

B16BL6: A public time-lapse microscopy dataset and data-centric annotation pipeline for cell and division detection

Mokhaled N.A. Al-Hamadani ^{a,b,c} ^{*}, Péter Szilágyi ^a, Gabor Szeman-Nagy ^d, Andras Hajdu ^a, Balazs Harangi ^a

^a Department of Data Science and Visualization, Faculty of Informatics, University of Debrecen, H-4032 Debrecen, Hungary

^b Doctoral School of Informatics, University of Debrecen, H-4032 Debrecen, Hungary

^c Department of Network and Computer Software Technologies, Polytechnic College – Hawija, Northern Technical University, 36001 Kirkuk, Iraq

^d Department of Microbial Biotechnology and Cell Biology, University of Debrecen, Life Sciences Building 1.102, 1 Egyetem Square, Debrecen, 4010, Hungary

ARTICLE INFO

Dataset link: <https://doi.org/10.17605/OSF.IO/5J276>

Keywords:

Microscopy imaging
Cell detection
Deep learning
YOLO
R-CNN
RT-DETR
Computer vision
Biomedical imaging

ABSTRACT

Automated cell detection and identification of mitotic division events in phase-contrast microscopy are essential for quantifying proliferation, phenotypic response, and population dynamics, yet practical deployment remains limited by annotation cost and unclear architecture selection under real-time constraints. This paper releases B16BL6, a public phase-contrast time-lapse microscopy dataset comprising 1800 frames (1600×1200) with 93,637 bounding-box annotations for two classes (Cell and Division). To enable scalable labeling at this density, we present a three-stage annotation pipeline that combines (i) expert manual labeling of a representative subset, (ii) YOLO-based pseudo-label generation with expert refinement, and (iii) temporally consistent label propagation using a detector–tracker hybrid with an enhanced DeepSORT variant.

Using this dataset, we benchmark seven contemporary detectors spanning one-stage CNNs (YOLOv8x, YOLOv11-L, SSD300/512), two-stage models (Faster R-CNN, Mask R-CNN, Libra R-CNN), and a real-time transformer (RT-DETR-L). Evaluation is multi-dimensional, covering detection accuracy (mAP@0.50 and mAP@0.50:0.95), class-specific AP, inference speed, parameter/GFLOP complexity, and imbalance sensitivity via the Cell-Division AP gap, together with a supplementary heuristic efficiency score for deployment-oriented comparison. Results show that YOLOv8x achieves the best accuracy (mAP@0.50 = 0.837), YOLOv11-L provides the best accuracy–speed balance (20.15 FPS, mAP@0.50 = 0.811), and two-stage detectors exhibit more stable class-level performance under imbalance, while SSD offers maximal throughput at reduced precision. Rather than proposing a new detection architecture, this work releases a densely annotated microscopy dataset, a scalable annotation workflow, and a unified benchmark of modern detectors, providing practical guidance for scalable microscopy pipelines and downstream tracking and division analysis.

1. Introduction

Accurate examination of cell presence, movement and division is a fundamental element of numerous biomedical applications, including cancer research, drug discovery and other cell population studies [1]. Quantitative analysis of these processes enables a deeper understanding of cell dynamics and behavioral patterns. Advances in microscopy have made it possible to capture large volumes of high-resolution visual data, increasing the demand for reliable automated methods to identify cells and biological events.

Manual annotation of cellular sequences is highly time-consuming and the involvement of multiple experts for validation makes the process difficult to scale [2]. These limitations obstruct long-term and

high-throughput studies, and have increased interest in automated and scalable cell analysis solutions. Cellular appearance varies substantially in shape, size and texture, even within the same sample, while cells frequently overlap and appear in dense clusters [3]. In addition, microscope imaging modalities have diverse visual characteristics further complicating the development of robust detection algorithms. Among cellular events, cell division is a particularly challenging task, as cells undergo complex morphological changes during division. The reliable detection of such transient events requires advanced algorithms that can later be implemented into automated cell detection pipelines.

Recent progress in machine learning, especially deep learning-based object detection, has significantly improved automated image analysis.

* Corresponding author.

E-mail address: alhamadani.mokhaled@inf.unideb.hu (M.N.A. Al-Hamadani).

<https://doi.org/10.1016/j.array.2026.100946>

Received 17 February 2026; Received in revised form 22 May 2026; Accepted 25 May 2026

Available online 29 May 2026

2590-0056/© 2026 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

These methods enable the simultaneous localization and classification of objects or events while providing confidence scores for each prediction. As a result, expert workload is reduced and reproducibility is increased, therefore analyzing large amounts of data becomes feasible. Detection of cells can be extended to more complex tasks such as movement tracking, counting or lineage tracking in sequences of samples [1].

Modern object detection architectures can be categorized into single-stage, two-stage, and more recently transformer-based models. While these approaches have been extensively benchmarked on general-purpose datasets, their comparative performance remains insufficiently explored on microscopic cell imagery. Existing studies typically focus on individual architectures or limited families of architectures, which makes it difficult to draw a real conclusion on the advantages and disadvantages of individual architectures in a specialized field such as microscopic cell studies. Furthermore, division detection remains underrepresented compared to general cell detection in comparative studies.

To address these gaps, this work presents a comprehensive comparative analysis of state-of-the-art object detection architectures for automated cell detection and division recognition in microscopy images. We evaluate representative detectors from multiple families—including YOLO variants, SSD, Faster R-CNN and its extensions, and RT-DETR—under consistent dataset splits and COCO-style evaluation, and we report not only accuracy but also computational and practical deployment characteristics.

While the present study builds upon components introduced in our previously published microscopy detection-and-tracking framework [4], its scope and contribution are distinct. The earlier work focused on a single integrated pipeline that combined a fine-tuned YOLOv8x detector with an enhanced DeepSORT tracker in order to improve recall, continuity, and missed-detection recovery in microscopy image sequences. In contrast, the present manuscript does not primarily aim to propose or evaluate a new detector-tracker integration. Instead, it makes a broader data-centric contribution through the public release of the B16BL6 dataset, the formalization of a three-stage annotation methodology for scalable dense label construction, and a unified benchmark of seven modern detection architectures under consistent evaluation settings. Thus, the novelty of the present work lies in the dataset, annotation framework, and comparative benchmark resource, rather than in the inherited tracking components themselves.

Accordingly, the present study makes a data-centric rather than architectural contribution, centered on the public release of a dense microscopy dataset, a scalable annotation methodology, and a unified benchmark for modern detection architectures under consistent evaluation conditions.

Contributions. The main contributions of this work are:

1. **Public dataset release:** We release B16BL6, a densely annotated phase-contrast time-lapse microscopy dataset of 1800 frames (1600×1200) with 93,637 bounding-box annotations for Cell and Division events.
2. **Scalable annotation pipeline:** We formalize and apply a three-stage annotation workflow (manual annotation $\rightarrow$ pseudo-labeling with expert correction $\rightarrow$ temporally consistent propagation) for scalable construction of dense microscopy annotations, extending components from our previously published microscopy detection-and-tracking pipeline [4] into a dataset-construction and benchmarking framework..
3. **Unified multi-paradigm benchmark:** We provide a controlled benchmark of seven modern object detectors spanning one-stage, two-stage, and transformer-based paradigms under standardized training and COCO-style evaluation settings.
4. **Multi-axis evaluation beyond mAP:** We quantify deployment trade-offs using accuracy, inference speed, GFLOPs/parameters, fine-tuning sensitivity, and class-imbalance robustness via the Cell-Division AP gap, together with a supplementary heuristic efficiency score.

5. **Actionable selection guidance:** We distill the empirical findings into practical guidelines for selecting models for real-time versus accuracy-critical microscopy pipelines and for division detection under strong class imbalance.

The remainder of our work is organized as follows: Section 2 reviews the related work on deep learning-based cell detection in microscopy imaging. Section 3 describes the dataset, annotation methods and state-of-the-art architectures used for the comparative analysis. Section 4 presents the experimental results and evaluates performance. Section 5 discusses the results, emphasizing their strengths and limitations, and finally, Section 6 concludes the paper and outlines future research directions.

2. Related work

2.1. Overview

Recent advances in deep learning-based object detection have significantly improved the automated analysis of microscopy images. Traditional image processing methods typically rely on handcrafted features and problem-specific heuristics to detect cells and recognize division events. In contrast, modern convolutional and attention-based architectures enable the detection of complex cellular morphologies through data-driven feature learning, which is difficult to achieve with traditional approaches.

Existing research spans two-stage and single-stage detectors, and more recently transformer-based methods, with emphasis placed not only on detection accuracy, but model complexity and inference speed. This section reviews prior work across the three major detector categories and highlights existing research gaps relevant to cell and division detection.

2.2. Two-stage detection in microscopy imaging

Two-stage object detectors, most notably Faster R-CNN [5], have been widely used in medical and microscopic image analysis due to their ideal balance between detection accuracy and model flexibility. Several studies demonstrate the applicability of Faster R-CNN to blood cell detection and classification tasks with various backbones and datasets [6–8]. These works reveal substantially improved detection accuracy compared to classical image processing approaches, particularly for complex cellular morphologies. However, they also report challenges in terms of reduced performance in overlapping cells and classes that are visually difficult to distinguish.

The Mask R-CNN architecture [9] extends Faster R-CNN by generating a pixel-level instance segmentation mask, which allows for the precise contour of objects, such as cells. Although the additional segmentation branch is advantageous for applications requiring pixel-level accuracy, it increases computational complexity and potentially limits practicality for real-time pipelines.

To address imbalances during the training process in object detection networks, Libra R-CNN [10] balances training on the sample, feature and objective levels. Even though IoU-balanced sampling, balanced feature pyramid and balanced L1 loss strategies have shown consistent improvements on large-scale benchmarks, their advantages in microscope imaging are inconclusive. Comparative studies on limited biological data report mixed results, suggesting that the effectiveness of balanced training may strongly depend on annotation consistency or dataset scale [11].

2.3. Single-stage detectors: SSD and YOLO

Single-stage object detectors perform localization and classification in a single pass. An early but defining architecture was the Single Shot MultiBox Detector (SSD), which used multiscale feature maps and predefined default boxes. SSD has been applied to various medical image processing tasks, including cervical cell detection and early cancerous lesion identification [12,13]. These studies demonstrate high sensitivity, which is crucial in the field of medical diagnosis. However, they highlight the limitations of SSD in handling subtle morphological differences and overlapping objects.

The YOLO family of architectures treat object detection as a regression problem, which enables them to be suitable for real-time applications due to their high inference speed and simplified pipeline [14]. Recent studies report strong performance across YOLO versions in the field of cell detection [15], but highlight that they tend to be more sensitive to noise compared to two-stage detectors [16]. Large-scale benchmarking of YOLOv11 variants for peripheral blood cell detection further confirms that single-stage architectures are effective in the medical field [17]. Medium sized variants provide the best balance between speed and accuracy, as smaller variants often confuse morphologically similar cell types, and large variants have higher computational complexity resulting in a lower inference speed.

2.4. Transformer-based object detectors

While convolution-based architectures are able to perceive global information on images through numerous layers, transformer-based detectors allow regions of the image to attend to other regions in a single self-attention layer [18]. Thanks to the attention mechanisms, DETR-based detectors have shown promise in handling densely clustered objects and complex backgrounds. Early applications in biological imaging, such as Cell-DETR, despite having a simpler architecture, demonstrated comparable detection accuracy to traditional two-stage methods, [19].

More recently, RT-DETR has emerged as a real-time transformer-based object detector, that achieved better performance than several YOLO architectures on large-scale benchmarks [20]. Preliminary evaluations on multiple medical datasets suggest improved accuracy on small and clustered objects compared to earlier DETR variants and single-stage detectors, such as YOLO or SSD [21]. Even though RT-DETR shows potential, systematic evaluation of its performance on cell detection and division recognition remains limited.

2.5. Summary and research gap

Prior research shows that deep learning-based object detectors have substantially improved automated cell detection in microscopy imaging. Two-stage detectors, such as Faster R-CNN and Mask R-CNN, provide high localization accuracy and handle complex cellular morphologies well, but their high computational complexity and slower inference limit scalability. Balanced two-stage approaches like Libra R-CNN aim to address training imbalances, yet their advantages over standard two-stage detectors are inconsistent on small, domain-specific biological datasets.

Single-stage detectors, including SSD and the YOLO family, are widely used due to their high inference speed and accuracy. Recent YOLO versions demonstrate strong sensitivity and robustness to data imbalance. However, these models often struggle with subtle morphological differences and precise size estimation, which are critical in cellular images.

Transformer-based detectors introduce global attention mechanisms that are well suited for clustered objects and complex backgrounds. Studies on DETR variants and RT-DETR suggest improved performance on small-object detection while approaching real-time speeds. Despite this potential, their evaluation on microscopy data remains limited.

Overall, existing studies typically focus on individual, or a family of architectures instead of systematic comparisons. Evaluations across two-stage, single-stage, and transformer-based detectors, including modern real-time state-of-the-art models such as YOLOv11 and RT-DETR, are largely missing. Moreover, the detection of cell division events remains underexplored compared to general cell detection. This work addresses these gaps by providing a unified, quantitative comparison of state-of-the-art detection architectures for cell detection and division recognition in microscopy imaging.

3. Materials and methods

3.1. Dataset description and preparation

In this study, we used a B16BL6 mouse melanoma microscopy dataset, a widely studied model in oncology and cell biology [22]. The annotated dataset and labels released with this work are publicly available on the Open Science Framework (OSF) under the Creative Commons Attribution 4.0 International (CC BY 4.0) License [23]. This dataset was provided by the Department of Microbial Biotechnology and Cell Biology at the University of Debrecen. It consists of 1800 frames, captured from time-lapse microscopy video with a spatial resolution of 1600×1200 pixels.

Many cells in different phases of the cell cycle can be found in each frame; representative examples of the B16BL6 microscopy data are provided in our previously published microscopy detection-and-tracking study [4]. There are two classes of ground truth annotations: the Cell class for the cells that are actively moving in the video (the Cells are classified as non-dividing cells) and the Division class for dividing cells (cells just prior to separating into two daughter cells who have increased in surface area) [24].

The annotation part of this dataset has been formalized in this manuscript as a three-stage methodology for scalable dataset construction. This workflow builds upon components established in our previously published microscopy detection-and-tracking pipeline [4], while extending them into a broader data-centric framework for public dataset release and comparative benchmarking. In addition, the design of the annotation process is conceptually informed by data-focused approaches used in large-scale visual cognition tasks [25], which are explained later in this section.

3.1.1. Three-stage annotation methodology

For time-lapse microscopy datasets, annotation is typically performed manually, which requires substantial time and human effort. To reduce this burden while maintaining label quality, we designed an annotation pipeline for scalable ground-truth generation across the full sequence. The pipeline combines expert review, automatically generated pseudo-labels, and a tracking-based refinement stage to improve annotation quality while reducing the amount of manual supervision required, as illustrated in Fig. 1.

Stage 1: Manual Annotation. We have completed the manual annotation of a representative sample of the B16BL6 database which represents our first step in developing our annotation methodology. This included training on 204 frames from the data set and validation on 138 frames resulting in a total of 342 frames for this first phase of study (19% of the total data set). The expert annotators worked alongside domain experts to ensure accurate bounding box measurements around relevant cell locations as well as correct class labels assigned to each annotation. The annotation guidelines were specifically to provide consistent and uniform standards for delineating the boundaries of cells during the time of cell division when the cells may change in shape or size. The first round of annotation required approximately 120 h of effort by expert annotators and created the core corpus of accurate labels for training purposes on the initial object detection model as well as establishing a quality benchmark for the second round of annotation. The manual annotations were performed by a trained

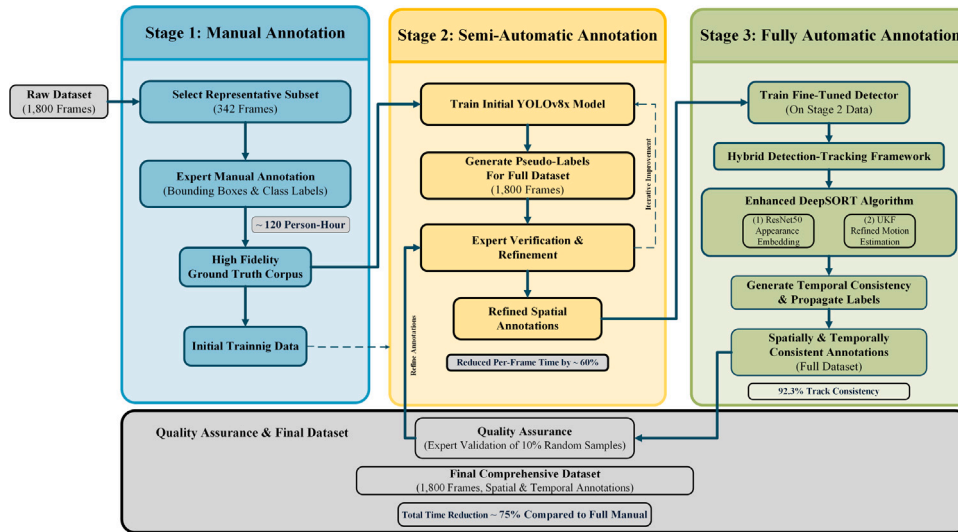


Fig. 1. Overview of the three-stage annotation methodology used to construct the B16BL6 dataset. Stage 1 consists of expert manual annotation of a representative subset; Stage 2 applies YOLOv8x-based pseudo-label generation followed by expert correction; and Stage 3 performs temporally consistent propagation using the detector-tracker framework to refine and extend annotations across the full sequence.

expert and validated by two domain specialists from the Department of Microbial Biotechnology, including a biology expert who is also a co-author of the present manuscript. Discrepancies were resolved by consensus to produce the final label set. A formal statistic of inter-annotator agreement was not calculated in the present study, as the annotation process was based on expert-guided consensus refinement rather than independent duplicate annotation.

Stage 2: Semi-Automatic Annotation. As mentioned in the first stage, where 19% of the dataset was annotated manually, this stage aimed to extend the annotation to the entire dataset while maintaining quality. We employed the YOLOv8x model trained on the manually labeled data and then utilized it to generate pseudo-labels for the entire 1800-frame dataset. Consistent with our previously published microscopy detection-and-tracking framework, this stage achieved a Modified-Recall score of 85.95 on the validation subset, providing workflow-level evidence that the pseudo-label generation process was sufficiently reliable for expert-guided refinement. This pseudo-label validation should therefore be interpreted as a practical quality-control step within the annotation workflow rather than as an exhaustive standalone validation of pseudo-label reliability. The pseudo-labels generated by the model underwent meticulous verification and rectification by expert reviewers, with particular emphasis on correcting classification inaccuracies, rectifying overlooked detections, and enhancing the precision of bounding box delineations. This semi-automatic phase led to a substantial acceleration of the labeling process, reducing per-frame annotation time by nearly 60% compared to full manual annotation, while preserving annotation accuracy through expert-guided corrections. This strategy demonstrated notable effectiveness, particularly in frames containing moderately dense cells and low-occlusion frames, where the model provided reliable predictions with high confidence and accuracy.

Stage 3: Fully Automatic Annotation. The final stage of our integrated pipeline included both a fine-tuned version of the YOLOv8x model from Stage 2 and a modified version of the DeepSORT tracking algorithm, which included two new enhancements to improve tracking results: better estimates of motion and better estimates of appearance. The enhancements in appearance were accomplished by using a multi-scale feature extraction process using a modified ResNet50 model pre-trained using cell microscopy images in order to extract domain-specific visual features. For modeling motion, we adopted the Unscented Kalman Filter (UKF) within the tracking module, following our previously published microscopy detection-tracking framework [4]. In

the present study, UKF was used as part of the annotation-propagation design to support irregular cell motion; however, a dedicated quantitative comparison against a standard linear Kalman Filter was beyond the scope of this work. During dense interactions, including cases in which two cells physically touch or partially overlap, identity continuity is maintained through joint motion-plus-appearance association: the UKF provides short-term trajectory prediction for existing tracks, while the appearance encoder supports re-identification when detections become ambiguous and subsequently separate again.

Therefore, this integrated detection-tracking approach enabled temporal consistency of label propagation across successive frames, thereby mitigating false positives, restoring missed detections by leveraging trajectory continuity, and reducing identity switch problems frequently encountered in dense cell populations. The temporal coherence introduced through the tracking component not only enhanced spatial annotation reliability but also allowed downstream analyses such as lifecycle modeling, cell lineage reconstruction, and quantitative analysis of population-level dynamics. Track consistency was evaluated by computing the fraction of tracks extending over at least 10 consecutive frames without identity switches, yielding a 92.3% consistency score (modified recall) on manually validated sequences.

The three-stage framework proposed in this paper formalizes a methodology that builds upon our previously published integrated pipeline and utilizes both the expertise of human annotators (manual) as well as the efficiencies gained by using a semi-automated and fully automated method to create an annotated dataset containing both spatial and temporal annotations. To provide an additional workflow-level quality check for the final automated stage, a random sample corresponding to 10% of the automatically propagated object annotations was manually reviewed by domain experts. The sampled annotations were inspected against their corresponding microscopy frames, and three categories of discrepancy were recorded: false positives, false negatives, and misclassifications. The reported error rate of less than 3% was computed as the proportion of reviewed sampled annotations associated with these discrepancy categories relative to the total number of sampled annotations inspected. This additional manual review supported the overall reliability of the annotation methodology. Overall, this approach achieved an approximate 75% decrease in total annotation time compared to fully manual annotation while preserving quality levels suitable for both training and assessing state-of-the-art detection models.

3.1.2. Dataset partitioning and statistics

Once finished with the annotation process, the dataset was divided into training and validation sets according to standard machine learning practice, which allocates 80% of the data for training and 20% for validation. In this case, a split was made from the data while maintaining temporal independence between images in the dataset, which is critical for microscopy data. As such, the first 1434 consecutive time-lapse frames of one sequence were allocated to training, while the remaining 366 were used for validation. This method of splitting prevents data leakage among data sets and minimizes the likelihood of biased performance estimates.

Although a temporal split may introduce a mild distribution shift as the cell culture matures over time, it was intentionally adopted here to preserve chronological independence between the training and validation frames and thereby avoid overly optimistic estimates caused by near-duplicate temporal neighbors. To assess whether the validation subset represented a substantially different regime, we compared split-level statistics. The class distributions were highly similar between the training and validation sets (Cell class: 88.2% vs. 88.4%), and the density statistics shown in Fig. 2 indicated only a moderate increase in average cell density in the validation partition rather than a qualitatively different population state. This protocol should therefore be interpreted as a conservative forward-in-time evaluation strategy for a single-sequence microscopy dataset: it does not eliminate all temporal dependence, but it substantially reduces the risk of leakage that would arise from random splitting of highly similar neighboring frames.

Fig. 2 provides a full statistical description of the B16BL6 data set. This dataset consists of a total of 1800 frames, within those frames there are a total of 93,637 labeled cellular instances. Of those labeled instances, 72,471 were allocated to the training set and 21,166 labeled instances to the validation set. The average density of the dataset is 52.0 cells per frame, although there is significant variance between frames (standard deviation ± 15.3 cells per frame).

The statistics on the B16BL6 data set indicate a significant class imbalance. There are 88.3% Cell-class instances (Class 0) and 11.7% Division-class instances (Class 1) resulting in an approximate 7.5:1 imbalance. The class imbalance reflects that a cell spends the majority of its lifecycle in the interphase stages of the cell cycle (G1, S and G2 combined will approximately last 20–23 h) compared with the time spent in mitosis (M phase) which lasts approximately 1 h in standard mammalian cell lines. The distributions of Cell-class instances between the training and validation sets were similar (88.2% vs. 88.4%, respectively) indicating that the temporal split preserved the class proportions closely and that the ratio of class instances remained stable across the two partitions.

In summary, these statistics demonstrate that, although the dataset is moderate in temporal scale, it offers extensive annotation coverage and a biologically oriented two-class framework. As such, the supervision accessible for detector training is significantly more comprehensive than a mere frame count would imply, while the two-class design intentionally targets the practically important distinction between general cell presence and mitotic division events in dense phase-contrast time-lapse imaging.

3.1.3. Dataset comparison and positioning

To contextualize the B16BL6 dataset within the landscape of existing public microscopy resources for cell detection and analysis, Table 1 presents a comparative overview against several widely referenced datasets. The comparison highlights a consistent pattern: existing public resources generally prioritize either large-scale heterogeneity among cell types, high-quality segmentation annotation, or a clear temporal sequence. To the best of our knowledge, there is currently no publicly available dataset directly comparable to B16BL6 in terms of temporal structure, detection-oriented annotation format, and semantic granularity for Cell/Division detection.

Scale and annotation density. As mentioned in 3.1.2, the B16BL6 comprises 93,637 bounding-box annotations distributed over 1800 frames with an average of 52.0 ± 15.3 objects per frame. This offers a comprehensive supervision signal to train and test the state-of-the-art deep learning detectors under standardized COCO-style protocols. While datasets like LIVECell [28] contain significantly higher annotation numbers, they compile images from various cell types and culture conditions into one dataset and are designed primarily for instance segmentation rather than temporally coherent detection-oriented benchmarking. The data science Bowl dataset in 2018 [26] similarly spans numerous independent biological experiments and imaging settings, offering extensive modality coverage without temporal continuity between frames. B16BL6 can therefore be interpreted as moderate in frame number, but significant in annotation density within the context of temporally structured phase-contrast microscopy.

Temporal structure. The 1800 frames of B16BL6 are consecutive images drawn from a single uninterrupted time-lapse acquisition. Therefore, each frame is captured using the same imaging configuration and in a continuous biological progression, facilitating assessment within a temporally uniform data framework and enhancing the annotation process employed to construct the dataset. Among the datasets included in Table 1, only the Cell Tracking Challenge [27] offers fully sequential temporal data in a broadly comparable sense. The Cell Tracking Challenge datasets are annotated with segmentation contours and do not include semantic labels to distinguish dividing from non-dividing cells. EVICAN [30] comprises heterogeneous microscopy data from several cell lines; the published dataset description does not indicate the temporal ordering within the cell lines, and its annotations are primarily segmentation-based rather than detection-focused.

Division class annotation. A key distinguishing characteristic of B16BL6 relative to the datasets included in this comparison is the explicit annotation of a *Division* class alongside the general *Cell* class, since none of the listed datasets provides a Division class in a detection-compatible bounding-box format. This is unsurprising given that labeling the pre-cytokinetic phase in unlabeled phase-contrast cell imagery is both labor-intensive and morphologically indeterminate. B16BL6 explicitly addresses this issue: the *Division* class captures the last mitotic state, prior to cytokinesis, using a bounding box format that is compatible with standard COCO-style evaluation methods. The observed 7.5:1 *Cell*-to-*Division* imbalance is not a stochastic sampling artifact; rather, it reflects the biological asymmetry between the long interphase period and the shorter mitotic phase of the cell cycle. Maintaining this imbalance thus provides a realistic benchmark for assessing architectural robustness in the context of minority-class scarcity.

Annotation format and evaluation compatibility. The datasets listed in Table 1 are annotated primarily with pixel-level instance segmentation masks, which encode object boundary information not present in bounding-box labels, necessitating assessment frameworks distinct from those employed for detection. On the other hand, B16BL6 provides dense bounding-box annotations that are natively compatible with COCO-style detection metrics [mAP@0.50, mAP@0.50:0.95], thereby facilitating standardized comparisons of one-stage, two-stage, and transformer-based architectures under a unified evaluation protocol. Additionally, the 1600×1200 native resolution preserves intricate morphological details essential for distinguishing transient mitotic features from adjacent non-dividing cells on dense phase-contrast images.

Overall, B16BL6 complements rather than replaces existing public microscopy resources. Its main contribution is to fill the specific niche of temporally consistent, semantically distinct, high-resolution, bounding-box-annotated phase-contrast microscopy data. Its utility resides in (i) developing and testing object detection models for phase-contrast time-lapse microscopy using COCO-style evaluation; (ii) examining architectural behavior under biologically realistic class imbalance; and (iii) facilitating scalable annotation workflows that can be adapted to other time-lapse microscopy datasets. At the same

B16BL6 Dataset: Comprehensive Statistics

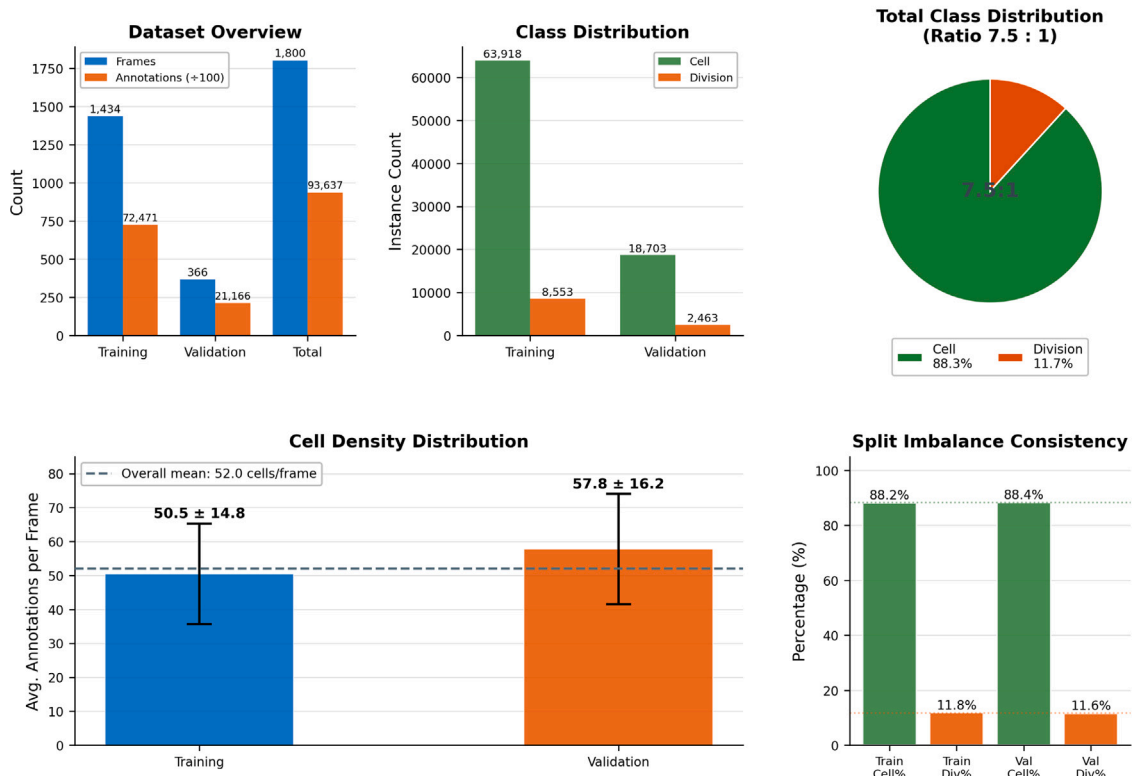


Fig. 2. Comprehensive statistics of the B16BL6 dataset, including dataset size, training/validation split, class distribution, and annotation-density characteristics. The figure shows the total number of frames and labeled instances, the Cell-Division imbalance, and the average number of annotations per frame across the two partitions.

Table 1

Comparison of B16BL6 with representative public microscopy datasets for cell detection and segmentation. “Varies” indicates that the property is not fixed across the dataset. Existing comparison datasets primarily use segmentation-mask annotations and do not provide detection-compatible bounding-box labels with an explicit Division class.

Dataset	Frames	Annotations	Resolution	Cell types	Modality	Temporal	Annotation method
B16BL6 (Ours)	1,800	93,637	1600 × 1200	Melanoma	Phase-contrast	Yes	3-Stage (Manual+Semi+Auto)
DSB 2018/BBBC038 [26]	841	37,333	Varies	Multiple	Multiple	No	Fully Manual
Cell Tracking Challenge [27]	Varies	Varies	Varies	Multiple	Multiple	Yes	Mixed
LIVECell [28]	5239	1,686,352	704 × 520	8 types	Phase-contrast	No	Mixed (Manual Annotations + Synthetic Datasets)
NeurIPS-CellSeg [29]	>1,500 ^a	Varies	Varies	Multiple	Multiple	No	Fully Manual Curated Labeled Benchmark
EVICAN [30]	4600	≈26,000	Varies	30 lines	Multiple grayscale contrasts	Not specified	Expert Manual/Partial Annotation

^a NeurIPS-CellSeg reports more than 1500 labeled images from over 50 biological experiments; exact annotation counts vary across released subsets [29].

time, this comparison highlights a practical constraint of the present benchmark: directly comparable public datasets for external validation remain scarce because most existing microscopy resources rely on segmentation-based annotations and do not provide an explicit Division class in a detection-compatible format.

3.2. Model architectures

In this study, we implemented seven different deep learning models, each representing a distinct detection paradigm or methodological

philosophy. The chosen models were based on well-established techniques and cutting-edge innovations, thereby enabling an exhaustive evaluation of the prevailing state-of-the-art competencies.

3.2.1. YOLOv8x

In 2023, Ultralytics released YOLOv8 [31], a series of detection models, with YOLOv8x as the largest model. To achieve this level of performance, the architecture of YOLOv8 contains a number of innovative features, including data augmentations like Mosaic Augmentation in its input pipeline; a backbone architecture that includes both Spatial Pyramid Pooling Fast (SPPF) and C2f modules; and a neck structure

based on PANet/FPN to improve feature propagation across different scales [32].

One of the most significant differences between YOLOv8 and earlier detection systems is the introduction of anchorless detection, which decreases the complexity of the learning process and therefore increases performance. Consequently, with fewer boxes predicted by the model, the speed of the Non-Maximum Suppression (NMS) post-processing step is dramatically improved [33]. With its combination of speed and accuracy, YOLOv8 is a suitable tool for live-cell microscopy and for monitoring divisions in real-time.

In addition to its many new features, YOLOv8 utilizes optimized tensor operations and hardware-aware design decisions that improve inference efficiency on modern GPUs. As a result, even the largest version, YOLOv8x, can provide real-time inference under appropriate hardware conditions while maintaining strong detection performance.

3.2.2. YOLOv11-L

The Ultralytics team released YOLOv11 in 2024 [34], building upon the success of YOLOv8 and the improvements of different community versions named YOLOv9 and YOLOv10. Substantial changes were made in architecture and training methods to further improve accuracy and real-time inference efficiency. In this paper, we used YOLOv11-L, the large version from the family. The classical three-stage architecture is maintained in this version, but there are meaningful changes at every stage of dataflow. The backbone component's main task is to extract features from the input image, in which a more efficient C3k2 block is used in place of C2f blocks. While the C2f block introduced dual fusion pathways for improved feature propagation compared to standard CSP blocks, C3k2 is designed to improve feature extraction efficiency [35]. After the same SPPF module used by previous versions, there is a new Cross Stage Partial with Spatial Attention (C2PSA) block that aims to strengthen multi-scale feature aggregation. The neck combines the data representing features at various scales, and the head is responsible for the final prediction [36].

The optimizations throughout the architecture of YOLOv11 enable faster throughput while maintaining high detection quality, making YOLOv11-L suitable for high-resolution microscopy workflows where rapid processing and precise identification of small cellular structures are critical.

3.2.3. RT-DETR-L

RT-DETR (Real-Time Detection Transformer) was introduced by Baidu as a solution to the limitations of DETR-style object detectors, particularly their slow convergence and high computational cost [20]. While RT-DETR preserves the overall encoder-decoder architecture of DETR, it incorporates several efficiency-oriented design choices that make end-to-end detection feasible in real-time settings. The model begins with a DINO-Deformable-ResNet-50 backbone, which extracts multi-scale features from the image. These features are then processed by a hybrid encoder composed of two complementary components: Attention-based Intra-scale Feature Interaction (AIFI), responsible for modeling relationships within each feature level, and CNN-based Cross-scale Feature Fusion (CCFF), which aggregates information across different resolutions [21]. After encoding, the model employs a Transformer decoder enhanced with auxiliary prediction heads and an uncertainty-driven query selection strategy that prioritizes the most reliable object queries. This refinement allows RT-DETR to generate accurate detections, bounding boxes, and class scores, while maintaining real-time performance, similar to how Swin Transformer efficiently captures hierarchical patterns in small-scale visual elements. In this paper, we used RT-DETR-L, the large version from the family.

3.2.4. SSD (single shot MultiBox detector)

SSD is a single-stage detector designed for fast inference and easy training without a region proposal stage [37]. It uses a convolutional backbone (commonly VGG16), and multi-scale feature maps are generated from progressively smaller resolution layers. Lower resolution layers detect larger scale objects, while higher resolution layers detect smaller scale ones. Each layer predicts bounding boxes and class scores simultaneously, enabling direct detection of objects with varying sizes in a single forward pass.

The model assigns predefined anchor boxes with different aspect ratios at each feature level and refines them directly during training. This approach is based on the observation that in various datasets, bounding boxes for the same classes tend to have similar shapes and sizes. This makes training the model more stable.

SSD is computationally efficient and well-suited for embedded or resource-limited environments. However, compared with newer designs, it may struggle with extremely small objects such as cells due to coarser feature resolution in deeper layers.

3.2.5. Faster R-CNN

Faster R-CNN is a two-stage object detection model that has been the foundation of multiple recent state-of-the-art models thanks to its high accuracy and strong localization performance [5]. The first stage is a Region Proposal Network (RPN) responsible for proposing regions in images that potentially contain classifiable objects. The second stage predicts the class of the object on the listed regions and refines their bounding box.

The first stage has a convolutional backbone (usually VGG or ResNet), which builds a feature map. After that, at every point on the feature map, anchor boxes are generated with different sizes and aspect ratios, which are then predicted to be either background or foreground, background being a region likely not containing an object, while foreground is an area that likely contains an object [38].

In the second stage, region proposals are fixed to the required input size for the networks which on the one hand classify the objects and on the other hand refine the bounding boxes using regression, adjusting their parameters to improve their fit to the objects.

Faster R-CNN excels in scenarios requiring precise detection of small or overlapping objects, making it a reliable candidate for tracking morphological variations in densely populated microscope images [39]. However, its two-stage design generally results in slower inference compared to one-stage detectors like YOLO.

3.2.6. Mask R-CNN

Built on Faster R-CNN, Mask R-CNN extends its architecture by adding a third branch responsible for predicting segmentation masks [9]. The original two branches output a class label and a bounding box, while the new branch outputs a pixel-level mask in addition to them. The segmentation technique called semantic segmentation classifies every pixel into a category but does not differentiate instances, while instance segmentation is able to assign each instance of an object into a different value on the output mask. The authors of Mask R-CNN highlight that while instance segmentation might seem like a complex challenge to achieve high accuracy, their simple and efficient method can compete with other state-of-the-art instance segmentation models.

Mask R-CNN applies a new RoIAlign operation instead of region-of-interest pooling (RoIPool). In Faster R-CNN, RoIPool was used to convert each region proposal into a fixed-size feature map. It relies on quantizing coordinates into discrete bins, which introduces small but significant misalignments between the original image and the extracted features. While these rounding errors are tolerable for bounding-box detection, they become problematic for pixel-level mask prediction, where precise spatial alignment is essential. RoIAlign addresses this issue by eliminating quantization altogether and using bilinear interpolation to compute feature values at exact floating-point locations. This simple modification greatly improves localization accuracy and leads to significantly better mask results. Consequently, Mask R-CNN achieves state-of-the-art performance in both object detection and instance segmentation.

3.2.7. Libra R-CNN

Libra R-CNN is a two-stage object detector designed to address the imbalanced feature distribution that commonly occurs in other CNN-based detection pipelines [10]. Instead of modifying the backbone or adding more layers, Libra R-CNN focuses on improving feature quality by enforcing balanced learning at the feature extraction, sample selection, and optimization levels. The model introduces IoU-balanced sampling, a strategy that selects region proposals according to their intersection-over-union scores rather than random sampling. This ensures a more even distribution of positive and negative samples and prevents the detector from being dominated by numerous low-quality or low-overlap candidates. Additionally, Balanced Feature Pyramid aggregates multi-scale features into a unified representation and refines them using a non-local operation, ensuring that all pyramid levels receive equally informative features [40]. To further stabilize training, Libra R-CNN employs Balanced L1 Loss, which reduces the influence of overly easy samples and encourages better regression performance on harder examples.

These combined enhancements allow Libra R-CNN to achieve stronger localization and classification accuracy compared to standard Faster R-CNN variants, making it particularly useful for microscopy applications where feature imbalance and object-scale variability are common challenges.

3.3. Training protocol and hyperparameters

Two training protocols were evaluated in this study. The first used the baseline hyperparameters associated with each model's standard implementation. The second introduced a limited fine-tuning strategy intended to test whether additional optimization could improve performance. In this setting, cosine learning-rate scheduling was enabled (`cos_lr=True`), and early stopping was applied using patience values of 5 and 15 epochs. These settings were chosen to examine whether modest optimization changes could improve generalization while reducing the risk of overfitting. In addition to the baseline and fine-tuning settings, we conducted an imbalance-aware training experiment for two of the strongest single-stage models (YOLOv8x and YOLOv11-L) using class-weighted loss in order to evaluate whether simple reweighting could reduce the Cell-Division performance gap under the dataset's approximately 7.5:1 class imbalance.

All experiments were conducted using a Google Colab Pro environment, leveraging an NVIDIA Tesla T4 GPU with 15 GB available memory. For this project, the High-RAM runtime configuration was enabled, allowing the use of up to 25 GB of system memory.

To ensure a controlled comparison, all models were trained and evaluated using the same training/validation split, the same class definitions, the same hardware environment, and the same COCO-style evaluation protocol described in Section 3.4. For the baseline benchmark, a matched training budget was used across models, and where implementation settings permitted alignment, the input size was set to 640×640 . However, the benchmark was not constructed by forcing all architectures into an artificially identical preprocessing regime. Architecture-specific input-size requirements, framework-native augmentation behavior, and standard post-processing settings were retained within their respective implementations, since these form part of each model's standard operating configuration. Accordingly, the present study should be interpreted as a controlled benchmark under matched data, hardware, and evaluation conditions, while still preserving architecture-appropriate implementation settings.

3.4. Evaluation metrics

Proper evaluation of a classification model goes far beyond the ratio of correct and total predictions, that is the accuracy metric. The accuracy of a model trained on unbalanced data may indicate promising performance, but testing in real-world conditions may reveal numerous

potentially costly false positives (FP) or false negatives (FN). Object-detection models also introduce the challenge of evaluating localization performance in addition to predicting labels of objects.

1. Precision and Recall

The ratio of true positives (TP) and all positive predictions (TP + FP) measures a model's precision, which numerically expresses its trustworthiness when a sample is classified as positive. Similarly, recall, the ratio of true positives and all positive samples (TP + FN), characterizes a model's ability to identify all actual positives.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3.4.1)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3.4.2)$$

2. F1 Score

Ideally, both precision and recall should be equally high. A large imbalance between the two can lead to misleading confidence, increasing the amount of either false positives or false negatives. To measure the balance between precision and recall, their harmonic mean, also known as the F1 Score is calculated.

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3.4.3)$$

3. Average Precision

To classify numerical confidence scores into classes, a threshold value is needed. If the threshold is treated as a variable, a precision–recall (PR) curve can be plotted, where resolution depends on the number of tested threshold values. Average precision (AP) is computed by calculating the area under the PR curve and weighting each precision value by the incremental change of recall at the corresponding vertex.

$$AP = \sum_{i=1}^N (R_i - R_{i-1}) P_i \quad (3.4.4)$$

4. Mean Average Precision

Finding AP for each class and then averaging them over the number of classes is the Mean Average Precision (mAP). This metric provides insights into object-detection model performance by evaluating both localization and classification accuracy across multiple classes and confidence thresholds. In such models, a prediction is considered a TP if the predicted class is correct and the Intersection-over-Union (IoU) of the predicted bounding box and the ground-truth box exceeds a set threshold (50% in case of mAP@0.5).

$$mAP = \frac{1}{C} \sum_{c=1}^C AP_c \quad (3.4.5)$$

5. Inference Speed

Measured in frames per second (FPS) or milliseconds, inference speed quantifies the time it takes for a model to process a data sample. This metric is critical for applications where rapid decision-making is a necessity. Its value heavily depends on input size, architecture, and available computational hardware.

6. Computational Complexity

In machine learning, Computational Complexity describes the computational resources required by a model, including the number of floating-point operations (FLOPs), the number of trainable parameters, and memory requirements. For models deployed on devices with strict resource constraints (for example edge devices), low computational complexity is essential to achieve acceptable performance.

Table 2

Comprehensive performance comparison of object detection models on B16BL6 dataset.

Model	mAP @0.50	mAP @0.50:0.95	AP@0.50 Cell	AP@0.50 Division	Best F1 / Conf	Precision (Peak)	Recall (Peak)	FPS (E2E)
<i>One-Stage Detectors</i>								
YOLOv8x (Baseline)	0.837	0.627	0.870	0.805	0.77/0.21	1.000	0.910	11.81
YOLOv8x (FT, Pat=5)	0.824	0.607	0.855	0.794	0.77/0.17	1.000	0.900	15.35
YOLOv8x (FT, Pat=15)	0.815	0.605	0.864	0.767	0.76/0.27	1.000	0.900	15.21
YOLOv11-L (Baseline)	0.811	0.607	0.861	0.761	0.75/0.23	1.000	0.910	20.15
YOLOv11-L (FT, Pat=5)	0.794	0.589	0.853	0.768	0.75/0.44	1.000	0.930	18.68
YOLOv11-L (FT, Pat=15)	0.798	0.589	0.859	0.737	0.76/0.34	1.000	0.900	18.92
RT-DETR-L (Baseline)	0.786	0.586	0.868	0.705	0.75/0.63	1.000	0.920	23.98
RT-DETR-L (FT, Pat=5)	0.776	0.578	0.819	0.734	0.72/0.49	1.000	0.740	25.05
RT-DETR-L (FT, Pat=15)	0.785	0.579	0.859	0.710	0.74/0.49	1.000	0.830	24.70
SSD300 (Baseline)	0.562	0.317	0.591	0.534	0.66/0.03	0.967	0.668	51.74
SSD300 (Pat=5)	0.669	0.408	0.691	0.647	0.70/0.16	1.000	0.849	55.15
SSD300/512 (Pat=5)	0.687	0.404	0.671	0.703	0.69/0.15	0.999	0.836	50.83
<i>Two-Stage Detectors</i>								
Faster R-CNN (Baseline)	0.787	0.566	0.816	0.758	0.80/0.54	0.992	0.875	12.78
Faster R-CNN (FT, Pat=5)	0.770	0.551	0.810	0.731	0.79/0.71	0.988	0.872	11.74
Mask R-CNN (Baseline)	0.769	0.552	0.811	0.727	0.80/0.70	0.986	0.868	5.98
Mask R-CNN (FT, Pat=5)	0.744	0.535	0.817	0.672	0.79/0.49	0.996	0.871	6.85
Libra R-CNN (Baseline)	0.761	0.554	0.789	0.733	0.78/0.20	0.993	0.844	12.34
Libra R-CNN (FT, Pat=5)	0.767	0.539	0.826	0.709	0.78/0.45	0.997	0.911	11.10

To provide a summary of deployment-oriented trade-offs, we also reported a supplementary efficiency score formulated as $\text{mAP}@0.50 \times \text{FPS}/\text{GFLOPs}$. We explicitly state that this should not be viewed as a standard metric and is included only as a heuristic indicator that summarizes deployment-oriented trade-offs among detection accuracy, inference speed, and computational cost in a single composite quantity. The main comparative conclusions of this study are derived from the standard metrics presented individually (e.g., $\text{mAP}@0.50$, $\text{mAP}@0.50:0.95$, class-specific AP, FPS, GFLOPs, and parameter count).

Computation of precision–recall and average precision (AP). Precision–recall curves were generated by sweeping the confidence threshold over its full range. Peak precision denotes the peak precision attained along the precision–recall curve at high confidence thresholds. For single-stage detectors, class-wise non-maximum suppression (NMS) followed the default Ultralytics post-processing settings. For two-stage detectors, region proposals and final detections were processed using the default per-class suppression within the torchvision detection pipeline. Average precision (AP) and mean average precision (mAP) were computed using COCO-style evaluation (continuous interpolation), reported at $\text{IoU} = 0.50$ ($\text{AP}@0.50/\text{mAP}@0.50$) and averaged over IoU thresholds 0.50–0.95 ($\text{mAP}@0.50:0.95$). Single-stage detectors were evaluated using Ultralytics’ built-in COCO evaluation pipeline, whereas two-stage detectors were evaluated using the pycocotools COCO API via the torchvision detection framework, under identical dataset splits, IoU settings, and category definitions to ensure comparability. Inference speed was measured in an end-to-end manner, including preprocessing and post-processing, following each framework’s standard inference pipeline.

3.5. Implementation details

Implementation of YOLOv8x, YOLOv11-L, and RT-DETR was carried out using the Ultralytics framework [31]. The remaining models, including SSD and the two-stage models, were implemented using PyTorch and torchvision libraries [41]. For the baseline protocol, all models were trained for 100 epochs. Fine-tuning experiments were then conducted as separate sensitivity analyses using the optimization changes described in Section 3.3.

4. Results

This section offers a comprehensive analysis of the seven SOTA detection models utilizing the B16BL6 microscopy dataset. The evaluation is designed to be progressively analyzed in the following subsections. All results were generated under consistent evaluation protocols to ensure fair comparisons among the models.

4.1. Overall performance comparison

Table 2 displays the overall performance for each of the model architectures tested based on various metrics (i.e., accuracy), robustness, and inference speed. It also illustrates the results from both the baseline and fine-tuned models. Thus, models can perform differently even among all of the evaluated models. The different levels of performance exhibited across the different models guided the model selection process for applications that involve the use of microscopy.

As seen in Table 2, YOLOv8x achieved the highest $\text{mAP}@0.50$ of 0.837, which demonstrates the strongest performance in detection capability. This outcome supports the effectiveness of large-capacity one-stage detectors in the context of dense cell microscopy data. Then the second-highest $\text{mAP}@0.50$ of 0.811 was achieved by YOLOv11-L, which corresponds to an approximate relative drop of 3.2%, highlighting diminishing accuracy gains despite increased computational cost.

For two-stage detectors, all three models achieved competitive performance but with lower accuracy compared to one-stage detectors. Faster R-CNN obtained the strongest performance among two-stage detectors with $\text{mAP}@0.50$ of 0.787, closely trailed by Mask R-CNN and Libra R-CNN, with $\text{mAP}@0.50$ scores of 0.769 and 0.761, respectively. Despite the benefits of these models from explicit region proposal mechanisms, their results remain inferior to the modern one-stage detectors under the same training protocol and dataset scale, as illustrated in Fig. 3.

In contrast to this, SSD based models showed a noticeable lower level of accuracy, with $\text{mAP}@0.50$ values between 0.562 (SSD300) and 0.687 (SSD300/512, pat=5). The gap between YOLOv8x and the SSD models is more than 27 percentage points which illustrates the limitations of earlier styles of single-shot detection regarding their ability to deal with small, low-contrast cellular objects.

The substantial gap between $\text{mAP}@0.50$ and $\text{mAP}@0.50:0.95$ (with an average reduction of 0.21 points), reveals the challenge of precise

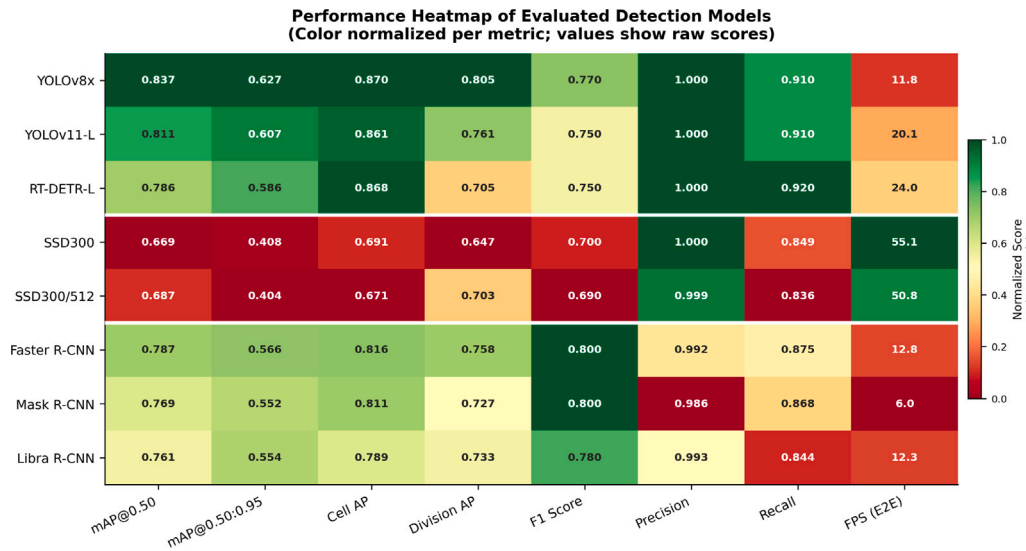


Fig. 3. Performance heatmap summarizing the baseline results of the evaluated detection models on B16BL6 across multiple metrics, including mAP@0.50, mAP@0.50:0.95, class-specific AP, F1 score, precision, recall, and end-to-end FPS. Values are shown directly in the cells to facilitate cross-model comparison of accuracy, class balance, and efficiency.

Table 3
Class-specific performance analysis (AP@0.50).

Model	Cell AP	Division AP	Gap (%)
YOLOv8x (Baseline)	0.870	0.805	6.5
YOLOv11-L (Baseline)	0.861	0.761	10.0
RT-DETR-L (Baseline)	0.868	0.705	16.3
Faster R-CNN (Baseline)	0.816	0.758	5.8
Mask R-CNN (Baseline)	0.811	0.727	8.4
Libra R-CNN (Baseline)	0.789	0.733	5.6
SSD300 (Pat=5)	0.691	0.647	4.4
SSD300/512 (Pat=5)	0.671	0.703	-3.2

bounding-box localization in microscopy images. Despite being able to detect objects very reliably, precise spatial alignment can still be difficult to achieve due to overlapping cellular environments and unclear boundaries as evidenced by the degradation seen in this experiment.

It should be emphasized that the provided precision values represent the greatest (peak) precision seen on the precision-recall curve, attained at high confidence thresholds. At these operational thresholds, models yield highly conservative predictions with few false positives, albeit at the expense of diminished recall. In this regime, all the models reach near unity levels (0.967–1.000), which reflects the robustness of predictions at high confidence thresholds. In contrast, recall exhibits substantial variation (0.668–0.930), indicating differences in the ability of each model to capture all relevant cells. Among all evaluated models, YOLOv11-L (Pat=5) achieves the highest recall (0.930), implying a sensitivity-precision trade-off that favors recall.

4.2. Class-specific performance imbalance sensitivity

One of the essential aspects of this study is the differing behavior exhibited by models in detection of Cell and Division instances. Fig. 4 and Table 3 illustrate these differences in terms of AP@0.50 values for each class and the corresponding performance gaps.

Based on the reported results in Table 3, all evaluated models demonstrate better performance on the Cell class compared to the Division class, except for SSD300/512 model. The performance gap between Cell and Division detection spans from 4.4% to 16.3%, confirming that division recognition is inherently more challenging. The observed disparity arises from three key factors: (i) substantial class

imbalance of approximately 7.5:1, as mentioned in 3.1.2, (ii) short-lived morphological changes during mitotic phases; and (iii) increased annotation uncertainty associated with division events.

Among all the architectures, RT-DETR exhibits the largest cell-division gap (16.3%), suggesting that, despite its competitive inference speed, transformer-based global attention mechanisms could be more sensitive to minority-class scarcity in dense microscopy images. On the other side, the two-stage detectors display smaller gaps, where Faster R-CNN (5.8%) and Libra R-CNN (5.6%) demonstrate the most balanced class-level performance. Under the evaluated training settings, two-stage detectors therefore appear more robust to class imbalance than the tested fast single-stage alternatives.

SSD300/512 represents an exception by achieving a higher AP for the Division class (0.703) than for the Cell class (0.671). This pattern suggests that increasing input resolution preferentially improves the detection of mitotic cells with distinct morphological characteristics, potentially retaining fine-scale spatial details that are not preserved at lower image resolutions.

To further investigate whether the class-level imbalance observed in one-stage detectors could be alleviated through loss reweighting, we performed an additional class-weighting experiment for both YOLOv8x and YOLOv11-L. The results are summarized in Table 4.

The additional experiment showed that simple class weighting did not improve minority-class division detection in either model. For YOLOv8x, mAP@0.50 decreased from 0.8370 to 0.8118. In addition, the Division AP@0.50 decreased from 0.805 to 0.759, while the Cell-Division gap increased from 6.5% to 10.5%. For YOLOv11-L, mAP@0.50 remained nearly unchanged (0.8110 to 0.8109), but the Division AP@0.50 decreased from 0.761 to 0.750, and the Cell-Division gap widened from 10.0% to 12.2%. These results suggest that, in the present microscopy setting, simple class weighting is insufficient to bridge the minority-class performance gap in fast single-stage detectors and may even degrade class-wise balance.

This subsection is intended to provide a focused analysis of Division detection, since minority-class mitotic events constitute the most challenging aspect of the benchmark.

4.3. Impact of fine-tuning strategies

The impact of fine-tuning pretrained models using the two hyperparameters (cosine learning rate scheduling and early stopping) is

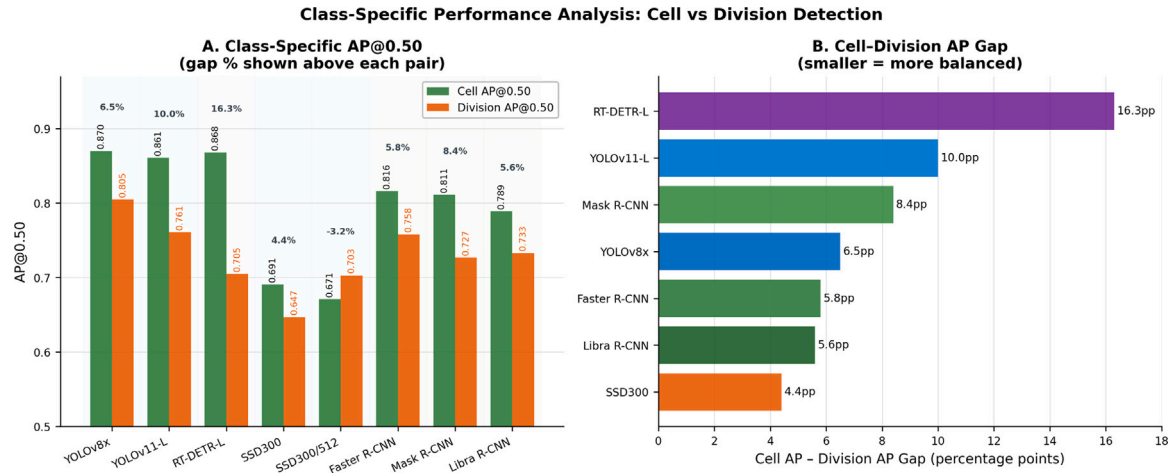


Fig. 4. Class-specific detection performance for the baseline models, comparing AP@0.50 for the Cell and Division classes together with the corresponding Cell-Division AP gap. Smaller gaps indicate more balanced performance under class imbalance, whereas larger gaps reflect greater difficulty in minority-class Division detection.

Table 4
Effect of class-weighted training on single-stage YOLO models.

Model	mAP@0.50	mAP@0.50:0.95	Cell AP@0.50	Division AP@0.50	Gap (%)
YOLOv8x (Baseline)	0.8370	0.6270	0.870	0.805	6.5
YOLOv8x (+ class weighting)	0.8118	0.5942	0.864	0.759	10.5
YOLOv11-L (Baseline)	0.8110	0.6070	0.861	0.761	10.0
YOLOv11-L (+ class weighting)	0.8109	0.5998	0.872	0.750	12.2

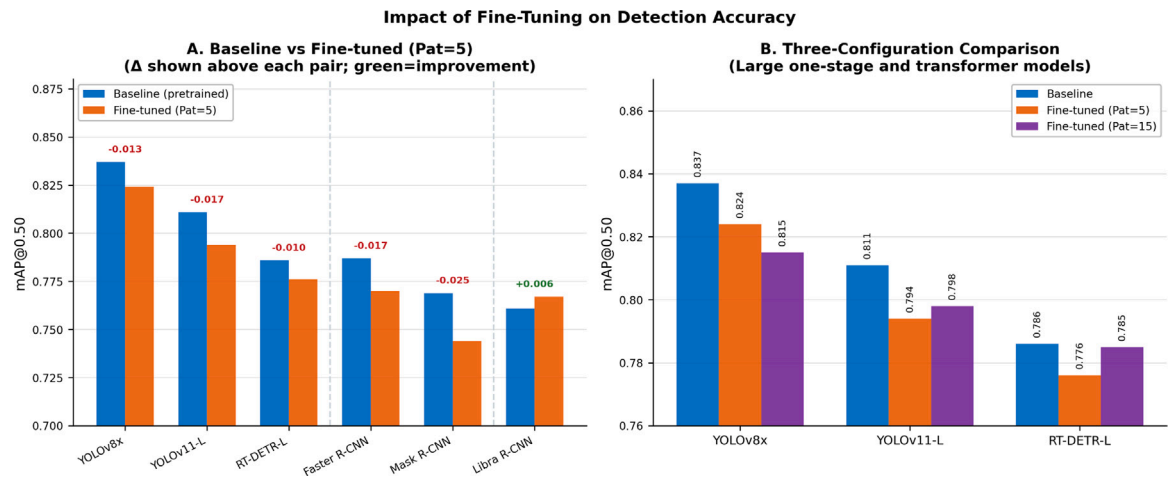


Fig. 5. Effect of fine-tuning strategies on detection accuracy, comparing baseline and fine-tuned configurations across the evaluated models. The figure illustrates how cosine learning-rate scheduling and early stopping affected mAP@0.50, highlighting that fine-tuning was not uniformly beneficial and in most cases produced small performance reductions.

summarized in Table 5 and illustrated in Fig. 5. Contrary to conventional expectations, fine-tuning does not universally improve detection performance.

Comparing the results among the large one-stage models shows a consistent decrease in overall performance. YOLOv8x experiences a reduction of 1.6% when using (Pat=5) and 2.6% with (Pat=15). Similarly, YOLOv11-L demonstrates performance degradation of 2.1% and 1.6%, respectively. These findings suggest that high-capacity models trained on large and diverse datasets already operate close to their optimal representations, and further fine-tuning with the specific hyperparameters used in this investigation may lead to overfitting effects.

RT-DETR-L records a minimal performance degradation and remains close to the baseline level with longer patience (−0.1%). The

observed behavior may be attributed to the regularizing properties of transformer architectures and their reliance on global contextual modeling.

For two-stage detectors, Libra R-CNN is the only model that benefits from fine-tuning (+0.8%), compared to Faster R-CNN and Mask R-CNN, which exhibit performance drops. Therefore, this observation emphasizes that the effectiveness of fine-tuning depends on the model architecture and should not be assumed to be a universally optimal approach.

These findings should not be interpreted as evidence that microscopy imaging as a field is broadly affected by over-optimization. Rather, under the limited fine-tuning strategies evaluated here, most models seem to operate close to a suitable operating point for this visually homogeneous dataset, such that further optimization yielded

Table 5
Impact of Fine-Tuning on Model Performance.

Model	Baseline mAP@0.50	FT (Pat=5) mAP@0.50	FT (Pat=15) mAP@0.50	Best Config
YOLOv8x	0.837	0.824 (−1.6%)	0.815 (−2.6%)	Baseline
YOLOv11-L	0.811	0.794 (−2.1%)	0.798 (−1.6%)	Baseline
RT-DETR-L	0.786	0.776 (−1.3%)	0.785 (−0.1%)	Baseline
Faster R-CNN	0.787	0.770 (−2.2%)	–	Baseline
Mask R-CNN	0.769	0.744 (−3.3%)	–	Baseline
Libra R-CNN	0.761	0.767 (+0.8%)	–	FT (Pat=5)

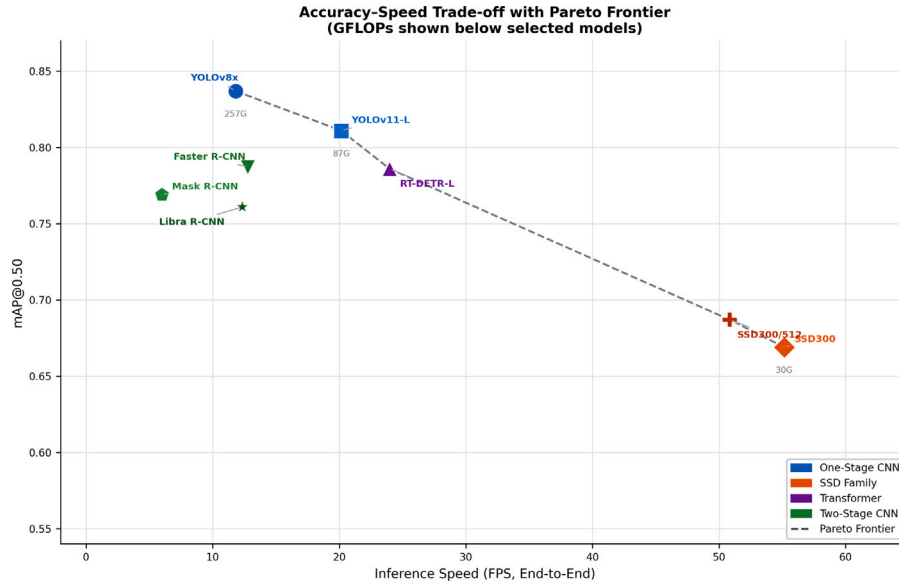


Fig. 6. Accuracy-speed trade-off for the baseline detector configurations on B16BL6, plotting mAP@0.50 against end-to-end inference speed (FPS). The dashed Pareto frontier highlights models that provide the strongest trade-offs between detection accuracy and throughput, while selected GFLOP values indicate relative computational complexity.

Table 6
Computational complexity and inference speed analysis.

Model	GFLOPs	Params (M)	FPS (E2E)
SSD300	30.48	23.88	55.15
YOLOv11-L	86.58	25.28	20.15
RT-DETR-L	103.44	31.99	23.98
Faster R-CNN	133.93	41.30	12.78
Libra R-CNN	133.93	41.30	12.34
Mask R-CNN	200.78	43.93	5.98
YOLOv8x	257.40	68.13	11.81

small benefit and in multiple cases led to mild degradation. The modest improvement observed for Libra R-CNN further indicates that this behavior is architecture- and setting-dependent rather than universal.

4.4. Accuracy-speed trade-off and computational efficiency

The strong trade-off between detection accuracy and inference speed is shown in Fig. 6, while Table 6 reports detailed computational complexity metrics. The efficiency score is reported solely as a supplementary heuristic summary of deployment trade-offs and should be interpreted in conjunction with the standard metrics rather than as a standalone ranking criterion.

SSD300 has the fastest detection time at 55.15 FPS (frames per second) using a relatively low computational budget of 30.48 GFLOPs (billion floating point operations per second), thus allowing for real-time processing of images in a live fashion. However, it has a correspondingly lower accuracy; therefore, its use will be limited primarily to situations where approximate detections can be made (i.e., high-throughput screening).

Conversely, the highest computational overhead is observed for Mask R-CNN (200.78 GFLOPs) with an inference speed of approximately 6 FPS. This made it unsuitable for real-time deployment despite providing competitive detection accuracy.

With respect to detection efficiency and architectural balance among all models tested, YOLOv11-L provides the most favorable accuracy-speed trade-off among the evaluated models, achieving both a detection rate of 20.15 FPS and an mAP@0.50 of 0.811, with only about one-third of YOLOv8x's computational costs. Thus, YOLOv11-L is 70.6% faster than YOLOv8x while achieving only a 3.1% decrement in detection accuracy due to the architectural improvements that have been implemented into YOLOv11.

RT-DETR-L has an approximate 24 FPS speed while utilizing the attention mechanism, demonstrating that hybrid architecture approaches such as CNN-Transformer can be developed with computational (deployment) efficiency in mind. However, the detection accuracy of RT-DETR-L for imaging cells from dense microscopy is lower than that of the CNN models.

The correlation analysis demonstrates a strong negative correlation between model complexity and inference speed (−0.87 for GFLOPs vs. FPS), reinforcing the importance of computational efficiency as a decisive factor for real-time biological imaging systems.

4.5. Summary of key findings

There is not an overall best architecture according to all metrics; therefore, the selection of models should occur according to needs of the application:

1. YOLOv8x achieved the highest detection accuracy in the current benchmark.

- YOLOv11-L provided the most favorable balance between accuracy and inference speed for real-time processing.
- If the presence of class imbalance was an issue, multi-stage detectors appeared more robust but had the slowest inference times.
- SSD-based models are for situations when the speed of inference is more important than accuracy.

These guidelines will help with determination of which types of detection models may work well within automated microscopy applications.

5. Discussion

This study provides a systematic comparison of state-of-the-art deep learning detectors for cell and division detection in phase-contrast microscopy. Rather than focusing only on peak accuracy, the analysis considers four practical aspects of deployment: detection performance, computational cost, robustness to class imbalance, and sensitivity to fine-tuning. Together, these factors determine how useful each architecture is for real microscopy applications.

It is important to note that the present study does not aim to introduce a new detection architecture; rather, its contribution lies in establishing a data-centric benchmark resource and a systematic empirical comparison for dense phase-contrast microscopy.

5.1. Accuracy–efficiency trade-offs in dense microscopy scenes

The experimental results reveal that high-capacity one-stage detectors, especially the YOLOv8x, demonstrate the highest absolute detection accuracy on the evaluated B16BL6 dataset. This finding suggests that densely populated cellular scenes can benefit from the use of deep hierarchical feature representations along with large spatial receptive fields. However, the modest accuracy improvement over compact models is associated with a substantial escalation in computational complexity, surpassing 250 GFLOPs.

From an implementation perspective, it is evident that there are diminishing returns on model complexity for detectors when there is a point at which they will no longer provide an improvement in detection accuracy when detecting cells in microscopy images. Even though YOLOv8x is appropriate for offline and accuracy-sensitive uses, its high computational cost prevents its use in resource-limited or real-time applications. In contrast, YOLOv11-L provides a strong balance between accuracy and efficiency. Its performance remains close to the top-performing models while requiring substantially less computational power, making it a practical choice for real-time microscopy applications.

5.2. Division detection and sensitivity to class imbalance

All evaluated architectures showed a consistent performance gap between Cell and Division detection, highlighting the inherent difficulty of mitosis recognition. Division events are less frequent, morphologically variable, and more visually ambiguous than non-dividing cells, which increases both the difficulty of model learning and the uncertainty of annotation.

As mentioned in Section 4.2, when comparing one-stage and transformer-based detectors with two-stage detectors, smaller performance gaps between Cell and division detection are observed in two-stage architectures. This suggests that explicit region proposal and refinement mechanisms may contribute to improved robustness under strongly skewed class imbalance in this microscopy settings. In contrast, RT-DETR-L exhibits the largest performance disparity, suggesting that global attention mechanisms may preferentially capture dominant visual structures when class distributions are highly imbalanced.

SSD300/512 represents an exception by achieving a higher AP for the Division class (0.703) than for the Cell class (0.671). This pattern suggests that increasing input resolution preferentially improves

the detection of mitotic cells because higher-resolution input is more likely to preserve fine-scale morphological cues associated with division events, such as localized boundary reshaping, rounding, and other high-frequency structural changes that may be attenuated at 300×300 resolution. However, the present study did not include a dedicated feature-attribution analysis, so this interpretation should be regarded as biologically and computationally plausible rather than directly verified.

To test whether the imbalance sensitivity of fast single-stage detectors could be reduced through a simple optimization change, we evaluated class-weighted training for YOLOv8x and YOLOv11-L. However, this strategy did not improve Division detection in either model. Instead, the Cell–Division AP gap increased in both cases, with YOLOv8x increasing from 6.5% to 10.5% and YOLOv11-L increasing from 10.0% to 12.2%. This finding suggests that the imbalance robustness observed in two-stage detectors is not merely a consequence of unweighted optimization in YOLO-based models, but may also be related to architectural properties such as region proposal and refinement mechanisms. At the same time, the present experiment only evaluated simple class weighting, and therefore more advanced imbalance-aware objectives remain worthy of future investigation.

Taken together, these results suggest three recurring sources of difficulty in Division detection within the present benchmark: (i) minority-class scarcity, which increases sensitivity to imbalance during optimization; (ii) the transient and morphologically ambiguous appearance of late mitotic cells, which can blur the boundary between *Cell* and *Division* instances; and (iii) the loss of fine-scale visual detail at lower input resolution, as indirectly reflected by the relative improvement of SSD300/512 on the Division class. These observations provide a more focused interpretation of likely failure modes in division detection, although they should be understood as comparative inferences from the reported results rather than as a formal per-instance error taxonomy.

5.3. Fine-tuning behavior in homogeneous microscopy data

The fine-tuning experiments indicate that additional optimization on the target dataset did not consistently improve performance. In most cases, fine-tuning produced a measurable reduction in detection accuracy. Large one-stage architectures were especially sensitive to this behavior, likely because the visual homogeneity of the microscopy data increased the risk of overfitting relative to large-scale natural-image pretraining.

The availability of pre-trained representations developed on large and diverse datasets suggests that these models already capture many of the low- to mid-level visual attributes relevant to the cellular morphology patterns observed here. Continued optimization under a limited set of task-specific hyperparameter changes may therefore increase sensitivity to dataset-specific variation rather than improve generalization. The moderate improvement of Libra R-CNN suggests that this behavior is not universal, but depends on both the model architecture and the optimization setting. Accordingly, the present results should not be interpreted as evidence of broad over-optimization in microscopy imaging models, but rather as an indication that additional fine-tuning on visually homogeneous microscopy data may yield diminishing returns and should be validated empirically rather than assumed to be beneficial.

The transferability of these findings should therefore be interpreted primarily within the domain of adherent phase-contrast microscopy, as significant morphological or modality changes are likely to necessitate further adaptation before reliable deployment can be expected.

5.4. Limitations of the study

Although the evaluation offers a comprehensive assessment, several limitations should be explicitly acknowledged to clarify the scope of the conclusions. These limitations are especially important because the

present benchmark is intentionally focused on a single cell line, a single microscopy modality, and a bounding-box-based detection formulation.

A first limitation is that all experiments were conducted on a single cell type (B16BL6) acquired using a single imaging modality (phase-contrast microscopy). Therefore, the present results do not support direct claims of zero-shot generalization to substantially different cellular morphologies or imaging modalities. Transfer would be expected to be most favorable for morphologically similar adherent cells imaged under comparable phase-contrast conditions, whereas performance would likely decrease for non-adherent cells and, more substantially, for fluorescence microscopy. Accordingly, additional domain adaptation, fine-tuning, or retraining would likely be required for reliable deployment in such settings. At the same time, the three-stage annotation methodology is more general than the currently trained detection models and could still be used to construct domain-specific labeled datasets for new cell types or imaging modalities.

A second limitation is that the study was restricted to bounding-box-based detection methods, excluding pixel-level segmentation and long-term lineage modeling. Although detection-based outputs support many quantitative tasks, applications requiring precise cellular boundary localization or temporally resolved lineage analysis may require segmentation-based or spatiotemporal modeling methods.

Another limitation is that the dataset was restricted to two classes (*Cell* and *Division*). Although this formulation represents a biologically meaningful and challenging detection task, the limited class taxonomy does not capture finer distinctions such as intermediate mitotic stages or additional cellular states. Moreover, the substantial Cell-Division imbalance reflects the natural temporal asymmetry of the cell cycle, but it also increases the difficulty of minority-class evaluation and limits the extent to which conclusions can be generalized beyond this focused detection setting.

In addition, although the Stage 3 tracking component employed a UKF-based motion model inherited from our previously published microscopy detection-and-tracking framework, the present study did not include a dedicated ablation against a standard linear Kalman filter. Accordingly, the use of UKF here should be interpreted as an adopted design choice within the annotation pipeline rather than as a separately benchmarked methodological claim.

A further limitation is that the evaluation was performed using a single-sequence temporal split. Although this choice reduces data leakage and better reflects forward-in-time deployment, later validation frames may still be moderately more challenging due to culture maturation effects such as increased density, altered morphology, or more frequent cell-cell interactions.

A further limitation is that the present benchmark was evaluated only on the proposed B16BL6 dataset and did not include cross-dataset validation. Although this is partly due to the scarcity of directly comparable public microscopy resources with temporally structured, detection-compatible annotations and an explicit Division class, the absence of external validation limits the extent to which the present findings can be generalized beyond this dataset. In addition, while annotation quality was supported through trained-expert labeling, domain-specialist verification, consensus adjudication, pseudo-label quality checks, and random-sample review of automated annotations, the study did not include a formal inter-annotator agreement analysis or a comprehensive per-instance error taxonomy. These aspects remain important directions for future work.

Although an additional imbalance-aware experiment using class-weighted training was conducted for YOLOv8x and YOLOv11-L, the present study did not evaluate other rebalancing strategies, including focal loss, asymmetric losses, resampling-based methods, or threshold calibration. Therefore, the negative result observed for simple class weighting should not be interpreted as ruling out all possible optimization-based improvements for minority-class division detection in fast single-stage detectors.

Similarly, although the improved Division performance of SSD300/512 suggests that higher input resolution preserves mitosis-related fine-scale morphology more effectively than lower-resolution input, the present study did not include a dedicated feature-attribution or error-analysis study to determine which specific visual cues were responsible for this effect.

More generally, although the present discussion identifies likely sources of failure affecting Division detection and temporal consistency, the study did not include a formal per-instance error taxonomy separating missed detections, class confusions, localization errors, and duplicate predictions. Likewise, although the enhanced DeepSORT-based tracker mitigated identity-switch errors through joint motion and appearance association, the present study did not include a dedicated overlap-specific error analysis or a separate quantitative breakdown of ID-switch events during prolonged cell-cell contact or severe occlusion. Accordingly, the reported 92.3% track consistency score should be interpreted as an aggregate indicator of temporal identity preservation rather than a complete characterization of all overlap-related failure modes.

Another limitation is that the fine-tuning analysis was limited to a narrow set of optimization changes, specifically cosine learning-rate scheduling and early stopping with two patience configurations. Therefore, the negative or neutral fine-tuning results observed for most models should not be interpreted as evidence against all forms of task-specific optimization, nor as a general statement about microscopy-specific architecture design more broadly.

In addition, the efficiency score reported in this study is a non-standard composite metric introduced only as a supplementary deployment-oriented heuristic. Because such a score depends on a particular mathematical formulation and implicitly weights accuracy, speed, and complexity in a specific way, it should not be interpreted as a definitive or universally applicable ranking criterion.

Finally, inference speed was evaluated using a specific hardware setup. While the comparisons between models can be reliably interpreted, absolute inference throughput may not generalize across different hardware systems.

These limitations do not compromise the comparative findings but instead define the boundaries of their applicability, thereby motivating future research extensions.

6. Conclusion

Rather than proposing a new object detection architecture, this work presents a data-centric contribution to microscopy image analysis through the release of the B16BL6 dataset, the formalization of a scalable annotation workflow, and a controlled benchmark of seven modern detection models. The study provides a comparative assessment of contemporary deep learning detectors for automated cell detection and division recognition in phase-contrast microscopy. In addition to benchmarking, we developed a scalable three-stage annotation workflow and produced the B16BL6 dataset (1800 frames at 1600×1200) with dense bounding-box labels for *Cell* and *Division* events, which is released alongside this study to support reproducible evaluation and future method development. Seven representative architectures spanning one-stage, two-stage, and transformer-based paradigms were evaluated under consistent splits and COCO-style metrics, with emphasis on accuracy, class-specific behavior under imbalance, computational cost, inference speed, and fine-tuning sensitivity.

Across models, one-stage detectors generally achieved higher overall detection accuracy, while two-stage detectors exhibited more balanced class-wise performance under severe class imbalance. YOLOv8x delivered the highest overall mAP, whereas YOLOv11-L offered the most favorable accuracy-speed trade-off for deployment-oriented pipelines. Two-stage detectors (Faster R-CNN, Mask R-CNN, and Libra R-CNN) showed stronger resistance to minority-class scarcity, while

SSD-based models provided high-throughput inference at a notable accuracy penalty. Fine-tuning was not uniformly beneficial and depended on the architecture and optimization configuration, indicating that additional task-specific optimization on homogeneous microscopy data should be treated as a stability-sensitive design choice and validated empirically rather than assumed to improve performance.

Overall, the results underline that detector selection in microscopy should be guided by application constraints: accuracy-critical studies favor high-capacity one-stage models, real-time pipelines benefit from balanced accuracy-speed solutions, and division-focused settings may prefer architectures with improved minority-class robustness. While the present study is limited to bounding-box detection in a single phase-contrast dataset, the released dataset and evaluation protocol provide a practical foundation for extending benchmarking to additional microscopy modalities, richer labels (e.g., instance masks or temporal phase labels), and spatiotemporal modeling that explicitly leverages time-lapse consistency.

Future work should extend the present benchmark into five directions: (i) controlled ablation of the inherited tracking components within the annotation workflow, including direct comparison between linear KF and UKF with respect to temporal consistency, identity-switch reduction, and computational overhead; (ii) evaluation of cross-cell-line and cross-modality generalization, especially for non-adherent cultures and fluorescence microscopy, to determine the extent of adaptation required beyond the current phase-contrast B16BL6 setting; (iii) investigation of which mitotic visual cues are most resolution-sensitive through targeted error analysis, saliency-based interpretation, or cross-resolution ablation studies; (iv) systematic study of fine-tuning strategies, including selective backbone freezing, layer-wise learning-rate schedules, and augmentation-based data diversification, to determine when adaptation of pre-trained representations is beneficial for specialized microscopy settings; and (v) expand the dataset by incorporating more detailed and complex class taxonomies (e.g., labels for various mitotic stages) and validating results using many different datasets to determine whether the results described in this study continue to remain stable in various biological contexts.

CRedit authorship contribution statement

Mokhaled N.A. Al-Hamadani: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Péter Szilágyi:** Writing – original draft. **Gabor Szeman-Nagy:** Validation, Resources, Data curation. **Andras Hajdu:** Funding acquisition. **Balazs Harangi:** Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Open access funding was provided by the University of Debrecen.

Data availability

The B16BL6 dataset is publicly available in the Open Science Framework (OSF). The permanent project DOI is <https://doi.org/10.17605/OSF.IO/5J276>.

References

- [1] Moen E, Bannon D, Kudo T, Graf W, Covert M, Van Valen D. Deep learning for cellular image analysis. *Nature Methods* 2019;16(12):1233–46. <http://dx.doi.org/10.1038/s41592-019-0403-1>.
- [2] Ker DFE, Eom S, Sanami S, Bise R, Pascale C, Yin Z, il Huh S, Osuna-Highley E, Junkers SN, Helfrich CJ, Liang PY, Pan J, Jeong S, Kang SS, Liu J, Nicholson R, Sandbothe MF, Van PT, Liu A, Chen M, Kanade T, Weiss LE, Campbell PG. Phase contrast time-lapse microscopy datasets with automated and manual cell tracking annotations. *Sci Data* 2018;5(1):180237. <http://dx.doi.org/10.1038/sdata.2018.237>.
- [3] Xing F, Yang L. Robust nucleus/cell detection and segmentation in digital pathology and microscopy images: A comprehensive review. *IEEE Rev Biomed Eng* 2016;9:234–63. <http://dx.doi.org/10.1109/RBME.2016.2515127>.
- [4] Al-Hamadani MNA, Poroszlav R, Szeman-Nagy G, Hajdu A, Hadjimetriou S, Ferrarini L, Harangi B. Improving cell detection and tracking in microscopy images using YOLO and an enhanced DeepSORT algorithm. *Sensors* 2025;25(14):4361. <http://dx.doi.org/10.3390/s25144361>.
- [5] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell* 2017;39(6):1137–49. <http://dx.doi.org/10.1109/TPAMI.2016.2577031>.
- [6] Wen T, Wu H, Du Y, Huang C. Faster R-CNN with improved anchor box for cell recognition. *Math Biosci Eng* 2020;17(6):7772–86. <http://dx.doi.org/10.3934/mbe.2020395>.
- [7] Sheng B, Zhou M, Hu M, Li Q, Sun L, Wen Y. A blood cell dataset for lymphoma classification using faster R-CNN. *Biotechnol Biotechnol Equip* 2020;34(1):413–20. <http://dx.doi.org/10.1080/13102818.2020.1765871>.
- [8] Raina S, Khandelwal A, Gupta S, Leekha A. Blood cells detection using faster-RCNN. In: 2020 IEEE international conference on computing, power and communication technologies. IEEE; 2020, p. 217–22. <http://dx.doi.org/10.1109/GUCON48875.2020.9231134>.
- [9] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 2961–9. <http://dx.doi.org/10.1109/ICCV.2017.322>.
- [10] Pang J, Chen K, Shi J, Feng H, Ouyang W, Lin D. Libra R-CNN: Towards balanced learning for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. Long Beach, CA, USA; 2019, p. 821–30. <http://dx.doi.org/10.1109/CVPR.2019.00091>.
- [11] Costa CS, Zanoni VAG, Curvo LRV, de Araújo Carvalho M, Boscolo WR, Signor A, dos Santos de Arruda M, Nucci HHP, Junior JM, Gonçalves WN, Diemer O, Pistori H. Deep learning applied in fish reproduction for counting larvae in images captured by smartphone. *Aquac Eng* 2022;97:102225. <http://dx.doi.org/10.1016/j.aquaeng.2022.102225>.
- [12] Jia D, Zhou J, Zhang C. Detection of cervical cells based on improved SSD network. *Multimedia Tools Appl* 2022;81(10):13371–87. <http://dx.doi.org/10.1007/s11042-021-11015-7>.
- [13] Wang Y-K, Syu H-Y, Chen Y-H, Chung C-S, Tseng YS, Ho S-Y, Huang C-W, Wu I-C, Wang H-C. Endoscopic images by a single-shot multibox detector for the identification of early cancerous lesions in the esophagus: A pilot study. *Cancers* 2021;13(2):321. <http://dx.doi.org/10.3390/cancers13020321>.
- [14] Wei J, As'arry A, Rezali KAM, Yusoff MZM, Ma H, Zhang K. A review of YOLO algorithm and its applications in autonomous driving object detection. *IEEE Access* 2025. <http://dx.doi.org/10.1109/ACCESS.2025.3573376>.
- [15] Topuz Y, Yildiz S, Varli S. Performance analysis of the YOLO series for object detection: Detection of mitosis cells in histopathology images. In: 2023 Medical technologies congress. IEEE; 2023, p. 1–4. <http://dx.doi.org/10.1109/TIPTEKNO59875.2023.10359203>.
- [16] Ghafari M, Mailman D, Hatami P, Peyton T, Yang L, Dang W, Qin H. A comparison of YOLO and mask-RCNN for detecting cells from microfluidic images. In: 2022 International conference on artificial intelligence in information and communication. IEEE; 2022, p. 204–9. <http://dx.doi.org/10.1109/ICAII54071.2022.9722616>.
- [17] Abou Ali M, Abdulfattah M, Al Hussein B, Dornaika F, Cherry A, Hajj-Hassan M, Hamawy L. Comprehensive benchmarking of YOLOv11 architectures for scalable and granular peripheral blood cell detection. 2025. <http://dx.doi.org/10.48550/arXiv.2509.24595>, arXiv preprint [arXiv:2509.24595](https://arxiv.org/abs/2509.24595).
- [18] Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. End-to-end object detection with transformers. In: Proceedings of the European conference on computer vision. Springer; 2020, p. 213–29. http://dx.doi.org/10.1007/978-3-030-58452-8_13.
- [19] Prangemeier T, Reich C, Koepl H. Attention-based transformers for instance segmentation of cells in microstructures. In: 2020 IEEE international conference on bioinformatics and biomedicine. IEEE; 2020, p. 700–7. <http://dx.doi.org/10.1109/BIBM49941.2020.9313305>.
- [20] Zhao Y, Lv W, Xu S, Wei J, Wang G, Dang Q, Liu Y, Chen J. DETRs beat YOLOs on real-time object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2024, p. 16965–74. <http://dx.doi.org/10.1109/CVPR52733.2024.01605>.

- [21] He W, Zhang Y, Xu T, An T, Liang Y, Zhang B. Object detection for medical image analysis: Insights from the RT-DETR model. In: Proceedings of the 2025 international conference on artificial intelligence and computational intelligence. 2025, p. 415–20. <http://dx.doi.org/10.1145/3730436.3730506>.
- [22] Overwijk WW, Restifo NP. B16 as a mouse model for human melanoma. *Curr Protoc Immunol* 2001;39(1):20.1.1–29. <http://dx.doi.org/10.1002/0471142735.im2001s39>.
- [23] Al-Hamadani MNA, Szeman-Nagy G, Hajdu A, Harangi B. B16BL6: Microscopy dataset and annotations (YOLO format). 2026, Open Science Framework (OSF) dataset. <https://doi.org/10.17605/OSF.IO/5J276>.
- [24] Al-Hamadani MNA, Fadhel MA, Alzubaidi L, Harangi B. Reinforcement learning algorithms and applications in healthcare and robotics: A comprehensive and systematic review. *Sensors* 2024;24(8):2461. <http://dx.doi.org/10.3390/s24082461>.
- [25] Zunair H, Khan S, Hamza AB. RSUD20K: A dataset for road scene understanding in autonomous driving. In: 2024 IEEE international conference on image processing. Abu Dhabi, United Arab Emirates; 2024, p. 708–14. <http://dx.doi.org/10.1109/ICIP51287.2024.10648203>.
- [26] Caicedo JC, Goodman A, Karhohs KW, Cimini BA, Ackerman J, Haghighi M, Heng C, Becker T, Doan M, McQuin C, Rohban M, Singh S, Carpenter AE. Nucleus segmentation across imaging experiments: The 2018 data science bowl. *Nature Methods* 2019;16(12):1247–53. <http://dx.doi.org/10.1038/s41592-019-0612-7>.
- [27] Ulman V, Maška M, Magnusson KEG, Ronneberger O, Haubold C, Harder N, Matula P, Matula P, Svoboda D, Radojevic M, Smal I, Rohr K, Jaldén J, Blau HM, Dzyubachyk O, Lelieveldt B, Xiao P, Li Y, Cho S-Y, Dufour AC, Olivo-Marin J-C, Reyes-Aldasoro CC, Solis-Lemus JA, Bensch R, Brox T, Stegmaier J, Mikut R, Wolf S, Hamprecht FA, Esteves T, Quelhas P, Demirel Ö, Malmström L, Jug F, Tomancak P, Meijering E, Muñoz-Barrutia A, Kozubek M, Ortiz-de Solorzano C. An objective comparison of cell-tracking algorithms. *Nature Methods* 2017;14(12):1141–52. <http://dx.doi.org/10.1038/nmeth.4473>.
- [28] Edlund C, Jackson TR, Khalid N, Bevan N, Dale T, Dengel A, Ahmed S, Trygg J, Sjögren R. LIVECell—A large-scale dataset for label-free live cell segmentation. *Nature Methods* 2021;18(9):1038–45. <http://dx.doi.org/10.1038/s41592-021-01249-6>.
- [29] Ma J, Xie R, Ayyadhury S, Ge C, Gupta A, Gupta R, Gu S, Zhang Y, Lee G, Kim J, Lou W, Li H, Upschulte E, Dickscheid T, de Almeida JG, Wang Y, Han L, Yang X, Labagnara M, Gligorovski V, Scheder M, Rahi SJ, Kempster C, Pollitt A, Espinosa L, Mignot T, Middeke JM, Eckardt J-N, Li W, Li Z, Cai X, Bai B, Greenwald NF, Van Valen D, Weisbart E, Cimini BA, Cheung T, Brück O, Bader GD, Wang B. The multimodality cell segmentation challenge: Toward universal solutions. *Nature Methods* 2024;21(6):1103–13. <http://dx.doi.org/10.1038/s41592-024-02233-6>.
- [30] Schwendy M, Unger RE, Parekh SH. EVICAN—A balanced dataset for algorithm development in cell and nucleus segmentation. *Bioinformatics* 2020;36(12):3863–70. <http://dx.doi.org/10.1093/bioinformatics/btaa225>.
- [31] Jocher G, Chaurasia A, Qiu J. Ultralytics YOLOv8. 2023, <https://github.com/ultralytics/ultralytics>. [Accessed: 05 September 2023].
- [32] Sohan M, Sai Ram T, Rami Reddy CV. A review on YOLOv8 and its advancements. In: International conference on data intelligence and cognitive informatics. Singapore: Springer; 2024, p. 529–45. http://dx.doi.org/10.1007/978-981-99-7962-2_39.
- [33] Terven J, Córdova-Esparza D-M, Romero-González J-A. A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Mach Learn Knowl Extr* 2023;5(4):1680–716. <http://dx.doi.org/10.3390/make5040083>.
- [34] Jocher G, Chaurasia A, Qiu J. Ultralytics YOLOv11. 2024, <https://github.com/ultralytics/ultralytics>. [Accessed: 12 July 2025].
- [35] Khanam R, Hussain M. YOLOv11: An overview of the key architectural enhancements. 2024, <http://dx.doi.org/10.48550/arXiv.2410.17725>, arXiv preprint [arXiv:2410.17725](https://arxiv.org/abs/2410.17725).
- [36] Khan AT, Jensen SM. LEAF-Net: A unified framework for leaf extraction and analysis in multi-crop phenotyping using YOLOv11. *Agriculture* 2025;15(2):196. <http://dx.doi.org/10.3390/agriculture15020196>.
- [37] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg AC. SSD: Single shot MultiBox detector. In: Proceedings of the European conference on computer vision. Springer; 2016, p. 21–37. http://dx.doi.org/10.1007/978-3-319-46448-0_2.
- [38] Maity M, Banerjee S, Chaudhuri SS. Faster R-CNN and YOLO based vehicle detection: A survey. In: 2021 5th international conference on computing methodologies and communication. IEEE; 2021, p. 1442–7. <http://dx.doi.org/10.1109/ICCMCS1019.2021.9418274>.
- [39] Yang S, Fang B, Tang W, Wu X, Qian J, Yang W. Faster R-CNN based microscopic cell detection. In: 2017 International conference on security, pattern analysis, and cybernetics. Shenzhen, China: IEEE; 2017, p. 345–50. <http://dx.doi.org/10.1109/SPAC.2017.8304302>.
- [40] Liu L, Zhou J, Zhang B, Dai S, Shen M. Visual detection on posture transformation characteristics of sows in late gestation based on libra R-CNN. *Biosyst Eng* 2022;223:219–31. <http://dx.doi.org/10.1016/j.biosystemseng.2022.09.003>.
- [41] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Köpf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S. PyTorch: An imperative style, high-performance deep learning library. In: *Advances in neural information processing systems*, vol. 32, 2019.