

高精細画像の高速解析を実現する階層的解析手法

A Hierarchical Analysis Method for High-Speed Analysis of High-Resolution Images

早川 翔梧[†]

Hayakawa Shogo

水谷 后宏^{‡,§}

Mizutani Kimihiro

1. はじめに

近年、デジタルカメラの性能向上に伴い、高精細画像の利用が一般化している。高精細画像は、複数の物体や細部を詳細に捉えることができるため、群衆の検出や混雑度の分析において重要な役割を果たしている。一方で画像解析には一般的に、YOLO[1]などのディープラーニング技術が広く用いられているが、これらの技術では、解析する画像を適正な入力サイズに圧縮する必要がある。高精細画像を圧縮する過程で画像の細部が失われ、特に微細な特徴や小さな物体の検出が難しくなるという問題がある。

既存研究 [2] では、高精細画像を物体が検出されなくなるまで再帰的に分割し、圧縮を行わずに解析する手法が提案されている。この手法により、小さな物体まで検出可能となり、高い精度を実現している。しかし、この手法では画像を小さな部分に分割し、それぞれを個別に解析する必要があるため、全体の処理時間が増大するという課題がある。特に、リアルタイム性が求められる状況や、大規模データセットを処理する場合には、解析速度の遅さが実用化の大きな障壁となる。

本研究では、高精細画像解析における精度と処理時間のトレードオフを改善し、リアルタイム性を求められる場面での実用性向上を図る。画像分割の効率を高めるために、画像や検出領域のサイズなどの条件を考慮した効果的な分割手法を設計することで、処理時間を短縮しつつも高い解析精度を維持することを目指す。

2. 関連研究

2.1 YOLO

YOLO (You Only Look Once) [1] は、Redmon らによって提案された物体検出手法で、画像全体を一度に処理するエンドツーエンドモデルである [1]。従来の物体検出では、領域提案と物体分類を別々に行うため、処理速度の遅延が課題となっていた。YOLO では、画像をグ

リッドに分割し、各セルで物体の存在確率、Bounding Box (物体を囲む矩形) の座標、クラス確率を同時に推定する。この手法により、リアルタイム性の高い物体検出が可能となっている。

また、検出結果に複数の Bounding Box が重なる場合、Non-Maximum Suppression (NMS) を適用し、最も信頼度の高い Bounding Box を選択して冗長な検出を削除する。この処理により、1つの物体に対して1つの Bounding Box を残し、検出精度が向上する。

YOLO は初版から世代を重ねるごとに改良が加えられ、YOLOv5[3]、YOLOv8[4]などが提案されている。これらの改良により、検出精度、処理速度、モデルの軽量化が進み、リアルタイム処理能力がさらに向上している。

本研究では、軽量で高速な YOLOv5n を用いることで、高解像度画像解析の効率化を目指している。

2.2 YOLOv5n

YOLOv5[3] は、単一のモデルではなく、計算コストと精度のトレードオフに基づいた複数のモデル群から構成されているのが特徴である。具体的には、ネットワークの深さ (depth) と幅 (width) を調整するスケーリング係数により、n (nano), s (small), m (medium), l (large), x (extra-large) といったサイズの異なるモデルが提供されている。これにより、ユーザーは用途に応じて、サーバーサイドの GPU で高精度な解析を行うための大規模モデルから、エッジデバイスで高速に動作する軽量モデルまで、最適なものを選択することが可能となっている。

本研究で採用する YOLOv5n は、YOLOv5 シリーズの中で最も軽量かつ高速なモデルとして設計されている。他のモデルと比較して、パラメータ数と FLOPs (浮動小数点演算回数) が大幅に削減されており、これにより、リアルタイムに近い速度での物体検出を実現する。

本研究が提案する階層的解析手法は、高精細画像を多数の画像パッチに分割し、それぞれに対して物体検出を繰り返し実行するアプローチを取る。このため、個々のパッチに対する検出処理は、可能な限り高速であることがシステム全体のパフォーマンスを決定づける。YOLOv5n の持つ卓越した推論速度と低い計算コストは、このような多数の推論タスクを並列処理する本研究のアーキテクチャに最適である。精度に関しても、高精細画像から切り出されたパッチ内の物体は比較的大きく写るため、軽

[†] 近畿大学大学院総合理工学研究科, Graduate School of Science and Engineering, Kindai University

[‡] 近畿大学情報学部, Faculty of Informatics (KDIX), Kindai University

[§] 近畿大学情報学研究所, Cyber Informatics Research Institute, Kindai University

量モデルでも十分な検出性能が期待できる。したがって、本研究ではリアルタイム性を最優先とし、YOLOv5n を基盤技術として採用した。

2.3 YOLOv8x

YOLOv8[4] は、Ultralytics によって開発された YOLO シリーズの最新モデルであり、その中でも YOLOv8x (extra large) は、YOLO シリーズの中で最も高精度な検出性能を実現している。このモデルは、従来の YOLOv5 からさらに改良されたバックボーンネットワークを採用しており、物体検出だけでなく、セグメンテーションやポーズ推定にも対応可能な汎用性を持つ。

YOLOv8x は、より多くのパラメータを使用することで検出精度を向上させており、特に小さな物体や複雑な背景が含まれる画像において優れたパフォーマンスを発揮する。一方で、YOLOv8x はモデル規模が大きいと、YOLOv8n などの軽量モデルと比較して推論速度が遅く、高い計算コストを伴う。このため、リアルタイム性を必要としない環境での使用が推奨される。

本研究では、提案手法の性能を評価するための比較対象として YOLOv8x を使用する。高精度を重視した場合のベースラインとして機能し、本研究が基盤とする YOLOv5n の高速性との間のトレードオフを確認する上で重要な役割を果たす。

2.4 階層的画像分割による物体検出

既存研究 [2] では、高精細画像における小さな物体の検出精度向上を目的として、階層的画像分割手法が提案されている。この手法では、画像を物体が検出されなくなるまで再帰的に分割し、各分割画像に対して YOLO による物体検出を適用する。

具体的には、スライディングウィンドウ分割（以下 SW 分割）を用いる。分割境界でのオブジェクト分断を防ぐ目的で、隣接する分割画像が縦横 50% ずつ重なり合うように設定する。このとき生じる分割画像間の重なりをオーバーラップ領域と呼ぶ。この領域の存在は多重検出の直接的な原因となるため、既存研究では検出された物体のバウンディングボックス領域を塗り潰すことで、この問題を解決する。

この既存手法により、高精細画像においても小さな物体まで高精度に検出することが可能となった。しかし、画像を細かく分割するほど処理すべき画像パッチ数が増加し、全体の処理時間が大幅に増大するという課題が明らかになった。特に、リアルタイム性が求められるアプリケーションでは、この処理時間の増大が実用化の大きな障壁となっている。

3. 研究内容

本研究は、高精細画像解析における検出精度と処理時間のトレードオフを改善し、リアルタイム性が求められる場面での実用性を向上させることを目的とする。特に既存研究 [2] で高精度な検出を実現しつつも、画像分割に伴う全体の処理時間増大という課題を抱えている点を解決することを目指すものである。

この課題に対し、本研究では、既存研究 [2] で用いられていた SW 分割を改良し、より効果的な階層的画像分割手法を提案する。解析には、検出精度と処理速度のバランスに優れた YOLOv5n を使用し、分割された画像を個別に解析するアプローチを採用する。

本研究のタスクは、図 1 に示すように、高精細画像を効率的に分割し、群衆の人物検出を高速かつ高精度に実行することである。

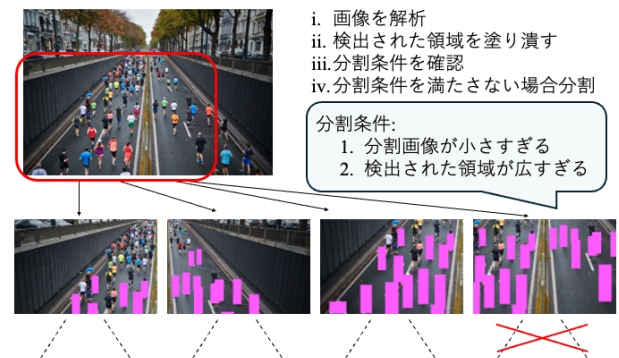


図 1: 提案手法：概要図

3.1 SW 分割

本研究では、既存研究 [2] で用いられていた SW 分割を基盤として使用する。画像を単純に等分割すると、分割した境界でオブジェクトが分断される恐れがある。それを防ぐために、SW 分割では、縦横に重なりを持って分割する。これにより、分割の境界におけるオブジェクトの分断を防ぎ、正しく検出することが可能となる。しかし、この方法では、オーバーラップ領域において同じオブジェクトを多重検出する恐れがある。この問題を検出されたオブジェクトのバウンディングボックスに基づき領域を塗り潰すことで解決された。

3.2 効果的な階層的分割手法

本研究では、効果的な階層的分割のために、以下の条件を設けて分割の効率を向上させる：

- **分割画像サイズによる停止条件:** 分割された画像が、YOLOv5n がデフォルトで使用するサイズである 640×640 ピクセルより小さい場合
- **検出領域密度による停止条件:** 検出領域が分割された画像の 80% 以上を占める場合

3.2.1 適応的 SW 分割アルゴリズム

従来の SW 分割が 50% などの固定的な比率で重なりを設定するのに対し、本研究では適応的 SW 分割を提案する。本手法は、ウィンドウサイズに対するオーバーラップ領域の割合であるオーバーラップ比率を、入力画像のサイズに応じて動的に調整する。これにより、分割後の各画像パッチが、解析に最低限必要なサイズ（本研究では 640 ピクセル）を下回ることを防ぐ。

変数の定義：SW 分割における各変数を以下のように定義する。

- W ：画像の横幅（ピクセル数）
- H ：画像の縦幅（ピクセル数）
- s ：YOLOv5n の入力サイズ（640 ピクセル）
- o_x ：横方向のオーバーラップ比率
- o_y ：縦方向のオーバーラップ比率

分割条件の判定：画像サイズに基づいて分割方法を決定する。

- $W > 1280, H > 1280$ ：横方向および縦方向で 3 分割を行う
- $1280 > W > s, 1280 > H > s$ ：横方向および縦方向で 2 分割を行う
- $W \leq s$ または $H \leq s$ ：該当方向は分割を行わない

オーバーラップ比率の計算：分割された各部分画像が最小サイズ 640 ピクセルを下回らないよう、以下の式でオーバーラップ比率を計算する。

$$o_x = 1 - \frac{s}{W}, \quad o_y = 1 - \frac{s}{H} \quad (1)$$

この適応的アルゴリズムにより、画像サイズに応じて最適な分割数とオーバーラップ比率が決定され、YOLOv5n に適したサイズの画像パッチを確実に生成できる。

これらの条件により、無駄な分割を避け、処理効率を大幅に改善する。これらの条件を満たさない限り、画像は階層的に分割し続け、細部まで解析を行う。

3.3 高速化手法

本研究では、分割した画像の効率的な処理のため、3 段階バッチ処理による高速化手法を提案する。

3.3.1 重なり回避のための段階的バッチ処理

SW 分割で生成された 9 つの画像パッチに対し、バッチ処理による高速化と、オーバーラップ領域における多重検出の回避を両立させるため、本研究ではパッチ間の隣接関係に基づいた 3 段階のバッチ処理手法を提案する。本手法の処理フローを以下に示す。

第 1 段階（四隅領域の処理） 最初に、互いに隣接しない四隅の画像パッチ（0, 2, 6, 8）を対象とする。これらのパッチはオーバーラップを持たないため、多重検出の懸念なく、4 枚を一つのバッチとして同時に YOLO による推論処理を実行する。検出された物体のバウンディングボックスは、即座に全体共有マスクに描画（塗りつぶし）される。

第 2 段階（辺中央領域の処理） 次に、辺の中央に位置する画像パッチ（1, 3, 5, 7）を処理する。この段階では、第 1 段階の処理で共有マスクが更新されているため、隣接領域で検出済みの物体はすでにマスクされている。各パッチに対し、マスクされた領域の割合が閾値（80%）を超えているかの事前チェックを行い、条件を満たさない（未処理領域が多い）パッチのみを収集してバッチ処理を適用する。

第 3 段階（中央領域の処理） 最後に、中央の画像パッチ（4）を処理する。このパッチは周囲の全てのパッチと隣接しており、第 1・第 2 段階の処理が完了した時点では、オーバーラップ領域の大部分が処理済みとなる。同様に事前チェックを行い、条件を満たさなければ単独で推論処理を実行する。

提案する段階的処理手法は、SW 分割の課題であったオーバーラップ領域の多重検出問題を、後処理（NMS など）ではなく、推論プロセス自体に組み込むことで効率的に回避する。特に処理が進むにつれて共有マスクの累積効果により後続の処理対象が削減され、システム全体の計算効率を向上させる効果がある。

4. 実験

本章では、提案手法の性能評価を行うために実施した実験について述べる。

実験環境：本実験は、Intel Core i7-13700K CPU、128GB のメモリ、そして 2 基の NVIDIA GeForce RTX 4070 GPU を搭載した計算機上で行った。

本研究では、4K 解像度（3840 × 2160 ピクセル以上）で 50 人から 450 人が写る画像を 15 枚用いて実験を行った。本実験では、以下の手法を比較対象とする：

- **YOLOv5n（機械的 SW 分割）：**画像を機械的に 640 × 640 ピクセルで SW 分割し、YOLOv5n で解析

- YOLOv8x: 高精度ベースラインとしての単純な画像の解析
- YOLOv5n (提案手法) : 効果的な階層的分割条件を適用した YOLOv5n 解析

実験の結果を表 1, 図 2, 図 3 に示す.

手法	解析精度 (%)	解析時間 (s)
機械的な SW 分割	150.34	0.717
YOLOv8x	23.17	0.153
YOLOv5n (提案手法)	92.95	0.269

表 1: 実験結果

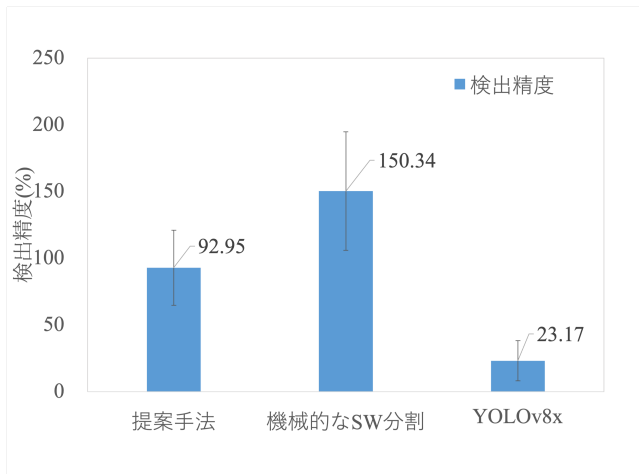


図 2: 実験結果：平均解析精度

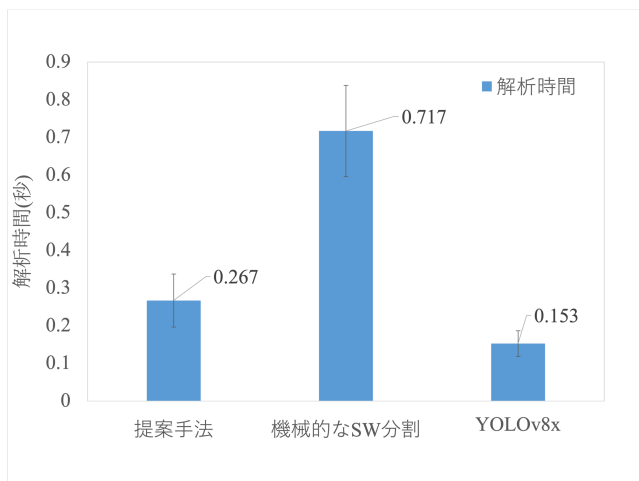


図 3: 実験結果：平均解析時間

4.1 機械的 SW 分割の課題

機械的な SW 分割では、精度が 150.34%と 100%を大きく超えている。これは、オーバーラップ領域での多重検出や誤検出により、過剰な検出が発生していることを示唆している。また、画像を機械的かつ網羅的に分割するため、処理時間も 0.717 秒と最も長く、非効率的である。

4.2 提案手法の優位性

提案手法は、精度 (92.95%) と処理時間 (0.269 秒) のバランスに最も優れている。機械的 SW 分割と比較して、不要な分割を回避する条件を設けたことで、処理時間を 0.717 秒から 0.269 秒へと約 62.5%削減した。同時に、過剰検出を抑制し、より現実に即した高い精度を達成した。

YOLOv8x と比較すると、処理時間の増加を約 0.1 秒に抑えつつ、精度を 23.17%から 92.95%へと約 4 倍に向上させた。これは、画像の詳細情報を失うことなく、効率的に解析できていることを示している。

以上のことから、提案手法は「高精度」と「高速処理」というトレードオフの関係において、既存手法の課題を解決し、実用的な解決策であることが実験的に示された。

5. 結論・今後の課題

表 1 より、提案手法は機械的 SW 分割手法の過剰検出問題を解決し、現実的な精度 (92.95%) を実現しつつ、解析時間を約 62.5%削減 (0.717 秒→0.269 秒) することができた。また、YOLOv8x と比較して、わずかな処理時間の増加 (約 0.1 秒) で精度を約 4 倍に向上 (23.17% → 92.95%) させることができた。

本研究で提案した効果的な階層的分割手法により、高精細画像解析における精度と処理時間のトレードオフを大幅に改善することができた。特に、適応的 SW 分割アルゴリズムによる不要な分割の回避と、3 段階バッチ処理による多重検出の防止により、既存手法では実現できない高精度かつ高速な解析を実現した。

今後の課題は、日本防犯設備協会 [5] が 5fps 以上と規定している処理速度基準を、精度を維持しつつ上回ることである。また、より多様な解像度やシーンでの実験を行い、提案手法の汎用性を検証する必要がある。さらに、分割条件のパラメータを動的に調整する手法の開発により、さらなる性能向上が期待される。

謝辞

本研究は JSPS 科研費 JP23K16876 の助成を受けたものである。

参考文献

- [1] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.
- [2] Shogo Hayakawa and Kimihiro Mizutani. An effective hierarchical object detection scheme for high resolution image analysis. In *Proceedings of the IEEE International Conference on Emerging Technologies for Communications (ICETC)*, pp. P2–33, 2024.
- [3] Glenn Jocher, et al. YOLOv5 by ultralytics. <https://github.com/ultralytics/yolov5>, 2020.
- [4] Ultralytics. YOLOv8 by ultralytics. <https://github.com/ultralytics/ultralytics>, 2023. Accessed: 2025-07-23.
- [5] 日本防犯設備協会. 防犯カメラシステム ガイドブック. <https://www.ssa-j.or.jp/pubdoc/pdf/guidebook/448.pdf>, 2022. Accessed: 2025-07-23.