Detection and Classification of Blood Cells

Group 30 - Yiwei Gao, Li He, Ruijia Peng, Yiting Zhang, Xiaoai Zhu

View on GitHub

Introduction/Background

Automated blood cell detection and classification are crucial for medical diagnostics, particularly for conditions like anemia, thalassemia, and sickle cell disease [1]. This project will leverage machine learning algorithms to detect, classify, and count red blood cells (RBCs) in microscopic blood smear images. We aim to use automation to enhance speed, consistency, and accuracy in diagnostics, reducing human error and providing real-time results, which are critical in emergency scenarios. The solution is designed to be scalable and cost-effective, improving healthcare accessibility.

Previous studies have demonstrated the potential of machine learning and deep learning methods in automating blood cell detection and classification. For instance, Katar and Yildirim [1] employed Vision Transformer models to classify and localize white blood cells, achieving high accuracy and interpretability through explainable AI methods. Similarly, Bu et al. [2] introduced TW-YOLO, an innovative detection model for blood cells based on multi-scale feature fusion, which significantly improved detection performance by leveraging multi-scale features for better accuracy and robustness.

Lu et al. [3] utilized a Shifted Window Vision Transformer (Swin Transformer) to classify blood cells, demonstrating the advantages of self-attention mechanisms in capturing intricate details in cell morphology. Banik et al. [4] presented an automatic segmentation and classification approach for white blood cells using convolutional neural networks (CNNs), highlighting the importance of accurate segmentation as a prerequisite for effective classification.

This project builds on these prior works by utilizing YOLOv8 for real-time RBC detection, Vision Transformer (ViT) for classification of RBC abnormalities, and K-Means clustering for unsupervised segmentation. By combining supervised and unsupervised techniques, we aim to provide a comprehensive solution for automated blood cell analysis that can be applied to various clinical scenarios.

The dataset for this project is sourced from Blood Cell Detection Dataset, which includes annotated images of blood cells. The dataset contains RBCs, WBCs, and platelets, each labeled and categorized, making it suitable for both supervised and unsupervised machine learning techniques.

Dataset Link: https://www.kaggle.com/datasets/adhoppin/blood-cell-detection-datatset/data

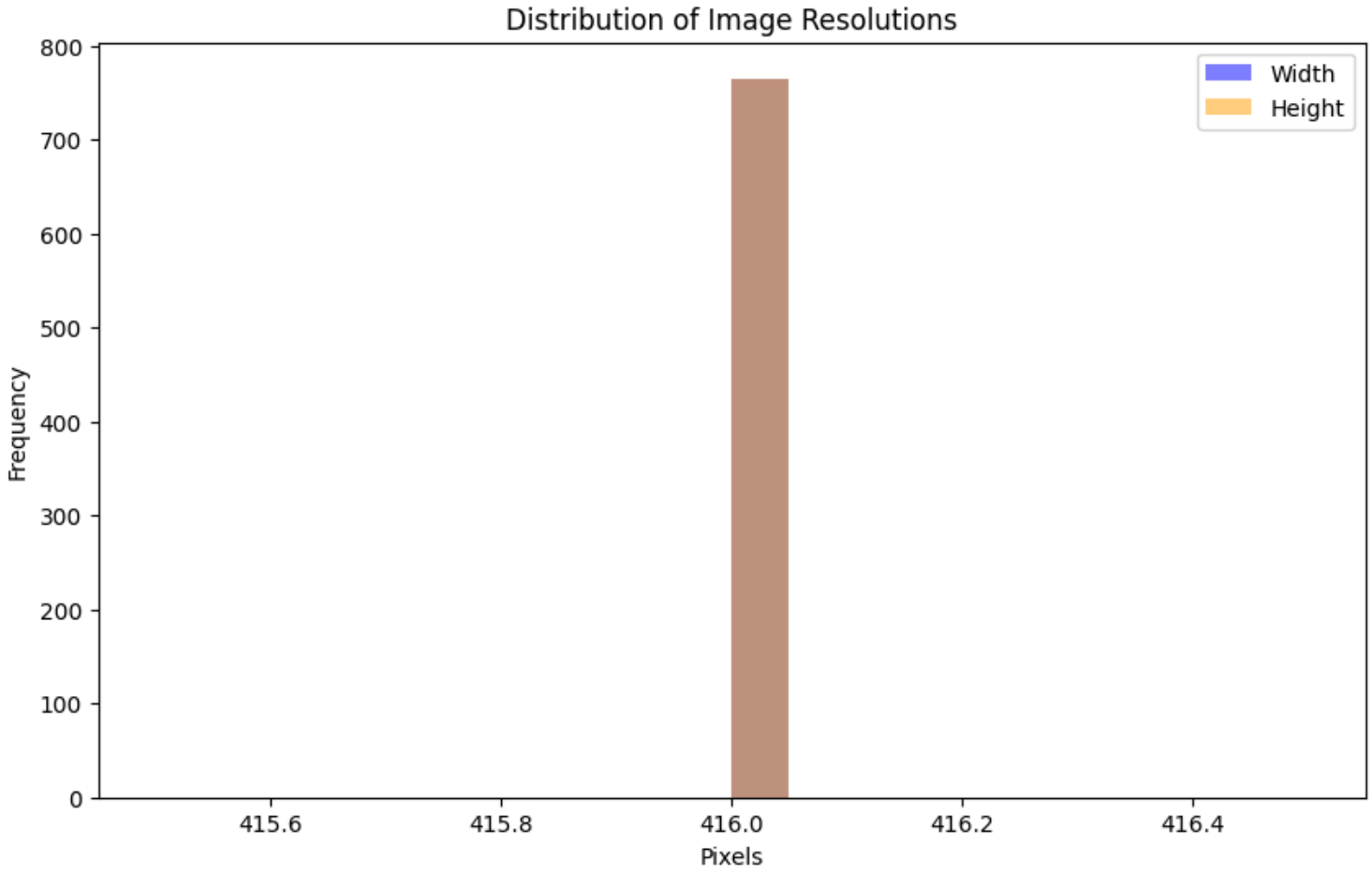
Problem Definition

Manually counting and identifying blood cells is time-consuming, subjective, and inconsistent, especially in busy clinical settings, affecting timely diagnosis for conditions like anemia or malaria. The motivation is to automate RBC detection and classification using machine learning, improving speed, accuracy, and consistency. This will reduce healthcare professionals' workload, minimize diagnostic errors, and enhance patient outcomes, especially in resource-limited regions.

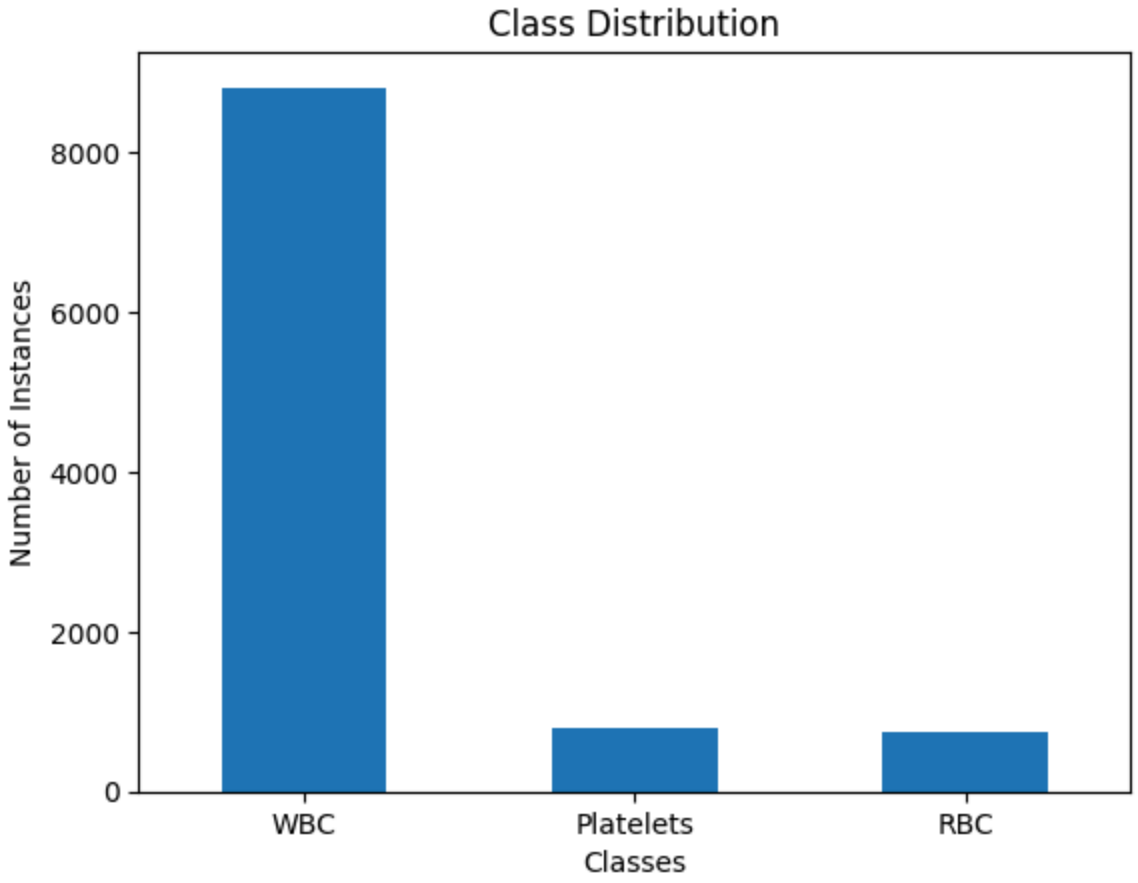
Methods

Data Preprocessing Methods:

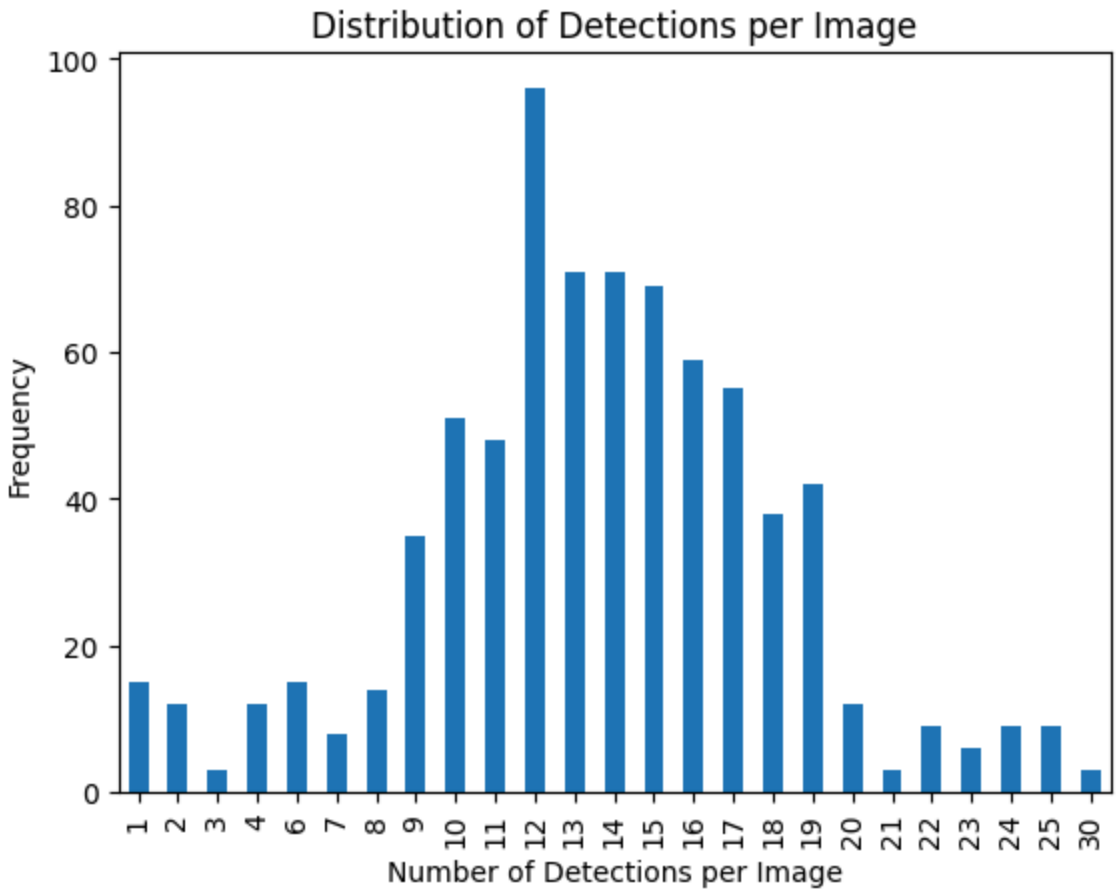
For the Image Resolution Analysis, we observed that all images have a uniform resolution, which is beneficial as it facilitates consistent input sizes, optimizing model performance and memory usage. This consistency in resolution contributes to better model generalization.

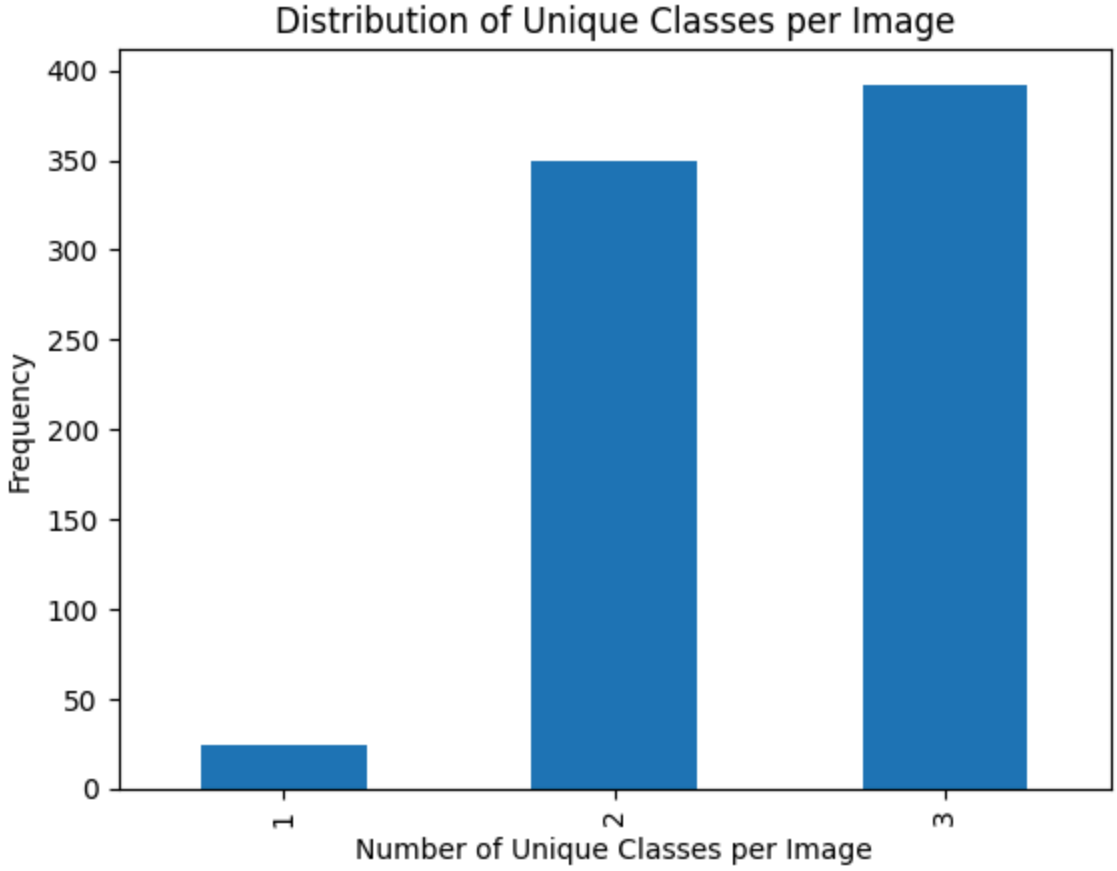


In the Class Distribution Analysis, we identified that most instances belong to the WBC class, while platelets and RBC classes have similar counts. This analysis allows us to detect any class imbalance, which is essential to prevent the model from developing biases toward more common classes, ensuring it performs well across all categories.

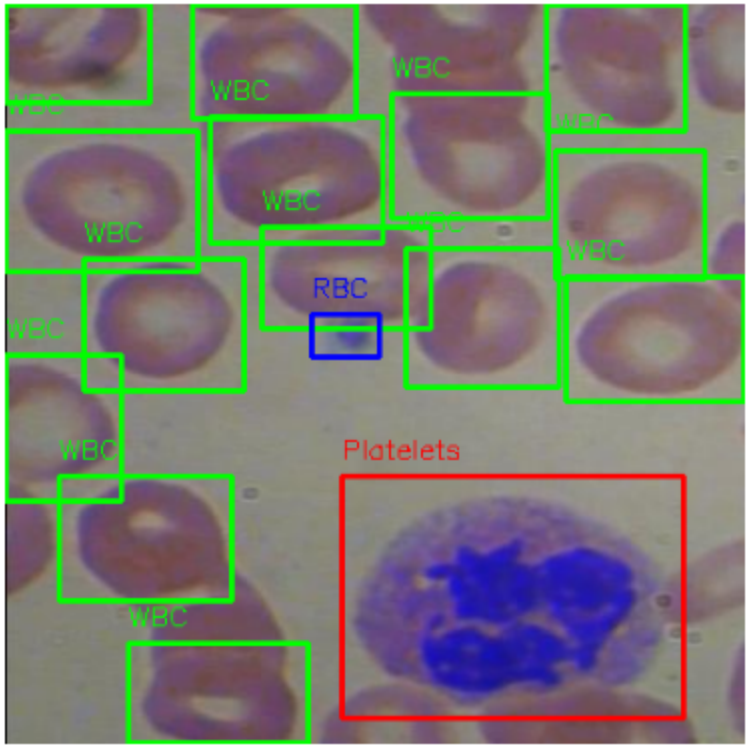


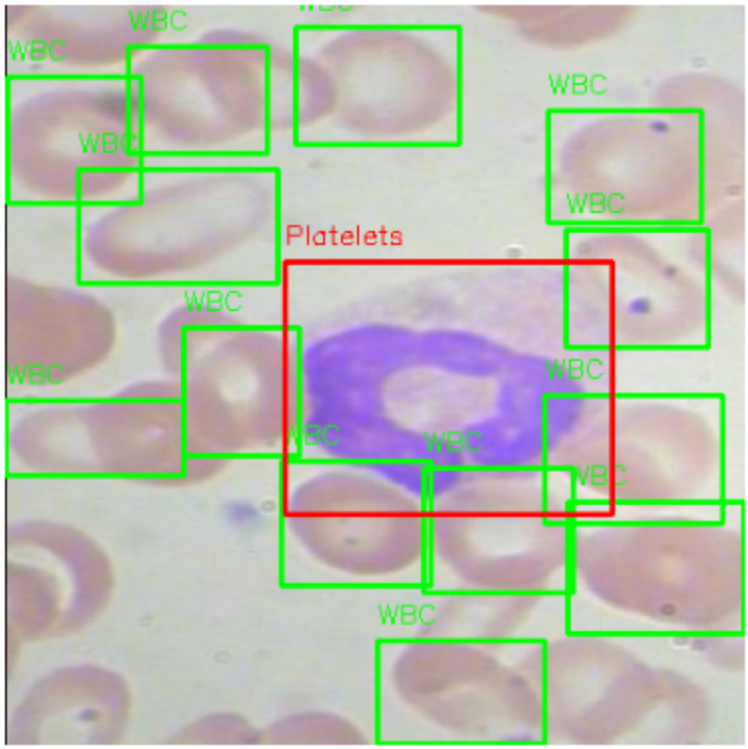
In terms of Number of Detections per Image, we found that the majority of images contain between 10 to 20 detections, with most images featuring 2 or 3 unique classes. This analysis helps us assess the complexity of the task, as images with more objects or varied classes can require additional computational resources and fine-tuning of the model.


Distribution of Detections per Image
Frequency
0
20
40
60
80
100
1
2
3
4
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
30
Number of Detections per Image


Distribution of Unique Classes per Image
Frequency
0
50
100
150
200
250
300
350
400
1
2
3
Number of Unique Classes per Image

Lastly, the Bounding Box Visualization provided a way to verify annotation quality, ensuring that the bounding boxes are accurate. This step helps us identify potential annotation errors, which is crucial for building a reliable dataset that will support effective model training.





- **Image Augmentation:** The dataset is augmented using transformations like rotation, zoom, and flipping to increase diversity and reduce overfitting. This is implemented using ImageDataGenerator from Keras.

- **Normalization:** By using MinMaxScaler from scikit-learn, images are normalized to ensure pixel intensity values are scaled between 0 and 1, improving model performance.
- **Feature Extraction (for K-Means):** Features like pixel intensities, shape descriptors, and texture properties are extracted from images using OpenCV. These features are then used as input vectors for K-Means clustering.

Machine Learning Algorithms:

- **YOLOv8 (Supervised):** YOLOv8 is used to detect and localize RBCs in real-time within microscopic images with bounding boxes [2]. It is selected due to its high speed and accuracy, which are essential for processing large medical images effectively. The model is implemented using the YOLOv8 framework from Ultralytics, where pre-trained weights are fine-tuned specifically on the RBC dataset to enhance detection performance.
- **Vision Transformer (ViT) (Supervised):** The ViT model is used to classify RBCs into categories such as normal, deformed, or infected [3]. It leverages its self-attention mechanism to capture subtle differences in cell morphology, which are crucial for accurate diagnosis of various blood disorders. The implementation involves loading a pre-trained ViT model from the timm library and fine-tuning it for RBC classification, allowing it to effectively distinguish between different types of cell abnormalities [4].
- **K-Means Clustering (Unsupervised):** K-Means clustering is used for unsupervised segmentation of RBCs to group similar cells based on features like pixel intensity, size, and shape. By clustering cells into distinct groups, K-Means provides a foundational analysis that helps differentiate normal cells from abnormal ones for further study. The implementation uses the KMeans() function from scikit-learn, with the number of clusters specified based on expected RBC types, such as normal and various abnormalities.

Results and Discussion

The project targets high accuracy in RBC detection and classification, aiming for an overall accuracy of over 90% to ensure reliability. Precision and recall metrics are set at over 85% to balance minimizing false positives and maximizing true positives, with YOLOv8's bounding boxes aiming for an average Intersection over Union (IoU) of at least 0.75. The goal is to develop a scalable solution for clinical use, emphasizing accuracy, reliability, and robust performance to support critical medical decision-making.

YOLOv8 with SE (Squeeze-and-Excitation) Blocks:

- **Final mAP50:** 0.9102
- **Final mAP50-95:** 0.6368
- **Final Precision:** 0.8846

- **Final Recall:** 0.8767

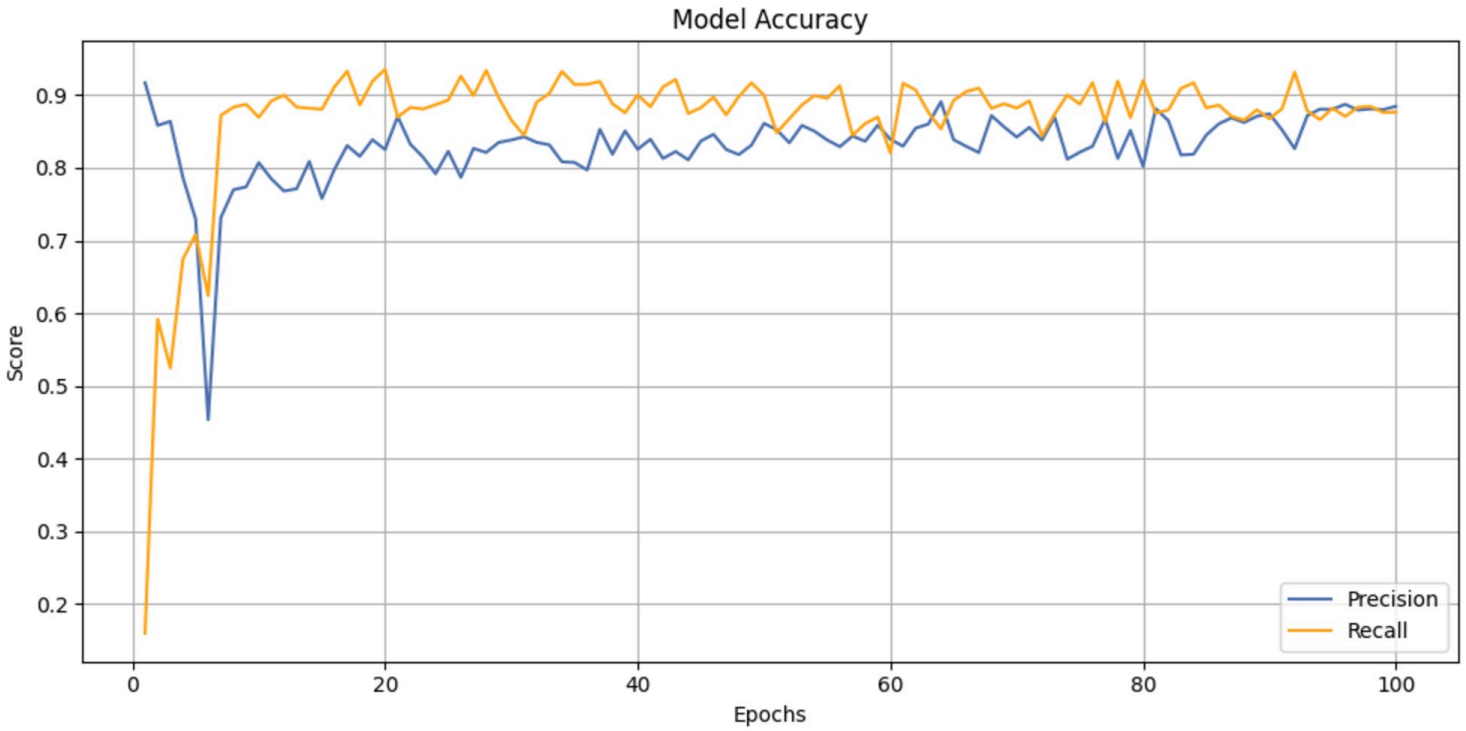
YOLOv8 shows consistent and robust training behavior, with precision at ~0.88 and recall at ~0.87, both showing gradual improvements through the early epochs before achieving consistent performance. The mAP50 metric reaches 0.91, demonstrating excellent cell detection at standard IoU thresholds, while mAP50-95 (~0.63) indicates good but slightly less effective performance with stricter bounding box localization. Training losses for YOLOv8 decrease smoothly and converge rapidly, with minimal fluctuations, reflecting an efficient optimization process. The visualizations highlight the model's stability and its capability to maintain consistent performance across multiple metrics, indicating its suitability for detecting and localizing cells with precision.

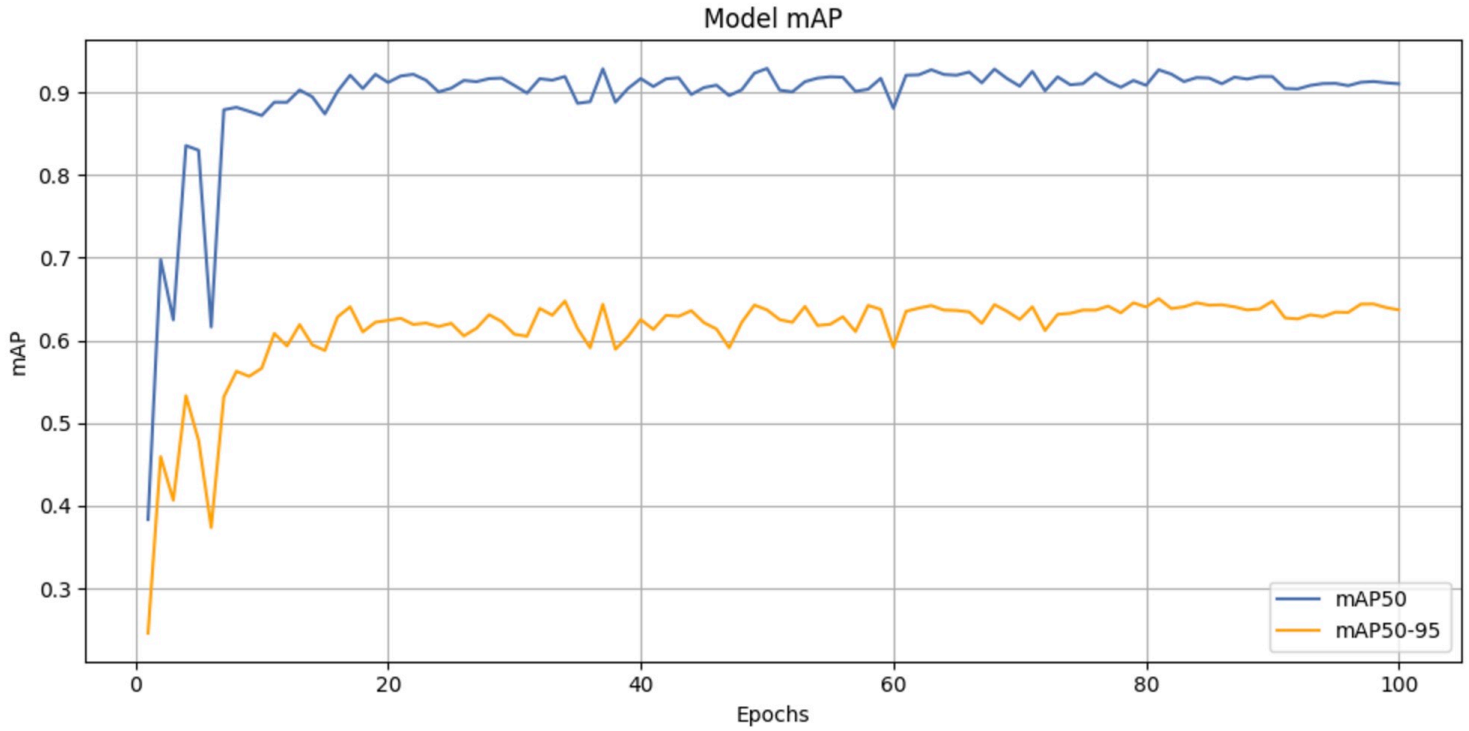
Strength:

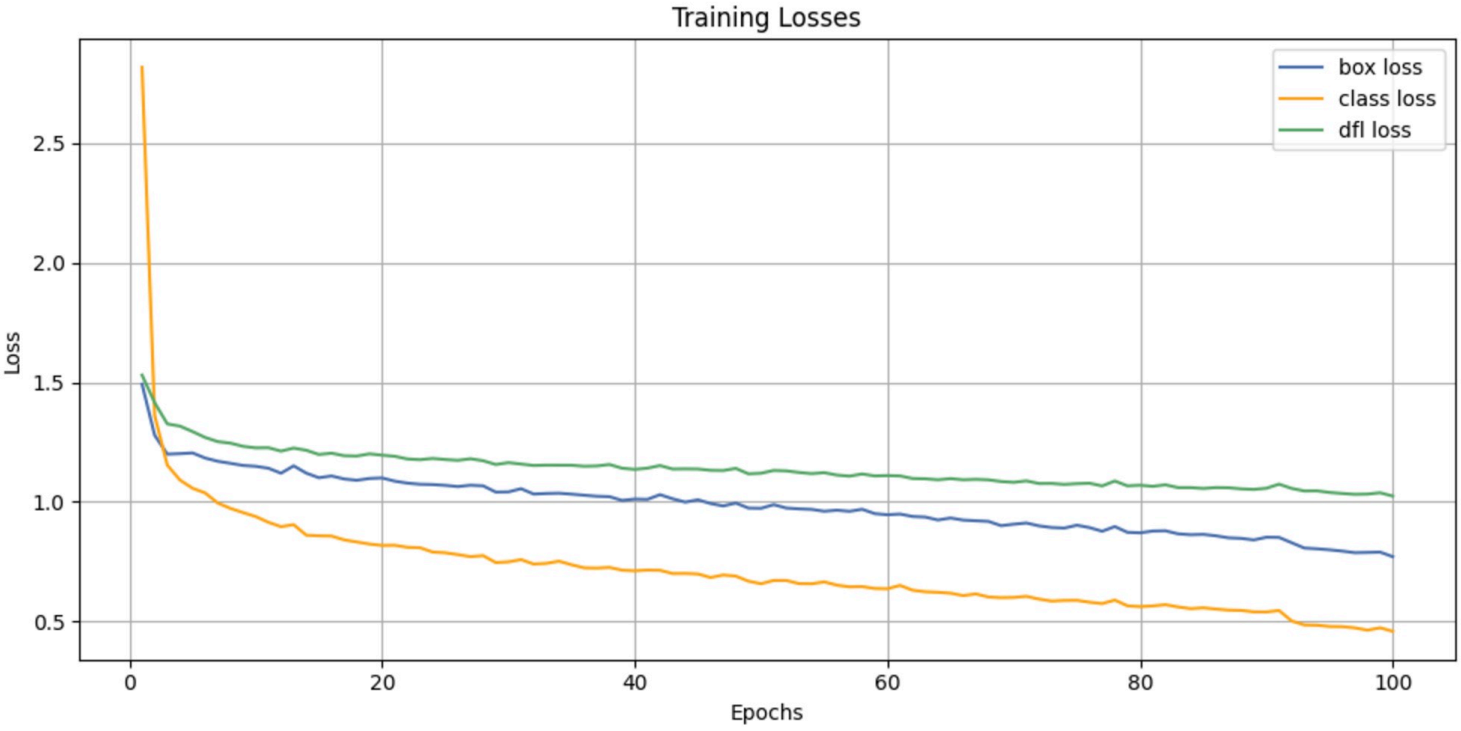
- Balanced performance across precision and recall makes it highly robust for real-world use cases.
- Superior in minimizing false positives compared to ViT, evidenced by higher precision.

Limitations:

- Slightly lower recall suggests potential under-detection of true positives.
- Computationally intensive compared to simpler methods like K-Means.







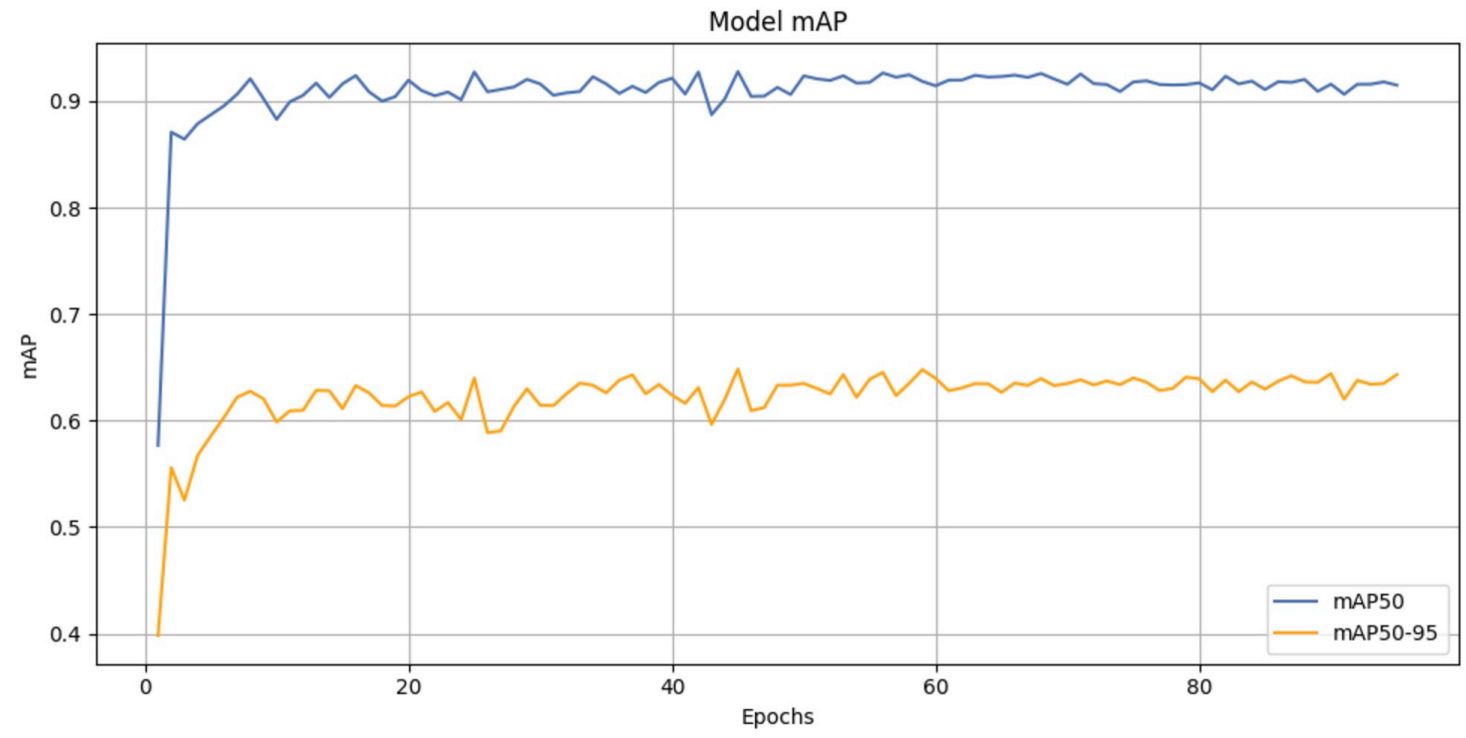
ViT (Vision Transformer):

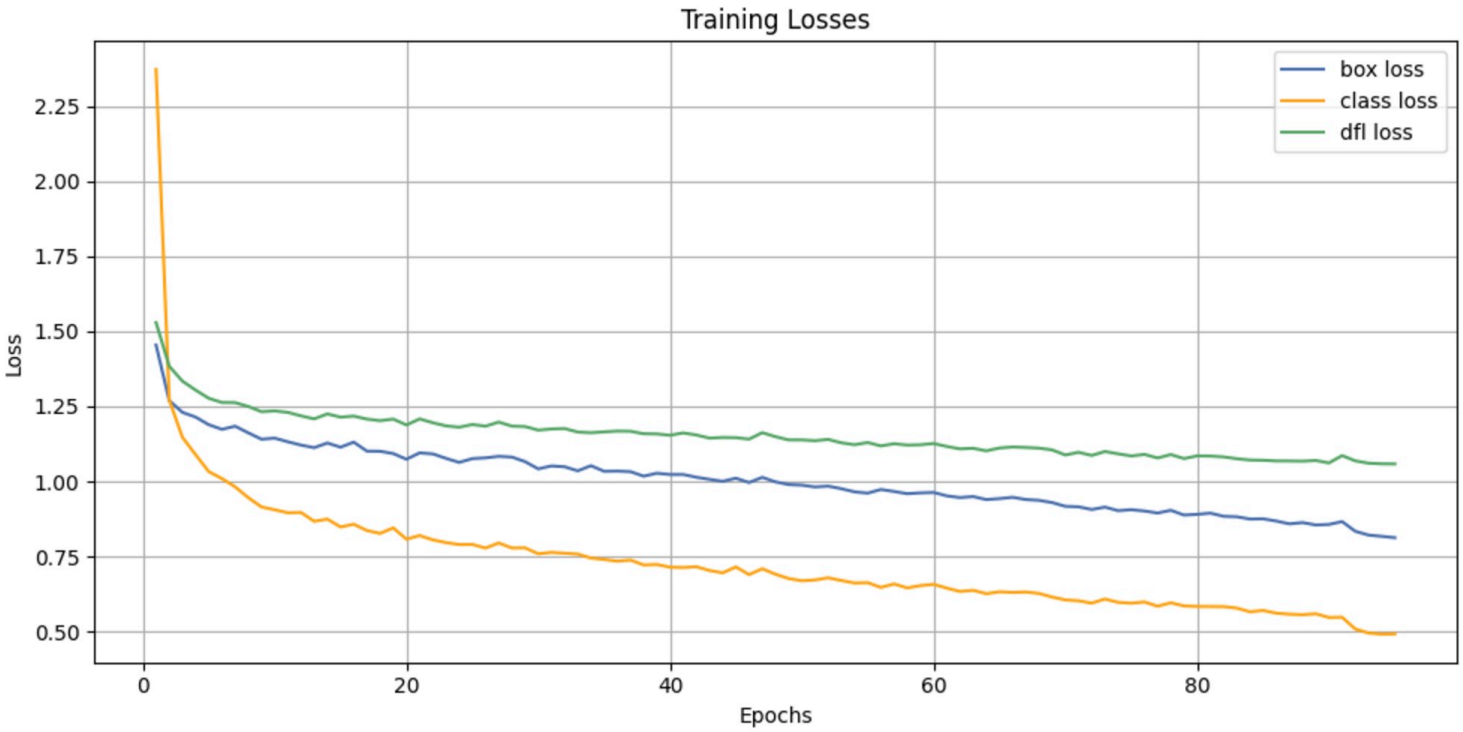
- **Final mAP50:** 0.9150
- **Final mAP50-95:** 0.6433
- **Final Precision:** 0.8438
- **Final Recall:** 0.9175

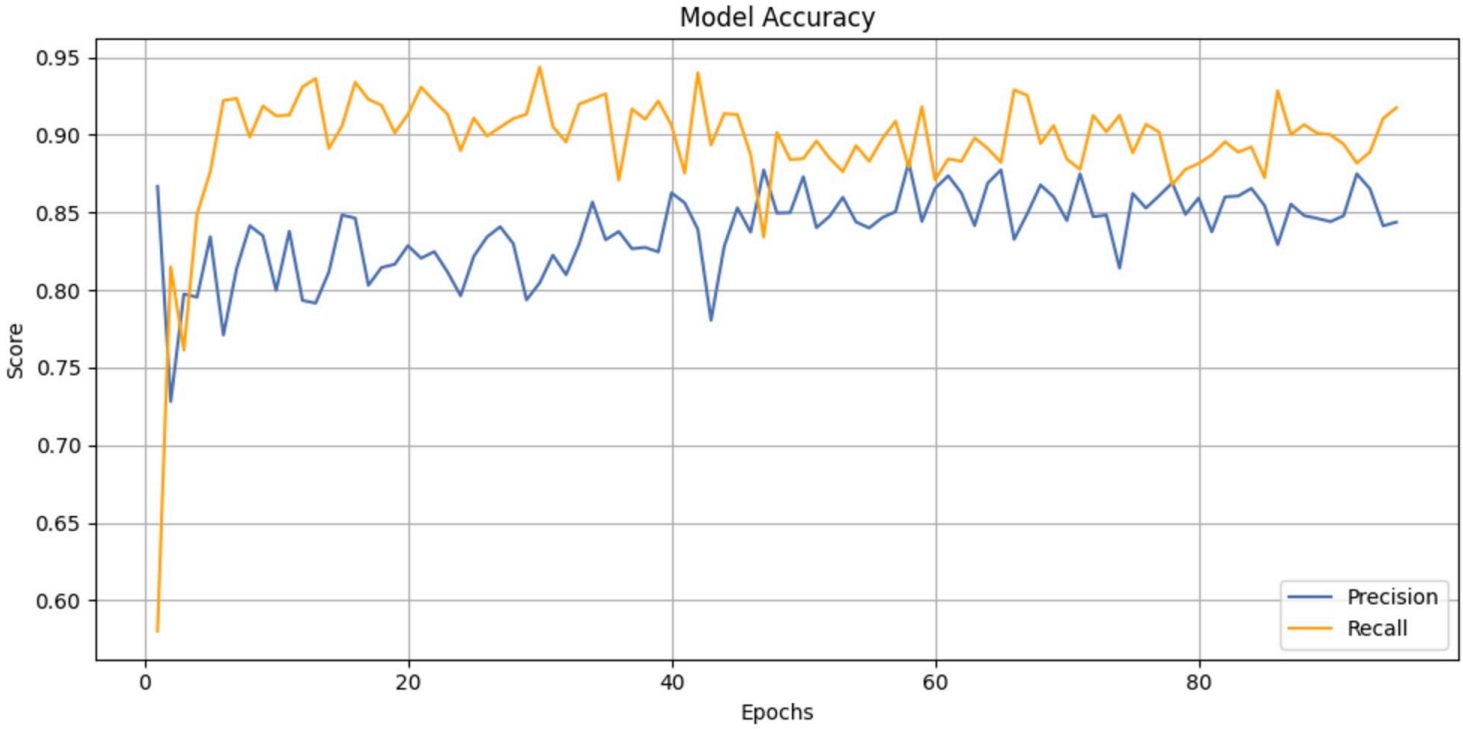
The hybrid ViT-YOLOv8 model demonstrates robust performance, combining the global feature extraction of the Vision Transformer with YOLOv8's detection capabilities. It achieves a high recall (~0.9175), stabilizing early in training, reflecting its ability to detect nearly all true positives. Precision reaches ~0.8438, indicating the model effectively reduces false positives while maintaining consistent performance. The mAP50 metric stabilizes at 0.9150, showing excellent detection accuracy at standard IoU thresholds, while the mAP50-95 (~0.6433) indicates solid but slightly reduced performance under stricter bounding box localization requirements. Training losses decrease steadily throughout training, with minimal fluctuations, highlighting effective optimization and model convergence. The integration of ViT's self-attention mechanisms enhances feature representation, while YOLOv8's architecture ensures efficient and reliable detection.

Strength:

- Excellent recall ensures nearly all relevant red blood cells are detected.
- Transformer-based architecture enables modeling of global dependencies, crucial for classifying cell abnormalities. **Limitations:**
- Lower mAP50-95 compared to mAP50 suggests slight difficulty in accurately localizing bounding boxes at varying thresholds.
- Precision lags slightly compared to YOLOv8, indicating room for improvement in minimizing false positives.







K-Means:

- **Final mAP50:** 0.2732
- **Final mAP50-95:** 0.0424
- **Final Precision:** 0.3457
- **Final Recall:** 0.3540 The K-Means method demonstrates significantly lower performance compared to supervised learning models. The mAP50 is 0.2732, indicating limited capability in detecting cells at a standard IoU threshold, while the mAP50-95 is extremely low at 0.0424,

reflecting its inability to perform well under stricter localization requirements. Precision is 0.3457, highlighting a high rate of false positives, and recall is 0.3540, showing that the model struggles to detect a substantial proportion of true positives. These results indicate that K-Means, as an unsupervised clustering method, is not well-suited for complex tasks like cell detection. Its limitations likely stem from its inability to leverage labeled data or effectively account for the spatial relationships and features necessary for accurate detection and classification.

Strength:

- Easy to implement and provides clear cluster assignments, making it highly interpretable for understanding basic patterns in data.
- Efficient, requiring minimal computational resources.
- Works without labeled data, suitable for limited datasets.

Limitations:

- Reliance on simplistic clustering techniques leads to poor performance in metrics like precision, recall, and mAP.
- Cannot capture non-linear or intricate data relationships.
- Lacks contextual understanding, leading to low accuracy.

Conclusion:

The three methods—ViT, YOLOv8, and K-Means—show different strengths, with ViT and YOLOv8 outperforming K-Means significantly. ViT achieves the highest mAP50 (0.9150) and recall (0.9175), making it ideal for applications requiring maximum detection accuracy and exhaustive identification of true positives. On the other hand, YOLOv8 delivers the highest precision (0.8846) and offers balanced performance with faster convergence and efficient optimization, making it well-suited for real-time or speed-critical applications. K-Means, while simple, fast, and interpretable, is unsuitable for complex tasks like object detection, given its significantly lower mAP50 (0.2732) and poor ability to capture spatial and contextual patterns. Ultimately, ViT is recommended for tasks prioritizing detection accuracy and recall, while YOLOv8 is preferred for scenarios requiring a balance of precision, recall, and speed. K-Means, due to its limited capabilities, is not recommended for this task.

Future Work:

- **Improve the Hybrid Model:** Work on enhancing the ViT-YOLOv8 hybrid model by improving feature combination and optimizing resource usage.
- **Test on More Datasets:** Evaluate the models on diverse datasets to ensure robustness and generalizability.

- **Combine Models:** Explore ensemble techniques to combine the strengths of ViT, YOLOv8, and K-Means.
- **Explainable AI and Visualization Tools:** Develop visualization tools to improve interpretability and trustworthiness of the models.

Gantt Chart

Here is a link to the Gantt Chart.

Contribution Table

Name	Project Contributions
Yiwei Gao	Data Sourcing and Cleaning, M1 Results Analysis, M2 Results Evaluation
Ruijia Peng	Data Sourcing and Cleaning, Model Coding, Results Evaluation and Analysis, Training Pipeline Setup, M2 Multi-Head Attention Implementation
Yiting Zhang	Data Pre-Processing, Presentation, Model Comparison, Results Evaluation and Analysis
Xiaoai Zhu	Model Selection, Recording, Model Comparison, M2 Transformer Block Implementation
Li He	Model Selection, Results Evaluation and Analysis, Midterm Report, Model Comparison
All	Final Report, Results Evaluation, Model Comparison

Video Presentation

Here is a link to the Video.

References

[1] O. Katar and O. Yildirim, “Explainable Vision Transformer Model Based White Blood Cells Classification and Localization,” *Diagnostics*, vol. 13, no. 14, 2023, doi: 10.3390/diagnostics13142459

[2] A. Bu et al., “TW-YOLO: An Innovative Blood Cell Detection Model Based on Multi-Scale Feature Fusion,” *Sensors*, vol. 24, no. 19, 2024, doi: 10.3390/s24196168

[3] S. Lu, S. Wang, and Y. Zhang, “Shifted Window Vision Transformer for Blood Cell Classification,” *Electronics*, vol. 12, no. 11, 2023, doi: 10.3390/electronics12112442.

[4] A. Banik, R. Saha, and K. Kim, “Automatic nucleus segmentation and CNN model based classification method of white blood cell,” *Expert Systems with Applications*, vol. 149, 2020, doi: 10.1016/j.eswa.2020.113211.
