する環境においては、この BBox 表現および IoU に依存した対応付けには本質的な問題が存在する。

まず、物体が部分的に遮蔽された場合や、複数の物体が互いに重なり合う場合、BBox は物体の実際の可視領域や形状を正確に反映できない。BBox は物体を外接矩形で近似するため、背景領域を多く含むことがあり、物体同士が接近すると、異なる物体であっても BBox 同士が大きく重なってしまう。この結果、IoU に基づく類似度評価において、異なる物体同士を高い類似度で誤って評価する、あるいは同一物体であっても遮蔽によって可視領域が減少することで IoU が過小評価されるといった問題が生じる。

さらに、小動物は急激な方向転換や速度変化を伴う非線形的な移動を行うことが多い。一方、従来手法では、カルマンフィルタなどにより物体が等速度で移動するという線形運動モデルを仮定して位置予測を行っている。そのため、非線形的な移動が生じた場合には、予測された Track の位置と実際の Detection の位置との乖離が大きくなる。この乖離により、正しい Detection が探索範囲外に位置する可能性が高まり、結果として BBox IoU が急激に低下する。

その結果、本来対応付けるべき Track と Detection がマッチングされず、追跡 ID の切り替わり (ID Switch) や Track の消失といった追跡性能の劣化が発生する。特に、遮蔽と非線形的な移動が同時に発生する状況では、これらの問題が連鎖的に生じ、追跡の継続性が著しく低下する。

以上のことから、BBox IoU に強く依存した従来の追跡手法は、遮蔽・重な

り・非線形的な移動が頻発する小動物追跡に対して十分に頑健であるとは言い難い。これらの課題に対しては、物体の可視領域や形状をより忠実に反映できる情報を導入することで、追跡の誤対応を一定程度抑制する手法の検討が必要である。

3. 重なり・非線形的な移動に強い小動物追跡

3.1. Mask IoU を加えた nasuSORT による小動物追跡精度の改善

前章で述べたように、従来の Tracking-by-Detection 型追跡手法では、物体の表現として BBox を用い、BBox 間の IoU に基づいて Track と Detection の対応付けを行うことが一般的である。しかし、小動物追跡のように遮蔽や重なりが頻発する環境においては、BBox が物体の実際の可視領域や形状を十分に反映できず、誤対応や ID Switch が発生しやすいという問題があった。

これに対し、本研究では Mask 情報に着目する。Mask は、物体の可視領域そのものを画素単位で表現したものである。そのため、物体が部分的に遮蔽された場合や、複数の物体が重なり合った場合であっても、遮蔽されていない残存部分の形状を正確に捉えることができる。この Mask に基づいて算出される Mask IoU は、BBox IoU と比較して、背景領域の影響を受けにくく、物体間の実際の重なり関係をより適切に評価できる指標である。

本研究では、nasuSORT を基盤とし、その二段階マッチング機構の第一段階マッチングにおいて、従来用いられていた BBox IoU の代わりに Mask IoU を導入する。これにより、追跡の初期段階において、遮蔽や重なりが発生している状況でも、形状情報に基づいた信頼性の高い対応付けが可能となる。一方、第二段階マッチングや追跡の基本構造は nasuSORT の枠組みを維持することで、既存手法の安定性を損なわない構成としている。

また、本研究では、検出器として YOLO11-seg を用い、物体の位置情報に加えて Mask 情報を取得する。YOLO11-seg の後処理は標準では NMS が用いられているが、本研究ではこれを nasuSORT の Soft-NMS に置き換えることで、遮蔽によって検出候補が削除されてしまう問題の緩和を図っている。この変更により検出候補数は増加するが、二段階マッチングおよび排他的処理を用いた対応付けと組み合わせることで、誤った追跡 ID の新規生成を抑制しつつ、追跡の継続性向上を目指している。

なお、本研究の目的は、重なり・非線形的な移動に起因する追跡過程での誤対応、特に ID Switch の発生を低減することにある。Mask IoU を第一段階マッチングに導入することにより、従来手法と比較して、重なりや遮蔽が発生する場面における対応付けの信頼性を向上させることを狙いとしている。

3.2. 提案手法の流れ

本節では、本研究で提案する小動物追跡手法の全体的な処理の流れを、図4に基づいて説明する。提案手法は、検出器としてYOLO11-segを用い、その出力結果を基にnasuSORTを拡張することで、遮蔽や重なりが発生する環境における追跡の安定性向上を目指すものである。

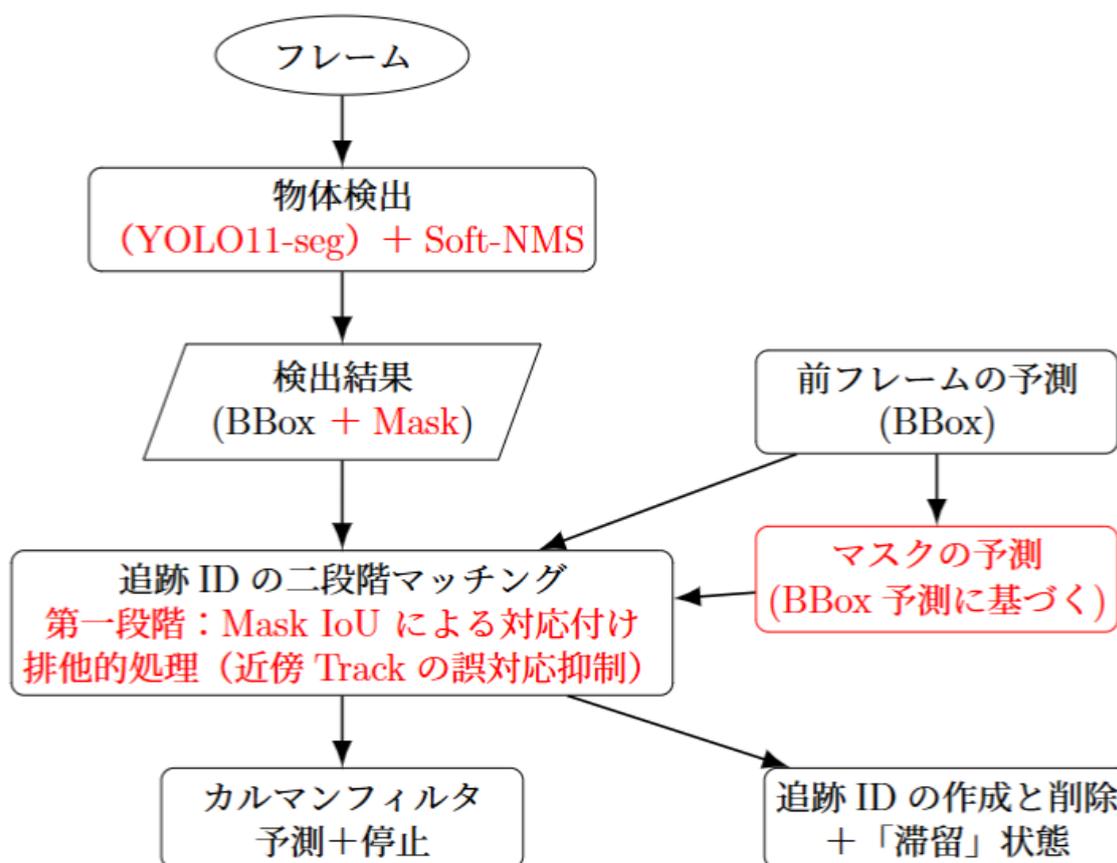


図4 nasuSORT を基盤とした提案手法の流れ

まず、各フレームに対してYOLO11-segを適用し、物体検出およびセグメンテーションを行う。これにより、各検出結果として、物体の位置を表すBBBoxに加え、物体の可視領域を画素単位で表現したMask情報(検出マスク)が得られる。YOLO11-segの標準的な後処理ではNMSが用いられるが、本研究ではこれをSoft-NMSに置き換えることで、オクルージョンによって検出候補が削除されてしまう問題の緩和を図っている。

次に、Soft-NMSを適用した検出結果をnasuSORTに入力する。nasuSORT

では、各 Track に対してカルマンフィルタを用いた位置予測が行われるが、本研究では、Mask の位置予測についても BBox の予測結果に基づいて行う。すなわち、予測された BBox の位置関係を用いて、前フレームの Mask を次フレームに対応付けることで、追跡中の物体の次フレームでの Mask (予測 Mask) を推定する。

得られた検出 Mask と予測 Mask を用いて、Track と Detection の対応付けを行う。本研究では、従来の BBox IoU と Mask IoU 両方を用いて類似度を算出し、対応関係の評価を行う。Mask IoU の具体的な計算方法については、次節において詳述する。

続いて、nasuSORT の二段階マッチング機構に基づき、Mask 情報を用いた第一段階マッチングを行う。第一段階では、Mask IoU に基づく対応付けを優先的に行い、信頼性の高い Track と Detection の対応を確定させる。一方、第二段階以降の処理は、従来の nasuSORT と同様に BBox IoU に基づく対応付けを行うことで、追跡の安定性を維持している。

さらに、本研究では、Soft-NMS により単一の物体に対して複数の Detection が出力される状況を考慮し、排他的処理を導入している。あるフレームにおいて、重複した複数の Detection の 1 つがある Track とマッチングされた場合、その Track の近傍に存在する他の Track が、残りの Detection と誤ってマッチングすることを防ぐ。これにより、複数の Track が同一の物体に対応付けられてしまう問題を抑制する。

以上の処理を各フレームに対して繰り返すことで、本研究の提案手法は、遮蔽や重なりが頻発する環境においても、追跡過程での誤対応、特に ID Switch の発生を低減することを目指す。

3.3. Mask IoU の計算

本研究では、Track と Detection の対応付けにおける類似度指標として、従来用いられてきた BBox IoU に加えて、Mask IoU を用いる。Mask IoU は、物体の可視領域を画素単位で表現した Mask 同士の重なり度合いを評価する指標であり、遮蔽や重なりが発生する状況においても、物体形状に基づいたより適切な類似度評価が可能となる。

3.3.1. BBox IoU の定義

まず、比較のために BBox IoU を定義する。二つの BBox B_1, B_2 に対する IoU は、式(1)で表される。

$$\text{IoU}_{\text{bbox}}(B_1, B_2) = \frac{|B_1 \cap B_2|}{|B_1 \cup B_2|} \quad (1)$$

ここで、 $| \cdot |$ は領域の面積を表す。BBox IoU は計算が容易である一方、物体の形状を外接矩形で近似するため、背景領域を多く含んでしまうという問題がある。特に、物体同士が接近・重なり合う場合や、物体が部分的に遮蔽される場合には、実際の可視領域とは乖離した評価となりやすい。

3.3.2. Mask IoU の定義

YOLO のセグメンテーションでは、各物体に対する Mask を直接予測するのではなく、Prototype Mask (基底 Mask) と Mask Coefficient (重み係数) を用いた合成的な表現を採用している [11]。本研究でもこの構造に基づき、実際に扱う Mask を生成する。

YOLO のセグメンテーションヘッドは図5で示したように、入力画像全体に対して 32 枚の Prototype Mask (P_k) を出力する。各 Prototype Mask は画像全体を覆う 160×160 の二次元行列であり、個々の物体に固有のマスクではなく、形状表現の基底として機能する。一方で、各 Detection i に対して、32 次元の Mask Coefficient (\mathbf{c}_i) が予測される。これらの係数は、「どの Prototype をどの程度組み合わせるか」を表す重みであり、各物体に固有である。

```
Masks P3: [1, 32, 80, 80]      # 32 mask coefficients
Masks P4: [1, 32, 40, 40]
Masks P5: [1, 32, 20, 20]

Protos: [1, 32, 160, 160]     # 32 prototype masks
```

図5 YOLO のセグメンテーションの出力

検出 i に対する生の Mask (M_i) は、Prototype Mask の線形結合として式(2)で定義される。

$$M_i(x, y) = \sum_{k=1}^{32} c_{ik} \cdot P_k(x, y) \quad (2)$$

この処理により、32枚の基底Maskを重み付きで合成した連続値のMaskが得られる。この時点では、各画素の値は実数であり、物体であるかどうかの確率的なスコアを表す。この計算は、行列積として式(3)のようにも表現できる。

$$M_i = c_i^T P \quad (3)$$

ここでPはPrototype Maskを縦に並べたテンソルである。

次に、線形結合後の生Mask(M_i)を正規化するためには式(4)の活性化関数(Sigmoid)を適用する。

$$\hat{M}_i(x, y) = \sigma(M_i(x, y)) \quad (4)$$

この処理により、各画素の値は(0, 1)の範囲に正規化され、 $\hat{M}_i(x, y)$ は「画素(x, y)が物体領域である確率」を表す値として解釈できる。

また、Prototype Maskは画像全体に対して生成されるため、Sigmoid後のMaskには背景領域も含まれる。そこで、検出された物体のBBBox(B_i)を用いて、対応する領域のみを切り出す。

$$\hat{M}_i^{\text{crop}} = \hat{M}_i \cap B_i \quad (5)$$

式(5)の処理により、Maskは検出された物体領域に対応した局所的なMaskとなる。

最後に、本研究では、追跡におけるMask IoUの計算を目的として、Maskを二値化する。所定の閾値 τ ($\tau = 0.5$)を用いて、最終的に扱うMask(M_i^{bin})を式(6)のように定義する。

$$M_i^{\text{bin}}(x, y) = \begin{cases} 1 & \hat{M}_i^{\text{crop}}(x, y) \geq \tau \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

ここで、

- 値1は物体領域
- 値0は背景領域

を表す。

Mask IoUは、セグメンテーションによって得られるMask同士(M_i)の重なりを直接評価する指標である。二つのMask(M_1, M_2)に対するMask IoUは、式(7)で定義される。

$$\text{IoU}_{\text{mask}}(M_1, M_2) = \frac{|M_1 \cap M_2|}{|M_1 \cup M_2|} \quad (7)$$

ここで、 M_1, M_2 はそれぞれ画素集合として表現されるMaskであり、図6に示したように $|M_1 \cap M_2|$ は両Maskに共通する画素数、 $|M_1 \cup M_2|$ はいずれかの

Mask に含まれる画素数を表す。

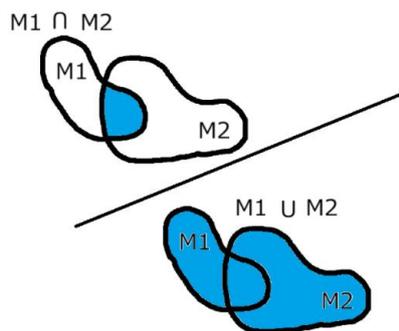


図 6 Mask IoU

Mask IoU は、背景画素を含まず、物体の可視領域のみを対象として計算されるため、物体間の実際の重なり関係をより忠実に反映できる。

3.3.3. 遮蔽・重なり環境における利点

物体が部分的に遮蔽された場合、図 7 に示したように、BBox は遮蔽された領域や背景を含んだまま大きく変形するため、BBox IoU は急激に低下しやすい。一方、図 8 に示したように、Mask IoU は、遮蔽されていない可視部分のみを用いて評価を行うため、同一物体に対する類似度を相対的に高く保つことができる。

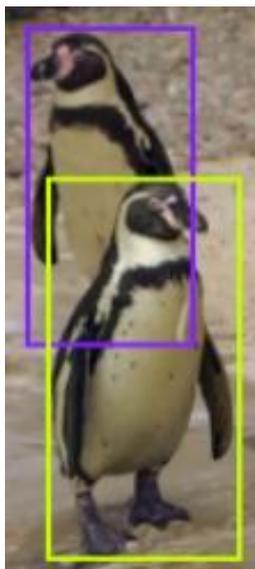


図 7 遮蔽時における BBox の表現



図 8 遮蔽時における Mask の表現

また、複数の物体が近接・重なり合う場面では、BBBox 同士が大きく重なってしまい、異なる物体間で高い BBBox IoU が算出されることがある。Mask IoU では、各物体の形状に基づいて重なりが評価されるため、異なる物体同士の誤った高類似度評価を抑制できる。

3.4. Mask の予測

本研究では、Track と Detection の対応付けに Mask IoU を用いるため、検出 Mask だけでなく、追跡中の Track に対する Mask の予測が必要となる。しかし、nasuSORT においては、カルマンフィルタによる予測は BBBox の位置と大きさに対して行われており、Mask そのものを直接予測する機構は備えていない。

そこで本研究では、BBBox のカルマンフィルタ予測結果を利用して、Mask の位置を平行移動することで Mask 予測を行う手法を採用する。具体的には、前フレームにおいて得られた Track の Mask を、カルマンフィルタによって予測された BBBox の移動量に基づいて平行移動させることで、次フレームにおける Mask を推定する。

いま、前フレームにおける BBBox の中心座標を (x_{t-1}, y_{t-1}) 、カルマンフィルタによって予測された次フレームの中心座標を (x_t, y_t) とすると、移動量 $(\Delta x, \Delta y)$ は式(8)で表される。

$$\Delta x = x_t - x_{t-1}, \Delta y = y_t - y_{t-1} \quad (8)$$

この移動量を用いて、前フレームの Mask (M_{t-1}) を式(9)のように平行移動することで、予測 Mask (M_t^{pred}) を得る。

$$M_t^{\text{pred}}(x, y) = M_{t-1}(x - \Delta x, y - \Delta y) \quad (9)$$

この処理により、物体の形状自体は変化しないと仮定した上で、位置のみを予測する簡易的な Mask 予測が実現される。

本手法では、Mask の回転や変形といった高次の運動は考慮していないが、小動物追跡においては、短時間 (1 フレーム間) での物体形状変化は限定的である場合が多い。そのため、位置の平行移動のみに基づく Mask 予測であっても、次フレームの検出 Mask との類似度評価に十分な精度を確保できると考えられる。

得られた予測 Mask は、次節以降で述べる、Mask による第一段階マッチングにおいて、検出 Mask との Mask IoU 計算に用いられる。これにより、遮蔽や検出の一時的な不安定さが生じた場合であっても、Track 側の Mask 情報を保持したまま対応付けを行うことが可能となる。

なお、本研究における Mask 予測は、BBox のカルマンフィルタ予測に依存しているため、予測精度は BBox 予測の精度に影響を受ける。本手法は Mask 予測を次フレームにおける完全な形状推定として扱うものではなく、マッチングにおける補助的な情報として利用することを目的としている。

3.5. Mask による第一段階マッチング

nasuSORT では、追跡の安定性を高めるために、Track と Detection の対応付けを二段階マッチングによって行っている。本研究ではこの枠組みを踏襲しつつ、第一段階マッチングにおける類似度指標として Mask IoU を導入する。

3.5.1. 第一段階マッチングの目的

第一段階マッチングは、信頼性の高い Detection を用いて、対応関係が明確な Track と Detection の組を優先的に確定させる段階である。従来の nasuSORT では、この段階において BBox IoU が用いられていたが、遮蔽や重なりが発生する状況では、BBox に基づく類似度評価が不安定となり、誤対応や未対応が生じやすいという問題があった。

そこで本研究では、物体の可視領域を直接反映する Mask 情報に基づく Mask IoU を用いることで、第一段階における対応付けの信頼性向上を図る。

3.5.2. Mask IoU に基づく類似度計算

フレーム t において、既存の Track 集合を $\mathcal{T} = \{T_1, T_2, \dots, T_N\}$ 、検出結果の集合を $\mathcal{D} = \{D_1, D_2, \dots, D_M\}$ とする。各 Track T_i に対しては、前節で述べた方法により予測 Mask M_i^{pred} が得られている。また、各 Detection D_j に対しては、YOLO11-seg によって生成された検出 Mask M_j^{det} が与えられる。

Track T_i と Detection D_j の類似度は、式(10)で定義される $\text{IoU}_{\text{mask}}(T_i, D_j)$ によ

って評価される。

$$\text{IoU}_{\text{mask}}(T_i, D_j) = \frac{|M_i^{\text{pred}} \cap M_j^{\text{det}}|}{|M_i^{\text{pred}} \cup M_j^{\text{det}}|} \quad (10)$$

ここで、 $|\cdot|$ は Mask に含まれる画素数を表す。

3.5.3. コスト行列の構築

第一段階マッチングでは、上記の Mask IoU を用いて、Track と Detection 間のコスト行列Cを構築する。本研究では、類似度が高い組ほど低コストとなるよう、式(11)によりコストを定義する。

$$C_{i,j} = 1 - \text{IoU}_{\text{mask}}(T_i, D_j) \quad (11)$$

このコスト行列 $C \in \mathbb{R}^{N \times M}$ を用いて、Track と Detection の最適な対応付けを求める。

3.5.4. 第一段階マッチングのアルゴリズム

第一段階マッチングは、信頼度スコアが高い Detection を対象として行われる。以下に、本研究で用いる第一段階マッチングの処理手順を示す。

アルゴリズム：Mask による第一段階マッチング

1. 各 Track T_i について、予測 Mask M_i^{pred} を生成する
2. 各 Detection D_j について、検出 Mask M_j^{det} を取得する
3. Mask IoU に基づき、コスト行列Cを構築する
4. コストが所定の閾値以下となる組のみを候補として残す
5. ハンガリアン法を用いて、コスト最小となる対応付けを求める
6. 対応付けられた Track と Detection の組を第一段階マッチング結果として確定する

3.5.5. 本研究における意義

Mask IoU に基づく第一段階マッチングにより、遮蔽や重なりが発生している状況においても、物体形状の一致度に基づいた対応付けが可能となる。これにより、

- 異なる物体同士の誤対応の抑制
- 同一物体であるにもかかわらず対応付けに失敗する事例の減少

が期待できる。

一方で、Mask IoU は Mask 予測精度や検出 Mask の品質に依存するため、単独で完全な対応付けを保証するものではない。そのため、本研究では、第一段階マッチングに排他的処理を組み合わせることで、全体としての追跡安定性向上を図っている。

次節では、第一段階マッチングで確定した対応関係を基に適用される排他的処理について詳述する。

3.6. 排他的処理

本研究では、Soft-NMS の導入により、遮蔽や重なりが生じる状況においても検出候補を多く保持する構成としている。一方で、この構成では、単一の実物体に対して複数の Detection が同時に出力される可能性が高まり、複数の Track が同一物体に引き寄せられてしまう問題が生じる。このような状況は、誤追跡や ID Switch の発生につながるため、追加の抑制機構が必要となる。

そこで本研究では、第一段階マッチングの結果を基に排他的処理を導入する。排他的処理とは、ある Detection が特定の Track と対応付けられた場合に、その Detection と同一物体に由来すると考えられる他の Detection に対して、他の Track が対応付けを行いにくくするための制約処理である。

具体的には、第一段階マッチングにおいて、Track T_l が Detection D_j と対応付けられたとする。このとき、検出集合内の他の Detection D_k に対して、式(12)により Mask の重なり度合いを評価する。

$$\text{IoU}_{\text{mask}}(D_j, D_k) \geq \theta_{\text{ex}} \quad (12)$$

ここで、 $\text{IoU}_{\text{mask}}(D_j, D_k)$ は Detection 同士の Mask IoU、 θ_{ex} は排他性判定のための閾値である。この条件を満たす Detection は、同一物体に由来する可能性が高いとみなし、排他的制約の対象とする。

排他的処理では、これらの Detection に対して、他の Track が対応付けを行う際のコストを増加させる。すなわち、Track T_l (ただし $l \neq i$) と Detection D_k の対応コスト $C_{l,k}$ を、式(13)のように修正する。

$$C'_{l,k} = C_{l,k} + \lambda \quad (13)$$

ここで、 λ は排他的ペナルティ係数であり、対応付けを完全に禁止するので

はなく、「選ばれにくくする」ための緩やかな制約を与える役割を持つ。この設計により、必要に応じて対応付けが行われる余地を残しつつ、誤対応を抑制できる。

この排他的処理の効果を、簡単な数値例で説明する。いま、Track T_1 が Detection D_1 と対応付けられ、 D_1 と D_2 の Mask IoU が 0.85 とする。また、閾値 $\theta_{ex} = 0.7$ とする。このとき、 D_2 は同一物体に由来すると判断される。仮に、別の Track T_2 と D_2 の元の対応コストが $C_{2,2} = 0.15$ であった場合、排他的処理を適用すると、 $C'_{2,2} = 0.15 + \lambda$ となり、例えば $\lambda = 0.5$ と設定すると、 $C'_{2,2} = 0.65$ となる。これにより、 T_2 が D_2 に対応付けられる可能性は大きく低下する。

以上の処理を擬似コードで表すと、次のようになる。

```
for ( $T_i, D_j$ ) in matched_pairs_stage1:
    for  $D_k$  in detections:
        if MaskIoU( $D_j, D_k$ ) >= theta_ex:
            for  $T_l$  in tracks where  $T_l \neq T_i$ :
                 $C[T_l, D_k] += \lambda$ 
```

この排他的処理は、第一段階で確定した対応関係を尊重しつつ、Soft-NMS によって生じる冗長な Detection の影響を抑制する役割を果たす。元々の二段階マッチングでは、Soft-NMS により Detection の信頼度にペナルティを付けることでマッチングの優先順位を変えていた。一方、この排他的処理では、対応コストにペナルティを付けることで誤対応を防ぐ。その結果、複数の Track が単一物体に誤対応する現象や、遮蔽後に誤った ID が付与される問題を低減できる。

なお、本処理は排他性を厳密に強制するものではなく、コスト増加という形で制約を与える設計としている。そのため、環境や状況に応じて柔軟な対応付けが可能であり、全体として追跡の安定性向上に寄与する。

4. 評価実験

本章では、本研究で提案した Mask IoU を加えた nasuSORT による小動物追跡手法の有効性を検証するために行った評価実験について述べる。はじめに実験に用いたデータセットと検出器の学習方法を説明し、次に比較手法および評価指標を示す。その後、実験結果を定量的に分析し、最後に得られた知見について考察を行う。

4.1. 実験データ

本研究では、小動物追跡を対象とした実環境に近い映像を用いて評価を行うため、動画共有サイト Pexels[12]に公開されている無料のペンギン群動画を収集した。収集した動画は合計 20 本である。一本の動画に対して、15 フレームおきに 1 フレームを学習データにした。集めた学習データは 304 枚の画像となった。これらの画像に対して、データセット作成およびアノテーション管理を容易にするため、Roboflow[13]を用いてアノテーションを行い、物体検出およびセグメンテーション学習用のデータセットを構築した。Roboflow 上では、ペンギン個体に対して BBox および Mask のアノテーションを付与し、学習に適した形式でデータセットをエクスポートした。

評価実験には、学習に使用していない別のペンギン動画 3 本を用いた。各動画は、カメラ条件や被写体の配置が異なり、追跡難易度に差がある構成となっている。

各テスト動画の特徴を以下に示す。

- 動画① (図 9)

カメラ非固定 (手持ち撮影による揺れあり)。近距離から数羽のペンギンを撮影しており、個体同士の重なりは少ない。



図 9 動画①

- 動画② (図 10)

カメラ固定. 遠景から撮影しており, 重なりがやや発生し, 一部のペンギンが水面に浮かぶ場面が存在する。



図 10 動画②

- 動画③ (図 11)

カメラ動作あり (パン動作)。遠方の群れが接近してくる過程で, 岩などペンギンの色に近い背景が存在して, 個体間の重なりが頻発する。



図 11 動画③

これらの動画は, 遮蔽・重なり・カメラ動作といった小動物追跡における典型的な困難要因を含んでいる。

4.2. 検出器の学習

物体検出器として、本研究では2種類のYOLO系モデルを使用した。

1つ目は、汎用的な物体検出用としてYOLOv5をベースに、前節でRoboflowを用いて作成したBBoxアノテーション付きデータセットを用いて学習したペンギン検出器である。2つ目は、セグメンテーション用としてYOLO11-segをベースに、同様にRoboflow上で作成したMaskアノテーション付きデータセットを用いて学習したペンギン検出器である。

いずれの検出器も、前述の20本の動画から合計304枚のフレームを抽出した学習用データセットを、教師データとして用いて学習を行った。

4.3. 実験設定

本研究では、小動物追跡における提案手法の有効性を検証するため、前節で述べた3本のテスト動画に対して複数物体追跡を適用し、各手法の追跡性能を比較した。すべての手法において、同一の入力動画を用いることで、追跡アルゴリズムの違いによる影響を公平に評価できるように設定した。

比較対象として、以下の6手法を用いた。実験手法の設定を表1に示す。

- nasuSORT (二段階マッチング + Soft-NMS)
- YOLO11-tracker (Ultralytics 実装の ByteTrack ベーストラッカ)
- My_nms_bbox (BBox IoU のみを用いた追跡, Mask 未使用)
- My_nms_mask (第一段階マッチングを Mask IoU を用いた追跡, 後処理は NMS)
- My_soft-nms_mask (Mask IoU を用いた追跡, Soft-NMS)
- My_soft-nms_mask_exclusivity (提案手法)

表 1 実験手法の設定

	二段階マッチング	Soft-NMS	Mask IoU	排他的処理
nasuSORT	✓	✓		
YOLO11-tracker	✓			
My_nms_bbox	✓			
My_nms_mask	✓		✓	
My_soft-nms_mask	✓	✓	✓	
提案手法 (*)	✓	✓	✓	✓

(*) My_soft-nms_mask_exclusivity

ここで、My_nms_bbox は Mask を使用しない従来型の BBox ベース追跡手法であり、My_nms_mask および My_soft-nms_mask は、提案手法の要素技術の一部を取り除いた Ablation 手法として位置付けられる。My_soft-nms_mask_exclusivity が、本研究で提案する完全な手法である。

評価指標としては、追跡 ID の切り替わり回数を表す ID Switch (IDS) と、追跡中に新たに生成された追跡 ID の数を表す Track ID 生成数 (TrackID) の 2 種類を用いた。IDS は、追跡中に同一物体に対して異なる ID が割り当てられた回数を示す指標であり、追跡の一貫性を評価するために用いられる。この実験では、IDS 数は目視で確認した。一方、TrackID は、各動画において最初に存在する個体数を基準とし、追跡途中で新規に割り当てられた ID の総数を表す指標である。TrackID が多い場合、追跡の途中でトラックが途切れ、新規トラックが頻繁に生成されていることを意味する。

すべての手法において、追跡処理はフレーム単位でオンラインに実行し、追跡結果から IDS および TrackID を集計した。これらの指標を用いて、各手法の追跡安定性および遮蔽・重なり環境における頑健性を定量的に評価した。

4.4. 実験結果

前節で述べた実験設定に基づき、各手法をテスト動画①～③に適用した結果を表2～4に示した。評価指標としては、追跡IDの切り替わり回数を表すID Switch (IDS) および、追跡中に新規に生成されたTrack IDの数 (TrackID) を用いる。

表2 動画1の結果

手法	IDS	TrackID
NasuSORT (StrongSORT+Soft-NMS)	9	13
YOLO11-tracker (ByteTrack)	2	6
My_nms_bbox (Mask 未使用)	2	6
My_nms_mask (提案: Mask IoU)	1	5
My_soft-nms_mask	1	5
My_soft-nms_mask_exclusivity	1	5

表3 動画2の結果

手法	IDS	TrackID
NasuSORT (StrongSORT+Soft-NMS)	3	7
YOLO11-tracker (ByteTrack)	8	17
My_nms_bbox (Mask 未使用)	21	30
My_nms_mask (提案: Mask IoU)	6	15
My_soft-nms_mask	5	14
My_soft-nms_mask_exclusivity	5	14

表4 動画3の結果

手法	IDS	TrackID
NasuSORT (StrongSORT+Soft-NMS)	19	21
YOLO11-tracker (ByteTrack)	15	21
My_nms_bbox (Mask 未使用)	13	19
My_nms_mask (提案: Mask IoU)	9	13
My_soft-nms_mask	9	13
My_soft-nms_mask_exclusivity	7	12

動画①（近距離・重なりが少ない環境）

動画①は、カメラが非固定であるものの、撮影距離が近く、個体同士の重なりが少ない比較的追跡が容易な条件である。この動画において、従来法である nasuSORT は IDS が多く発生しており、ID Switch が頻発している。一方で、提案手法では IDS が大幅に低減されており、追跡 ID の一貫性が向上していることが確認できる。

YOLO11-tracker や Mask を使用しない My_nms_bbox においても IDS は比較的低い値に抑えられているが、TrackID の観点では提案手法が最も少ない値を示している。これは、提案手法が Mask IoU の使用により、一度割り当てた追跡 ID を安定して維持し、新規トラックの生成を抑制できていることを示している。

動画②（遠景・水面に浮かぶ場合がある）

動画②では、カメラが固定されている一方で、遠景から複数のペンギンを撮影しており、重なりが時々発生し、一部のペンギンが水面に浮かびながら移動する状況が存在する。この動画において、nasuSORT は IDS および TrackID が一見低い値を示しているが、これは水面上に浮かんでいるペンギンをほとんど検出できておらず、追跡対象として認識された個体数自体が少なかったことに起因する。したがって、この結果は必ずしも追跡性能の高さを反映したものではない。

一方、YOLO11-tracker では IDS および TrackID が増加しており、個体数が多い状況では誤対応付けやトラックの途切れが発生しやすいことが分かる。Mask を使用しない My_nms_bbox では、さらに IDS および TrackID が増加しており、BBox IoU に強く依存した追跡が、重なりを含む環境において不安定であることが示唆される。

これに対し、Mask IoU を用いた手法では、全体的に IDS および TrackID が抑えられている。特に、Soft-NMS と Mask IoU を組み合わせた手法では、検出漏れを抑制しつつ、誤った新規 ID の生成を減らすことができている。提案手法は、動画②においても他の手法と比較して安定した追跡結果を示している。

動画③（遠距離・遮蔽および重なり多発）

動画③は、カメラのパン動作に加え、ペンギンの色に近い背景が存在し、個体間の重なりが頻発する、最も追跡難易度の高い条件である。この動画において、nasuSORT および YOLO11-tracker は、いずれも IDS および TrackID が大きく増加しており、頻繁な ID Switch や Track 分断が発生していることが確認できる。

Mask を使用した手法では、全体として追跡性能の向上が確認され、特に Mask IoU を用いることで、重なり合った個体間における誤対応付けが効果的に抑制されている。後処理として NMS または Soft-NMS を用いた場合、IDS および TrackID 数自体には大きな差が見られないように見えるが、両者には追跡挙動に明確な違いが存在する。具体的には、NMS を用いた場合には遮蔽が発生した際に Track が消失しやすいのに対し、Soft-NMS を用いた場合には、複数の Track が単一の真の物体に収束してしまう傾向が見られた。

提案手法では、Soft-NMS に加えて排他的処理を導入することで、複数の Track が同一物体に収束する現象を抑制している。その結果、最も混雑度の高い動画③においても、IDS および TrackID 数を他手法より低い値に抑えることができおり、遮蔽や重なりが頻発する環境において有効であることが確認できる。

以上の結果より、Mask IoU を用いた対応付けは、BBox IoU に基づく従来手法と比較して、重なりや遮蔽が生じる状況において特に有効であることが示された。また、Soft-NMS の導入によって検出漏れを抑制し、さらに排他的処理を組み合わせることで、誤対応付けや不要な新規 Track の生成を効果的に低減できることが確認された。

これらの要素を統合した提案手法は、すべての動画において IDS および TrackID の両面で安定した性能を示しており、小動物追跡における追跡の一貫性および継続性を向上させる手法であるといえる。

4.5. 考察

複数の動画を用いて評価を行った結果、提案手法は従来の nasuSORT と Ultralytics 実装の ByteTrack ベーストラックと比較して、すべての動画において ID Switch 数を削減できていることを確認した。特に、遮蔽や重なりが頻発する条件下では、Mask IoU を用いた第一段階マッチングの導入により、BBox IoU のみを用いた場合よりも安定した対応付けが可能であった。

また、Soft-NMS により検出数が増加した場合においても、Mask を用いたマッチングおよび排他的処理を適用することで、単一物体に対する複数 Track の生成や誤対応を抑制できていることが確認された。これにより、検出漏れを防ぎつつ、追跡 ID の安定性を維持できたと考えられる。

以上の結果から、本研究で提案した Mask を用いた対応付けと排他的処理は、遮蔽や重なりが発生する小動物追跡において、ID Switch の削減に有効であり、追跡精度の向上に寄与したと考えられる。一方で、強い遮蔽が発生する状況では依然として ID Switch が発生しており、完全な解決には至っていない。

5. おわりに

本研究では、遮蔽や重なり・非線形的な移動が頻発する小動物追跡を対象として、nasuSORT を基盤に Mask IoU、および排他的処理を導入した追跡手法を提案し、その有効性を評価実験により検証した。

具体的には、Mask IoU を用いた第一段階マッチングを導入し、重なりが生じる状況において、従来の BBox IoU に基づく手法よりも安定した対応付けを可能にした。BBox は物体の外接矩形であるため、可視領域が部分的に失われた場合や複数物体が密集した場合には、実際の物体間関係を正確に反映できない。一方、Mask は物体の可視領域を画素単位で表現するため、Mask IoU による類似度評価は、実際の重なり関係をより適切に捉えることができる。この効果は、特に動画③のように重なりと非線形的なが同時に発生する環境において顕著であり、ID Switch の大幅な削減につながったと考えられる。さらに、本研究で導入した排他的処理は、Soft-NMS によって生じやすい「単一物体に対する複数 Track の収束」という問題を抑制する役割を果たした。実験結果からも、排他的処理を加えた手法は、特に混雑度の高い動画③において、IDS および TrackID の双方を低減できている。これは、Mask IoU による局所的な類似度評価に加え、近傍 Track 間の競合を明示的に制御したことが、誤対応付けの抑制につながったためであると考えられる。

一方で、本研究の提案手法は、重なりと非線形的な移動に起因する追跡誤りを完全に解決したわけではない。実験結果においても、最も困難な条件である動画③では、依然として一定数の ID Switch が発生している。これは、Mask 自体が検出結果に依存しており、強い遮蔽や検出精度の低下が生じた場合には、Mask IoU による対応付けでも十分な識別力が得られないことが要因の一つと考えられる。また、Mask の予測を BBox の平行移動によって近似している点も、形状変化や姿勢変化が大きい場合には誤差を生じる可能性がある。

以上より、本研究の提案手法は、小動物追跡における ID Switch や不要な Track ID 生成を削減する点で有効であることが示されたが、追跡誤りを根本的に解消するには至っていない。今後の課題としては、Mask 予測の高精度化や、外観特徴や時系列的な Mask 変化を考慮した対応付けの導入が考えられる。また、より多様な環境や対象動物に対する評価を行うことで、提案手法の汎用性と限界を明らかにする必要がある。

謝辞

本研究を遂行するにあたり，多大なるご指導とご助言を賜りました指導教員の椋木雅之教授に，心より感謝申し上げます。研究の方向性から論文構成に至るまで，常に的確なご指摘と温かいご指導をいただき，本研究を最後まで完成させることができました。

また，本研究の基礎となる先行研究を通じて多くの示唆を与えてくださった，長友祐磨先輩および奈須泰樹先輩に深く感謝いたします。お二人の研究成実は，本研究の着想および手法検討において大きな指針となりました。

さらに，日頃より研究活動を共にし，議論や助言を通じて多くの刺激を与えてくださった研究室の皆様へ感謝いたします。皆様との意見交換は，本研究を進める上で大きな支えとなりました。

加えて，本研究で使用したペンギン動画データを無償で提供してくださった制作者の方々，ならびにそれらの動画を公開・共有する場を提供しているウェブサイトに対し，深く感謝の意を表します。これらの貴重なデータがなければ，本研究を実施することはできませんでした。

また，これまで私を支え，学業に専念できる環境を整えてくれた両親に深く感謝いたします。生んでくれてありがとうございます。

最後に，本研究を支えてくださったすべての方々に，心より御礼申し上げます。

参考文献

- [1] MOT アルゴリズムの歴史について
<https://tech.acesinc.co.jp/entry/2021/11/08/133336>
- [2] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, Ben Upcroft, 『Simple Online and Realtime Tracking』 2016 IEEE International Conference on Image Processing (ICIP)
- [3] 奈須 泰樹, 『Soft-NMS を追加した SORT による重なりに強い複数物体追跡』 令和 5 年度 宮崎大学工学部 卒業論文(2023)
- [4] Nicolai Wojke, Alex Bewley, Dietrich Paulus, 『Simple Online and Realtime Tracking with a Deep Association Metric』 2017 IEEE International Conference on Image Processing (ICIP)
- [5] Yunhao Du, Zhicheng Zhao, Yangzhou Song, Yanyun Zhao, Fei Su, Tao Gong, Hongying Meng, 『StrongSORT: Make DeepSORT Great Again』 IEEE Transactions on Multimedia, 2023
- [6] 長友 祐磨, 『滞留状態と二段階マッチングを追加した SORT による複数小物体追跡の同時追跡』 令和 4 年度 宮崎大学工学部 卒業論文(2022)
- [7] 【物体検出手法の歴史 : YOLO の紹介】
https://qiita.com/cv_carnavi/items/68dcda71e90321574a2b
- [8] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi 『You Only Look Once: Unified, Real-Time Object Detection』 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
- [9] YOLO11
<https://docs.ultralytics.com/models/yolo11/>
- [10] YOLOv5
<https://docs.ultralytics.com/models/yolov5/>
- [11] Mihajlov, N., “yolo-custom-segmentation”,
<https://github.com/mihajlov39547/yolo-custom-segmentation>, (参照 2026/01)
- [12] Pexels, <https://www.pexels.com/zh-cn/search/videos/penguin/>, (参照 2025/06)
- [13] Roboflow, “Roboflow: Give your software the power to see objects in images and video,” <https://roboflow.com>