エッジAI向け物体検出モデル作成における アノテーション効率化手法

森 裕輝[1]　浅井 悠佑[1]　東浦 圭亮[1]　片山 晋[1]　浦野 健太[1]　米澤 拓郎[1]　河口 信夫[1,2]

概要：近年，物流に対する世界的な需要が拡大し続けており，倉庫内業務の効率化は重要な課題である．倉庫内業務の効率化に向け，大規模カメラ基盤を構築し，その映像データを基に倉庫内オブジェクトの位置や動きのデータ化を試みている．しかし，すべての映像データをクラウドに送信し，処理を行うことは，膨大な通信量や処理電力がかかる．エッジ AI カメラの利用により，映像をカメラ内で分析・データ抽出，必要な情報のみを転送し，通信量と消費電力を大幅に削減できる．しかし，エッジ AI カメラの計算能力の制約と倉庫内の複雑な環境下という条件より，軽量で高精度な物体検出モデルが必要となる．エッジ AI カメラごとに物体検出モデルを作成すれば，高精度を期待できるが，アノテーションコストは膨大になる．本研究では，学習データ生成時においてもエッジ側で処理し，通信量・ストレージ量を抑えつつ，アノテーションを効率化する方法を提案する．具体的には，エッジ AI カメラを用いて，倉庫内の映像からオプティカルフローを用いて動的物体・静的物体の位置情報を取得し，その位置情報を基に表現学習を用いて倉庫内オブジェクトをセグメンテーションした．その後，サーバ上でクラスタリングを行い類似オブジェクトを一つのクラスタにまとめ，クラスタごとに一括でラベリングしてアノテーションコストを大幅に削減した．また，セグメンテーションまでをエッジ AI カメラ上で実行し，学習データ生成時においても通信量・ストレージ量を抑えた．評価実験として，提案手法で収集したデータを YOLOv8n を用いて学習した．結果として，通信量・ストレージ量を抑えつつ，物体検出モデル作成に必要なアノテーションコストを著しく削減した．

An Efficient Annotation Method for Object Detection Modeling in Edge AI Architecture

YUKI MORI[1]　YUSUKE ASAI[1]　KEISUKE HIGASHIURA[1]　SHIN KATAYAMA[1]　KENTA URANO[1]
TAKURO YONEZAWA[1]　NOBUO KAWAGUCHI[1,2]

1. はじめに

電子商取引（EC：Electronic Commerce）の普及に伴い，物流市場の拡大，倉庫面積の拡大が続いている [1], [2]．さらに，少子高齢化に伴う労働力不足や需要増加に伴う人手不足に陥っている．これにより，物流倉庫内業務の激務化が問題となり，業務の効率化が重要な課題となっている．物流倉庫の業務効率化という課題に対し，ロボットの最適経路探索を用いた効率化 [3] や倉庫レイアウト最適化 [4] など様々な取り組みが行われている [5], [6], [7]．その中でも，デジタルツインを用いたアプローチが注目されている [8]．デジタルツインとは，リアル空間にある情報を IoT などで収集し，それらの情報をもとに仮想空間上で人やモノの動きをリアルタイムでシミュレートする技術である [9]．デジタルツインの利用により，効率化に向けた実験を低コストで仮想的に実施でき，課題に対する有望な解決策を提供できる．しかし，デジタルツインの構築には物理空間の正確なデジタル化が必要であり，物理空間をセンシングし，得られたデータを抽出しなければならない．

我々は，物流倉庫にて，60 台以上の定点カメラで構成される大規模カメラ基盤を構築し，その映像データを基に

[1] 名古屋大学大学院 工学研究科
Graduate School of Engineering, Nagoya University
[2] 名古屋大学 未来社会創造機構
Institutes of Innovation for Future Society, Nagoya University

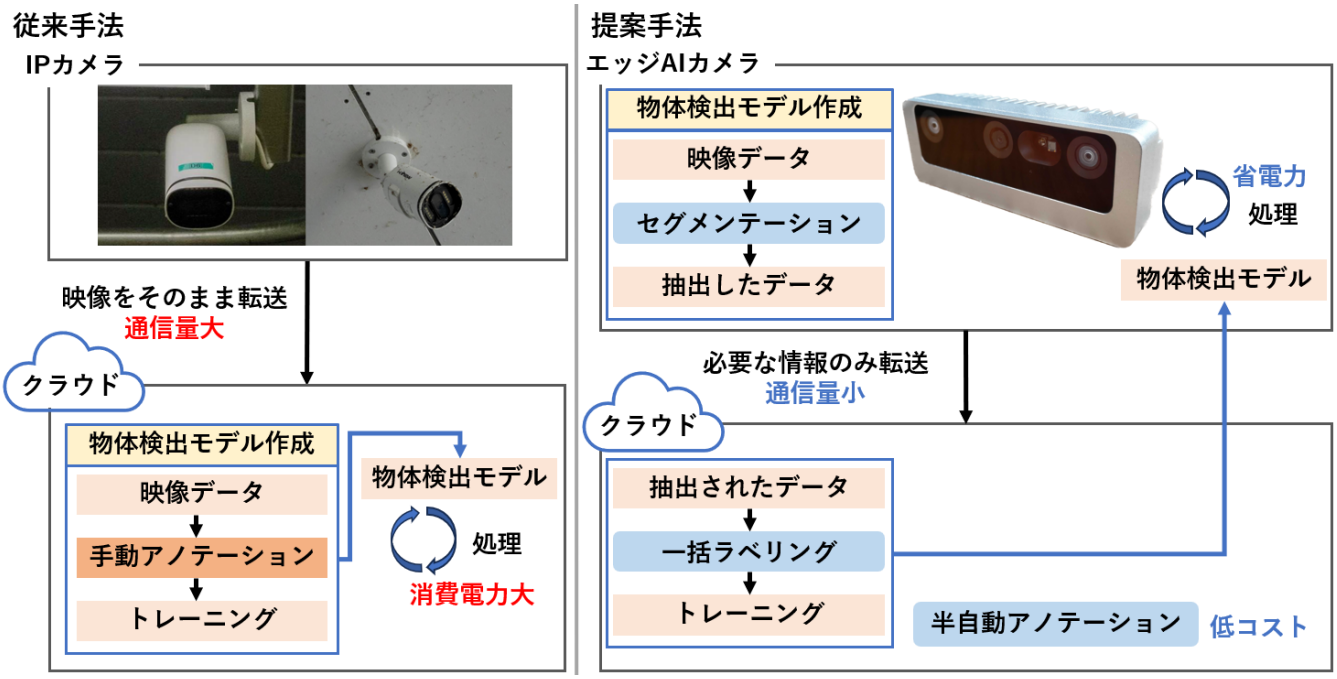

図 1: 従来手法と提案手法における物体検出モデル作成手法

データ抽出を試みている．しかし，倉庫内で撮影されたビデオ映像のような生データをすべてクラウドに送信し，処理を行うことは，多くの課題がある．例えば，エネルギー消費量の問題が挙げられる．複数のカメラからの映像を一つのクラウドに送信すると，膨大な通信量や処理電力が必要となる．60 台以上のカメラ映像を転送すると，フル HD，5fps にしても，1 日あたり約 1.2TB のデータが蓄積される．倉庫面積の拡大が続いている現在，通信量・消費電力削減は重要な課題である．通信量・消費電力削減を果たす手法として，エッジ処理が近年注目を浴びている [10]．エッジ AI カメラとは，撮影した映像をカメラ内で分析・データ抽出し，必要な情報のみを転送するデバイスである．したがって，エッジ AI カメラを用いて処理した場合，カメラで撮影した映像をクラウドに転送しての処理と比較し，通信量・消費電力を大幅に削減できる [10], [11]．一方で，計算能力の制約やエッジ AI カメラ上でのデータ抽出による情報量の低下などの制約を伴う．また，物流倉庫には多種多様なオブジェクトが存在するため，アノテーションの対象が多く，複雑な環境である．さらに，カメラの設置場所や画角により，同じオブジェクトであっても全く異なる形や大きさで撮影される．したがって，エッジ AI カメラで処理できる軽量さで，高精度な物体検出モデルが必要である．高精度な物体検出モデルを作成するには，エッジ AI カメラごとにデータセットを用意し，各カメラ画角に適した物体検出モデルを学習するのが理想的である．しかし，エッジ AI カメラごとに物体検出モデルを学習する場合，エッジ AI カメラ台数分のアノテーションが必要となり，データを集めるコストやそのデータへのアノテーションコストは膨大になる．

本研究では，学習データ生成時においてもエッジ側で処理を行い，通信量・ストレージ量を抑えつつ，セグメンテーションを自動化，複数データをまとめてラベリングしアノテーションを効率化した．図 1 に示すように，エッジ AI カメラを用いて，倉庫内の映像からオプティカルフローを用いて動的物体・静的物体の位置情報を取得し，その位置情報を基に表現学習を用いて倉庫内オブジェクトのセグメンテーションを行う．動的物体のみをセグメンテーションする場合，パレットなどの静止時間が長いオブジェクトに対して精度の良い物体検出モデルを作成できない．したがって，動的物体だけでなく静的物体に対してもセグメンテーションを行い，倉庫内のあらゆる物体に対して有用な物体検出モデルを作成する．また，エッジ AI カメラ上で自動的にオブジェクトを識別し，セグメンテーションを行うことで，大量のオブジェクトを一つ一つ手作業で矩形で囲う必要がなくなり，アノテーションのコストを大幅に削減する．さらに，エッジ AI カメラ上でセグメンテーションを行うことで，全フレームではなく，セグメンテーション結果とセグメンテーションを行ったフレームのみをサーバに転送し，通信量・ストレージ量を削減する．その後，サーバ上でクラスタリングを行い類似オブジェクトを一つのクラスタにまとめる．そして，クラスタごとに一括でラベリングをする．この過程を経て，エッジ AI カメラ画角に適した学習データセットを通信量・ストレージ量を抑えつつ，効率的に作成できる．また，作成されたデータセットを用いて，物体検出モデルを学習，エッジ AI カメラにデプロイし，消費電力を抑えた処理ができる．

本研究の貢献は以下のようにまとめられる．

- 倉庫内環境下におけるアノテーション効率化手法の流れを示した点
- アノテーションプロセスにおける通信量・ストレージ量を抑えた手法を実現した点
- エッジ AI カメラ向けの物体検出モデルを学習し，アノテーション時間を 94 %以上削減した点

2. 関連研究

2.1 汎用的なモデル

近年，Segment Anything Model(SAM)[12] などの，汎用的なモデルの研究が多く行われている [13], [14], [15]．これは学習データセットの中に存在しないオブジェクトに対しても，認識精度を発揮するゼロショット物体認識が可能なモデルである．SAM を医学の分野に発展させた研究 [13] や動画に対して SAM を適用し，トラッキング機能を付与した研究 [14] も登場している．また任意のテキストを入力として，学習データに存在しない未知のクラスに対しても，適切なセグメンテーションマスクを生成する Open-Vocabulary Object Detection の研究も行われている [15]．これらの手法は特定のオブジェクトを認識するために，アノテーションを実行する必要がない．しかし，本論文の対象としている倉庫のような非日常で多種多様なオブジェクトが混在する環境下で，どのくらいの精度を発揮するかは不明である．また，汎用モデルの構築には，膨大なデータセットが必要であり，Meta 社の Segment Anything Model(SAM)[12] には，1100 万枚以上の画像が使用されている．また，SAM は一般的にパラメータ数が多く計算コストが非常に高い．したがって，エッジ AI カメラのような計算能力の限られた環境下で，リアルタイムで実行するのは不可能であると考えられる．

2.2 アノテーションコストの削減

アノテーションのコスト削減を果たすため，アクティブラーニングに関する研究が行われている [16][17]．アクティブラーニングとは，ラベリングが未完成のデータから学習に有用なデータを何らかの方法で抽出し，それらに対して優先的に人間がラベリングを行う手法である．Lin ら [16] は，完全畳み込みネットワーク (FCN) とアクティブラーニングを組み合わせたディープアクティブラーニングフレームワークを提案した．この手法では，モデルが特定の領域に対してどれだけ自信を持っているかを示す不確実性と注釈された領域が他のトレーニングサンプルとどれだけ類似しているかを示す類似性の 2 つの指標を取得している．不確実性が高く，かつ類似性の低い領域が，注釈されるべき重要な領域として選択され，優先的にアノテーションを行う．その結果，トレーニングデータ全体の 50% のみで最先端のセグメンテーションパフォーマンスを達成した．また，Yoo ら [17] は，汎用的な新しいアクティブラーニング手法を提案した．出力の損失を推定するモデルを学習し，推定損失の大きい，つまり誤った予測をする可能性の高い領域を優先してアノテーションし，より少ないアノテーションで高い精度を達成した．しかし，これらは，従来の半分や数千件のデータセットが必要で，多種多様なオブジェクトが存在する倉庫内環境において，アノテーションコストの削減は不十分であると考えられる．

他にもアノテーションコストを減らす研究が行われている．Zhang ら [18] は，園芸において対象 (果物や品種など) が変わった場合に，新たにアノテーションをしなくても，高い認識精度を発揮できるモデル汎化法を提案した．また，Lu ら [19] は，自己教師あり学習を用いて，アノテーションを部分的に自動化し，わずかなアノテーション数で肺結節の悪精度を推定した．しかし，これらは園芸や医療など特定の環境下において実装されたものであり，倉庫内環境への適用は難しい．

また，我々はこれまでも物流倉庫のデジタルツイン構築のために，アノテーション効率化に取り組んできた．Higashiura ら [20] は，倉庫内環境において，アノテーション時間を大幅に削減可能なフレームワークを提案した．しかし，倉庫内で対象としているオブジェクトは人などの動的物体のみに限られ，パレットなどの静的物体に対しては有用ではない．さらに，本研究の目標とするエッジ AI カメラ向けの手法ではなく，通信量や消費電力は考慮していない．

2.3 エッジ AI カメラを用いた物体検出

数十、場合によっては数百台のカメラで撮影された生のビデオデータに対して，リアルタイムで物体検出および追跡アルゴリズムを実行するには、大容量メモリの GPU クラスタと膨大なストレージ容量が必要であり，さらに電力消費，実行遅延，データストレージなど，多くの課題と膨大なコストが生じる [21]．エッジコンピューティングは，それらの問題の解決手段として注目されている [22]．実際に，エッジ AI カメラを用いた物体検出や追跡アルゴリズムの研究が行われている [21], [23], [24]. Hao ら [21] は，高速道路における，車両検出とマルチカメラによるトラッキングを行った．車両検出には，YOLOv4 を使用し，オープンデータセットによって再学習したモデルを使用している．また，Mazzia ら [24] は，果樹園でリアルタイムのリンゴ検出を行った．リンゴの検出には，YOLOv3 アーキテクチャを用い，Google のデータセットを用いて再学習したモデルを使用している．これらの研究のように，エッジ AI カメラを用いた物体検出や追跡アルゴリズムの研究は行われているが，オープンソースのデータセットで十分な，日常環境のオブジェクトを対象としている．しかし，我々の対象とする倉庫内環境は多種多様なオブジェクトが混在す

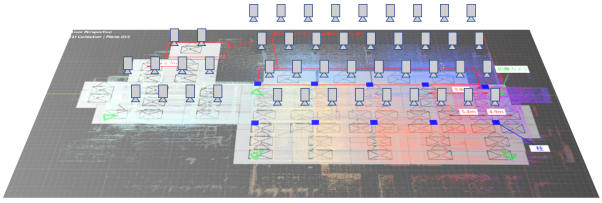

図 2: 研究対象の倉庫の大規模カメラ基盤 [20]

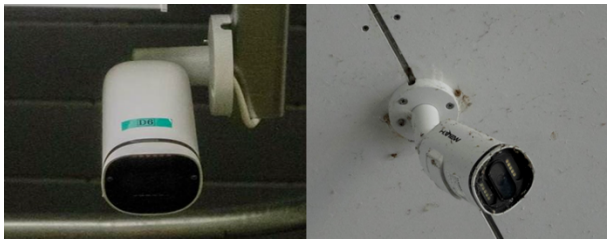

図 3: 倉庫内に設置されたカメラ [20]

る．このような複雑な環境下では，環境に合わせたデータセットを作成する必要がある．また，複数のエッジ AI カメラを用いる場合，それぞれ個別に学習データセットを作成すれば高精度が期待できるが，非常に労力やコストがかかる．そのようなエッジ AI カメラ向けの学習データセット作成手法に関する研究は著者の探す限りなかった．本研究では，エッジ AI カメラ向けの，労力やコストを削減した効率的な学習データセット作成手法を検討する．

3. 対象環境

本研究では，愛知県に位置する物流倉庫を対象として検討を行った．本倉庫は，図 2 のように，60 台以上の定点カメラで構成される大規模カメラ基盤を構築している．図 3 に示すように倉庫の天井に取り付けられ，床面を真上から撮影するものと，全体を俯瞰するように取り付けられたものがある．使用カメラは H.View 製の H V-800G2A5 であり，図 3 は倉庫内に設置されている様子である．本研究で使用する動画は，倉庫の天井に取り付けられたカメラにより取得した映像である．撮影動画の解像度は 1920 × 1080，フレームレートは 5fps の動画である．また，エッジ AI カメラとして図 4 に示す Luxonis 製の OAK-D-Pro W PoE を用いた．OAK-D-Pro W PoE は、AI とコンピュータビジョンの分野で広く利用されている．Intel の Movidius Myriad VPU を搭載しており，深度情報と色情報を取得しながら高度なニューラルネットワークを動作させることができる．そのため，物体検出、顔認識、オブジェクトトラッキングなどのタスクを実行できるエッジデバイスである．

4. 提案手法

本章では，提案手法を説明する．まず全体概要を述べ，その後各プロセスを詳細に説明する．

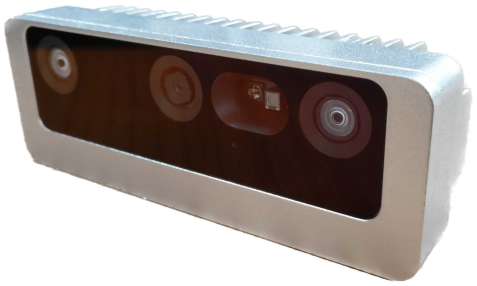

図 4: Luxonis 製の OAK-D-Pro W PoE

4.1 全体概要

本研究で提案する物体認識モデル生成のフレームワークを図 5 に示す．提案するフレームワークは，大きく分けて，「オブジェクト検出」「セグメンテーション」「クラスタリング」「ラベリング」の 4 つのプロセスに分けられる．セグメンテーションまではエッジ AI カメラ上で実行し，以降はサーバ上で実行する．また，手動で実行する部分はラベリングのみで，ラベリング以外は自動的に実行される．

最終的に，倉庫内の動画からアノテーションしたデータが取得できる．このデータを用いて，物体認識モデルを学習し，エッジ AI カメラ上で様々な分析に利用できる．

4.2 オブジェクト検出

オブジェクト検出のステップでは，入力映像に対して動的物体と静的物体の検出を行う．エッジ AI カメラで処理を行うため，この検出は計算負荷の小さい疎なオプティカルフローを用いて行う．疎なオプティカルフローの出力例を図 6 に示す．

初めに，Shi-Tomashi コーナー検出 [25] を用いて，特徴点検出を行う．その後，オプティカルフロー [26] を用いて，検出された特徴点が 2 フレーム間で 15 ピクセル以上動いた場合，その特徴点の座標を動的物体の点とする．一方，静的物体の検出では，動的物体として認識された特徴点のうち，動的物体の条件を満たさなくなったものを静的物体の点とする．これにより，動的物体から静的物体への遷移を効率的に捉えることが可能となる．また，荷置場にある特徴点のうち，動的物体ではないものを静的物体の点とする．荷置場とは，床面のうち通路ではない荷物の保管や検品を行うエリアである．これにより，荷置場にある段ボール，パレットなどを捉えられる．

4.3 セグメンテーション

セグメンテーションのステップでは，動的物体・静的物体として認識した特徴点に対し，表現学習を用いて倉庫内オブジェクトのセグメンテーションを行う．

本研究では, セグメンテーション手法として NanoSam[27] を用いた．NanoSam とは，汎用的な物体セグメンテーションモデルとして知られる Segment Anything

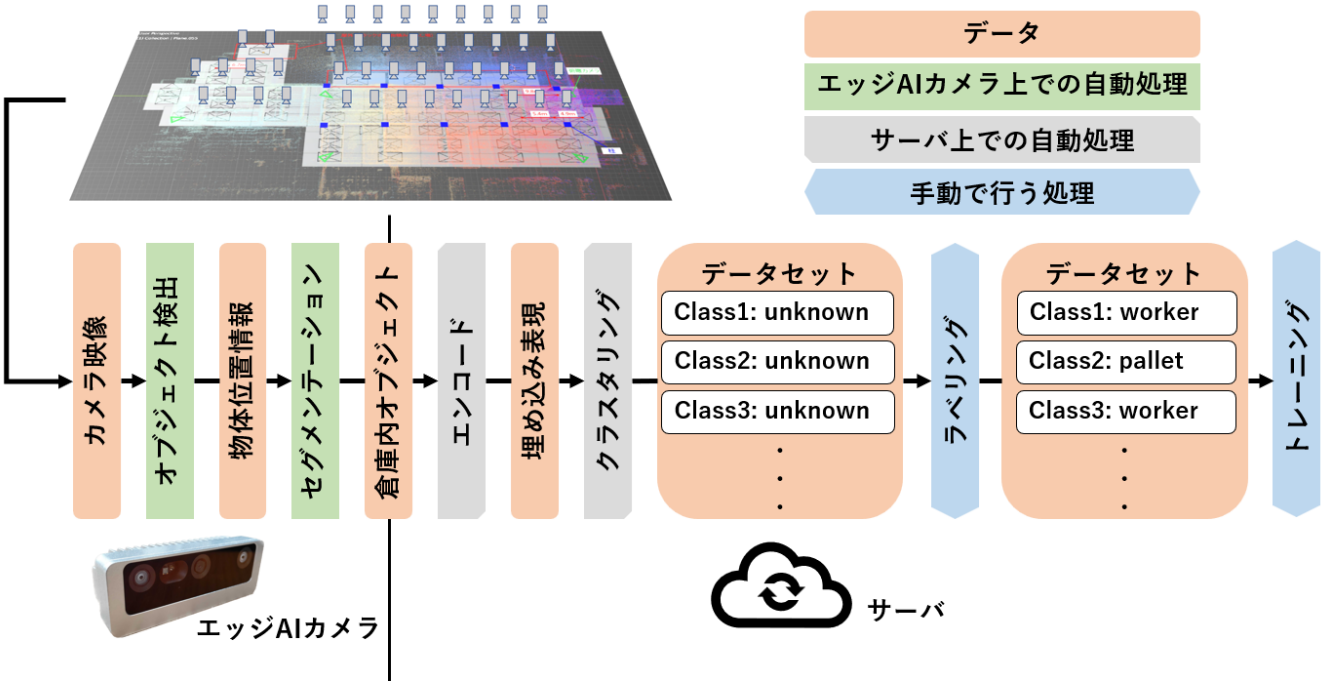

図 5: 提案手法のフレームワーク

図 6: 疎なオプティカルフローの出力例

図 7: セグメンテーションの出力例

Model(SAM)[12] を NVIDIA TensorRT でリアルタイムに実行できるように抽出したものであり，エッジ AI カメラ上で計算処理を行える．NanoSam のモデルへの入力準備としてフレームの解像度を 1024 × 1024 とし，元の画像のアスペクト比を維持するために，上下均等にパディングを追加する．動的物体・静的物体として認識した特徴点に対して NanoSam を適用し，セグメンテーションを行う．出力例を図 7 に示す．その後，セグメンテーションデータから OpenCV の findContours 関数を利用し輪郭を抽出する．

通信量・ストレージ量を減らすため，後述する制約により必要なデータのみを抽出し，サーバに転送する．まず，セグメンテーション部分の面積が画像全体の 0.5%以下のものをノイズとして除去する．また，縦横比が 4:1 より大きいものは，倉庫内オブジェクトとしては過度に細長いため除去する．1 つのオブジェクトに対し，動的物体・静的物体として認識した特徴点が複数ある場合，複数のセグメンテーション結果ができてしまう．したがって，複数のセグメンテーション結果が 30%以上重なっている場合，それらを足し合わせ 1 つにまとめる．静的物体として認識した特徴点に対し，毎フレームセグメンテーションし，データを転送してもその特徴点も静的物体も変化していない．したがって，同じデータが何度も送られてしまう．これを防ぐため，静的物体として認識した特徴点をリストとして保存し，そのリスト内容が変化したタイミングでのみセグメンテーションを行う．そのデータをサーバに転送する際には，静的物体である情報を付与し，後続の「クラスタリング」「ラベリング」ではセグメンテーションされたデータだけを使用する．最終的に，物体検出モデルを学習させる前に，適切なすべてのフレームのデータにラベリング済みの静的物体データを加える．以上のアプローチを行い，必要なデータのみを抽出してサーバに転送し，通信量・ストレージ量を削減する．

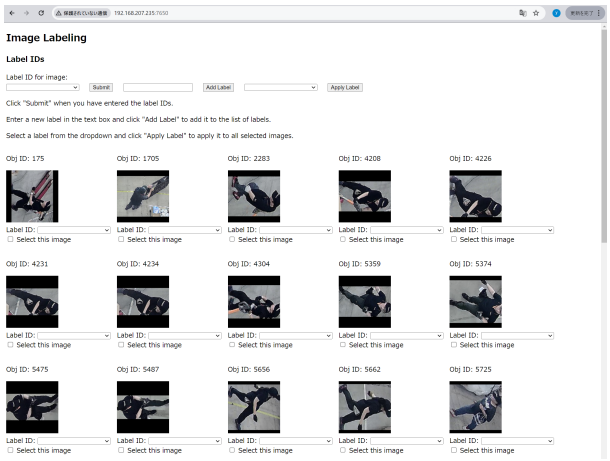

図 8: ラベリングツール

表 1: TP，FP，FN，TN の定義

	正しい	誤り
検出	TP	FP
未検出	FN	TN

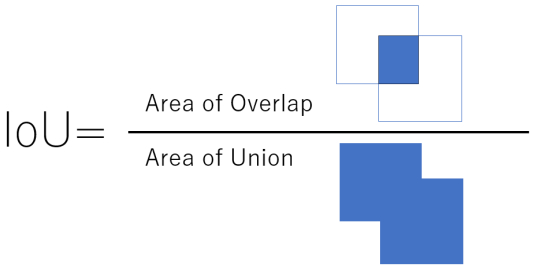

図 9: IoU(Intersection over Union)

4.4 クラスタリング

クラスタリングのステップでは，4.3 節でセグメンテーションを行った画像に対し，類似したオブジェクトを同一のクラスタにまとめる．

まず，4.3 節で得られたセグメンテーション情報をもとに，マスク部分以外を黒色で塗りつぶす．そして，マスク画像の輪郭情報をもとに，画像のトリミングを行う．その後，トリミングを行ったオブジェクト画像に対し，SimSiam[28] を用いて，形状や色などの特徴が類似したオブジェクトが距離的に近く，特徴が異なるオブジェクト同士が遠くなるようにエンコードを行い，画像から埋め込み表現を生成する．その後，クラスタリングの前に，エンコーダから得られた埋め込み表現に次元削減手法を適用する．この過程は，埋め込み表現からより本質的な特徴を抽出する他，クラスタリングに要する計算量を削減するために行う．本研究では次元削減手法として，UMAP[29] を使用する．UMAP は，データの大域的な構造と局所的な構造をバランス良く保持したまま次元削減を行う手法である．そして，次元圧縮された埋め込み表現を用いて，Kmeans 法を用いてクラスタリングを行う．Kmeans は古典的なクラスタリング手法であり，ユークリッド距離が近いオブジェクト同士をクラスタリングする．これらの過程を経て，類似したオブジェクトを同一のクラスタにまとめている．

4.5 ラベリング

ラベリングのステップでは，4.4 節で作成したクラスタごとに一括でラベルを付与する．

本研究では，独自に作成したラベリングツールを用いる (図 8)．これは，4.4 節で作成したクラスタを順に表示し，クラスタに対し，適切なラベルを付与できるツールである．このツールを用いて，クラスタごとにラベリングを行う．初めに，クラスタごとに画像を表示する．その後，画像に対し適切なラベルを付与する．各画像に対して個別にラベルを付けることもできる．基本的にはクラスタごとに一括でラベルを付与し，明らかにふさわしくない画像があれば，適したラベルを付与できる．

5. 評価実験

提案手法の有効性を検証するために評価実験を行った．まず 5.1 節で，提案手法に対するベースライン手法を定義する．次に 5.2 節で，提案手法の評価に用いる具体的な指標を定義する．その後 5.3 節で，評価のために実施した実験内容について詳細を述べ，最後に 5.4 節で，実験結果と考察を示す．

5.1 ベースライン手法の定義

提案手法との比較対象として，完全手動で行うアノテーションをベースライン手法とする．ベースライン手法では，アノテーション対象のオブジェクト一つ一つに対して，対象オブジェクトを矩形で囲うセグメンテーションと適切なラベルを付与するラベリングを全て手動で行う．この手法は，一般的に行われるアノテーションと同様のタスクである．本論文の評価実験では，提案手法とベースライン手法について，5.2 節で述べる評価指標を元に比較を行う．

5.2 評価指標の定義

本論文では提案手法の評価のため，物体検出精度，アノテーション時間の 2 つの評価指標を定義する．物体認識精度は，実行したアノテーションの品質を測る指標，アノテーション時間はアノテータがアノテーションの実行に要した時間である．本節では，各評価指標についてそれぞれ詳細に説明する．

5.2.1 物体検出精度

アノテーションされたデータは，主に物体検出モデルの学習データとして利用される．一般に，学習データの精度が高いほど，物体検出モデルの精度は高くなる．そこで本論文では，提案手法及びベースライン手法でアノテーショ

ンしたデータセットを用いて物体検出モデルを学習し，その物体検出モデルの検出精度をアノテーションの精度とする．

今回は，平均適合率 (AP:Average Precision) を物体検出精度の指標とする．AP は 0-1（百分率の場合 0-100）の間の値で評価され，AP が高いほど精度が良い．AP について解説する前に，適合率 (Precision) と再現率 (Recall)，IoU(Intersection of Union) について解説する．TP(True Positive)，FP(False Positive)，FN(False Negative)，TN(True Negative) を表 1 に示す通りに定義すると，適合率は，予測結果で検出されたものの内，正しかった割合を表し，式 (1) で示される．再現率は，すべての正解データの内，正しく検出された割合を表し，式 (2) で示される．Recall の値が r のときの Precision の値を P(r) とすると，AP は式 (3) のように表せる．また，IoU(Intersection over Union) は予測された bbox と正解データの bbox がどれくらい重なっているかを示し，図 9 のように定義される．AP50 とは，IoU が 0.5 以上の時を TP とした場合の AP の値となる．

$$\text{precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{recall} = \frac{TP}{TP + FN} \tag{2}$$

$$AP = \int_0^1 P(r)\, dr \tag{3}$$

5.2.2 アノテーション時間

アノテーションでは，より短時間でより高品質なデータセットが作成できるのが理想である．本論文では，提案手法とベースライン手法の両方でアノテータの作業時間を計測する．提案手法ではラベリングに要した時間，ベースライン手法では対象オブジェクトを矩形で囲い，ラベリングを行う時間が計測対象となる．ストップウォッチで時間を計測し，ベースライン手法，提案手法のそれぞれの所要時間を比較し評価する．

5.3 実験内容

5.3.1 使用データ

本研究では，ある 1 日に 1 台のカメラで撮影された既存の動画データを使用した．これらの動画は，30 分ごとに個別のファイルとして保存されている．1 日分の全ての動画ファイルを精査し，評価対象とした図 10 に示すパレット (荷物少ない)，パレット (積んである) の頻出頻度が高い動画ファイルを計 5 時間分使用した．

5.3.2 提案手法のデータセット作成

評価実験は図 5 の流れで行う．本来は，リアルタイムのカメラ映像を基に実施すべきであるが設置コストなどに配慮し，撮影済みの映像をカメラから直接取得しているかのように，リアルタイムで映像を処理できる機能を用いて，評価実験を行った．その機能を用いて，事前に撮影した映像を AI プロセッサに送信した．そして，その映像に対して動的物体・静的物体検出を行った．疎なオプティカルフローを用い，4.2 節の先述の条件で物体の特徴点の位置を検出した．オプティカルフローに用いた特徴点の数は最大 480 とした．次に，物体の特徴点の位置に対し，NanoSam を用いてセグメンテーションマスクを生成した．その後，生成されたマスク画像から OpenCV の findContours 関数を利用し輪郭を抽出した．その輪郭情報に基づき，オブジェクトを各インスタンスごとに分離し，SimSiam を用いて 2048 次元の画像の埋め込み表現を生成した．SimSiam のハイパーパラメータ，学習モデルは公式に公開されているものに基づいて実装した．そして，UMAP を用いて SimSiam で得られた 2048 次元の埋め込み表現を 512 次元に圧縮した．圧縮した画像の埋め込み表現を，Kmeans 法を用いてクラスタリングした．本実験では，1 つの動画ファイルから収集されたデータを 300 のクラスタに分類した．最後に，4.5 節で説明したラベリングツールを用いて，ラベリングを行った．5.2.2 項で定義したアノテーション時間を算出するため，ストップウォッチを用いてラベリングに要した時間を計測した．最終的にパレット (荷物少ない)，パレット (積んである) のラベルを付与したデータを 844,376 件を生成した．

5.3.3 ベースライン手法のデータセット作成

5.1 節で説明したベースライン手法を用いて，5.3.2 項と同様にパレット (荷物少ない)，パレット (積んである) のデータセットを作成した．5.3.1 項で説明した倉庫内映像の各フレームに対し，パレット (荷物少ない)，パレット (積んである) を手動でセグメンテーション・ラベリングし，提案手法で生成したものと同数のアノテーションを実施した．手動でのアノテーションは，著者と日常的にアノテーション業務に従事している 1 人のアノテータによって行わ

(a) パレット (荷物少ない)　**(b)** パレット (積んである)

図 10: 評価対象

表 2: 評価結果

データセット	平均適合率 (%)	アノテーション時間 (分)
提案手法	70.8	10.2
ベースライン手法	78.1	190.6

れた．また，5.2.2 項で定義したアノテーション時間を算出するため，ストップウォッチを用いてラベリングに要した時間を計測した．

5.3.4 評価用のデータセット作成

5.2.1 項で説明した物体認識精度を測定するために，5.3.3 項と同様の手法で，5.3.5 項で学習する物体検出モデルの評価用データセットを作成した．評価用データセットには，5.3.2 項，5.3.3 項に使用した映像とは別の 2 日分の映像データを使用した．それらの映像データから，パレット (荷物少ない)138 件，パレット (積んである)30 件のアノテーションを行い，評価用のデータセットとした．

5.3.5 物体検出モデルの学習と平均適合率の算出

提案手法とベースライン手法それぞれについて，5.2.1 項で説明した物体認識精度を測定するために，5.3.2 項，5.3.3 項で作成したデータセットを用いて物体検出モデルを学習した．5.3.4 項で作成した評価用データセットを用いて，平均適合率を算出し，提案手法とベースライン手法で生成したデータセットの精度を評価した．本実験では，物体検出モデルに，YOLO(You Only Look Once)[30] を用いた．学習するエポック数は 500 にし，モデルファイルは YOLOv8 の中で最も軽量である YOLOv8n を用いた．また，入力画像サイズは 640 × 640 ピクセルとした．その他のパラメータはデフォルトの値を用いた．

5.4 実験結果と考察

5.3.5 項で算出した提案手法とベースライン手法それぞれにおける平均適合率 (AP50) とデータセット作成に要したアノテーション時間を表 2 に示す．平均適合率は提案手法で 70.8，ベースライン手法で 78.1 であった．また，アノテーション時間は提案手法で 10.2 分，ベースライン手法で 190.6 分であった．

提案手法はベースライン手法と比較して，精度面でやや劣った．主な原因として，段ボールなどのオブジェクトが重なった際に，上手くセグメンテーションできず，品質の悪いアノテーションが混入されてしまうことが考えられる．例えば，図 11 のように，右上のパレットが段ボールの影響を受け，ベースライン手法と比較して，アノテーションの精度が悪くなっている．また，疎なオプティカルフローを用いた手法より，動的物体・静的物体検出時の検出漏れや，クラスタリング時にうまくクラスタリングされず適切でないラベルを付与した可能性などが考えられる．

アノテーションの効率化に関しては，アノテーション時間を 94 %以上削減でき，アノテーションコストを大幅に削減できた．これは，自動でオブジェクトにセグメンテーションを行い，クラスタごとに一括でラベルを付与できることが主な要因である．

エッジ AI カメラの計算能力の限られた環境下において物体検出・セグメンテーションを行い，必要なデータのみを転送した．今回の評価実験では，使用動画は全 89510 フレームであったが，エッジ AI カメラ上で抽出しサーバ上で使用したフレーム数は 38453 フレームで，57%以上通信量を削減した．人の動きの少ない時間帯やカメラ画角によってはさらに削減可能である．以上より，アノテーションプロセスにおいて，通信量・ストレージ量を抑えることができた．また，学習した物体検出モデルをエッジ AI カメラにデプロイし，通信量・消費電力を大幅に削減した処理も可能となる．提案手法は特定の画角に依存した手法ではない．したがって，エッジ AI カメラごとに，提案手法を実行すると同様の精度のデータセットが作成できると考えられる．

6. まとめと今後の展望

本論文では，エッジ AI 向け物体検出モデル作成におけるアノテーション効率化手法を提案した．セグメンテーションの自動化，複数データをまとめてラベリングし，倉庫内環境下におけるアノテーション効率化を行った．また，エッジ AI カメラ上で，セグメンテーションを行い，必要なデータのみをサーバに転送し，アノテーションプロセスにおける通信量・ストレージ量を抑えた手法の実現した．結果として，エッジ AI カメラ向けの物体検出モデルを学習し，精度面ではベースライン手法にやや劣るものの，アノテーション時間を 94 %以上削減し，本手法の有用性を示した．

今回は床面を真上から見下ろすカメラ映像を用いて実験を行った．全体を俯瞰して設置されたカメラ映像の場合，倉庫内オブジェクトが重なる機会の増加や，セグメンテーション対象の増加に伴う，計算量の増加など課題が多く残されている．さらに，計算コストを減らすために，疎なオプティカルフローを用いて動的物体・静的物体検出を行っ

(a) ベースライン手法　　(b) 提案手法

図 11: アノテーション品質の違い

たが，特徴点が検出されない場合はオブジェクトをセグメンテーションできないなど，精度面向上に向けたアプローチも今後の課題となる．

謝辞　本研究は国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務（JPNP23003），科研費挑戦的研究（開拓）22K18422，トラスコ中山株式会社に支援いただいている．

参考文献

[1] 経済産業省. 令和 4 年度 電子商取引に関する市場調査. 2024. https://www.meti.go.jp/press/2023/08/20230831002/20230831002-1.pdf.

[2] 国土交通省. 令和 4 年度倉庫事業経営指標. 2024. https://www.mlit.go.jp/seisakutokatsu/freight/content/001736629.pdf.

[3] Ping Li and Jiachen Zhao. Optimal path allocation of robot based on modern logistics warehouse. In *Proceedings of the 2022 5th International Conference on E-Business, Information Management and Computer Science*, pp. 378–383, 2022.

[4] Xiulian Hu and Yi-Fei Chuang. E-commerce warehouse layout optimization: systematic layout planning using a genetic algorithm. *Electronic Commerce Research*, Vol. 23, No. 1, pp. 97–114, 2023.

[5] Hui Sun and Xue Hao Gao. Research on the optimization of warehouse logistics efficiency based on order sequencing. In *Proceedings of the 2022 10th International Conference on Information Technology: IoT and Smart City*, pp. 274–279, 2022.

[6] Xiangwei Gong. Optimization algorithm of logistics warehousing and distribution path based on artificial intelligence technology. In *2022 International Symposium on Advances in Informatics, Electronics and Education (ISAIEE)*, pp. 371–375. IEEE, 2022.

[7] Ruozhen Qiu, Yue Sun, and Minghe Sun. A robust optimization approach for multi-product inventory management in a dual-channel warehouse under demand uncertainties. *Omega*, Vol. 109, p. 102591, 2022.

[8] Praveen Kumar Reddy Maddikunta, Quoc-Viet Pham, B Prabadevi, Natarajan Deepa, Kapal Dev, Thippa Reddy Gadekallu, Rukhsana Ruby, and Madhusanka Liyanage. Industry 5.0: A survey on enabling technologies and potential applications. *Journal of Industrial Information Integration*, Vol. 26, p. 100257, 2022.

[9] Barbara Rita Barricelli, Elena Casiraghi, and Daniela Fogli. A survey on digital twin: Definitions, characteristics, applications, and design implications. *IEEE Access*, Vol. 7, pp. 167653–167671, 2019.

[10] Haochen Hua, Yutong Li, Tonghe Wang, Nanqing Dong, Wei Li, and Junwei Cao. Edge computing with artificial intelligence: A machine learning perspective. *ACM Computing Surveys*, Vol. 55, No. 9, pp. 1–35, 2023.

[11] Yusuke Asai, Yuki Mori, Keisuke Higashiura, Kodai Yokoyama, Shin Katayama, Kenta Urano, Takuro Yonezawa, and Nobuo Kawaguchi. Towards a real-time and energy-efficient edge ai camera architecture in mega warehouse environment. The 3rd Real-time And intelliGent Edge computing workshop, 2024.

[12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

[13] Jun Ma, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang. Segment anything in medical images. *Nature Communications*, Vol. 15, No. 1, p. 654, 2024.

[14] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Alexander Schwing, and Joon-Young Lee. Tracking anything with decoupled video segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1316–1326, 2023.

[15] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. Vol. 36, 2024.

[16] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III 20*, pp. 399–407. Springer, 2017.

[17] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 93–102, 2019.

[18] Wenli Zhang, Kaizhen Chen, Jiaqi Wang, Yun Shi, and Wei Guo. Easy domain adaptation method for filling the species gap in deep learning-based fruit detection. *Horticulture Research*, Vol. 8, p. 119, 2021.

[19] Jiahao Lu, Chong Yin, Oswin Krause, Kenny Erleben, Michael Bachmann Nielsen, and Sune Darkner. Reducing annotation need in self-explanatory models for lung nodule diagnosis. In *International Workshop on Interpretability of Machine Intelligence in Medical Image Computing*, pp. 33–43. Springer, 2022.

[20] Keisuke Higashiura, Kodai Yokoyama, Yusuke Asai, Hironori Shimosato, Kazuma Kano, Shin Katayama, Kenta Urano, Takuro Yonezawa, and Nobuo Kawaguchi. Semi-automated framework for digitalizing multi-product warehouses with large scale camera arrays. In *2024 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 98–105. IEEE, 2024.

[21] Hao Frank Yang, Jiarui Cai, Chenxi Liu, Ruimin Ke, and Yinhai Wang. Cooperative multi-camera vehicle tracking and traffic surveillance with edge artificial intelligence and representation learning. *Transportation research part C: emerging technologies*, Vol. 148, p. 103982, 2023.

[22] Hai Lin, Sherali Zeadally, Zhihong Chen, Houda Labiod, and Lusheng Wang. A survey on computation offloading modeling for edge computing. *Journal of Network and Computer Applications*, Vol. 169, p. 102781, 2020.

[23] Muhammad Imran Zaman, Usama Ijaz Bajwa, Gulshan Saleem, and Rana Hammad Raza. A robust deep networks based multi-object multi-camera tracking system for city scale traffic. *Multimedia Tools and Applications*, Vol. 83, No. 6, pp. 17163–17181, 2024.

[24] Vittorio Mazzia, Aleem Khaliq, Francesco Salvetti, and Marcello Chiaberge. Real-time apple detection system using embedded systems with hardware accelerators: An edge ai application. *IEEE Access*, Vol. 8, pp. 9102–9114, 2020.

[25] Jianbo Shi, et al. Good features to track. In *1994 Pro-*

ceedings of IEEE conference on computer vision and pattern recognition, pp. 593–600. IEEE, 1994.

[26] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. Vol. 2, pp. 674–679, 1981.

[27] Accessed : 2024-02-22. https://github.com/NVIDIA-AI-IOT/nanosam.

[28] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15750–15758, 2021.

[29] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[30] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.