Image processing and machine learning techniques for Chagas disease detection and identification

No Author Given

No Institute Given

Abstract. Chagas disease, caused by the Trypanosoma cruzi parasite, poses a significant health threat, particularly in Latin America, with millions affected globally. This research introduces a novel approach using deep learning techniques for the automated detection of Trypanosoma cruzi in blood smear images provided by Zoonoses Laboratory (CIR) in Mexico. Advanced deep learning architectures like Faster RCNN, RetinaNet, YOLOv8, and FCOS have been adapted, trained, and compared with each other in terms of the detection accuracy of each image. Our selection of those models is based on their ability to swiftly and accurately detect anomalies, measured through rigorous assessment using pivotal metrics like Mean Average Precision (mAP) across varying Intersection over Union (IoU) thresholds. Notably, the YOLOv8 model has showcased outstanding performance, boasting a remarkable mAP score of 0.951 for parasite detection and localisation. Specifically, YOLOv8 outperforms with a leading mAP of 0.951 at 50% IoU and maintains commendable precision with a score of 0.594 for IoU thresholds ranging from 50% to 95%. This research reduces dependence on skilled manual analysis holding a significant implications for healthcare in Chagas-affected regions by providing a rapid, automated solution to disease detection. This work has the potential to revolutionise diagnostics in resource-limited settings. Moreover, the models' adaptability to other parasitic infections enhances their global health impact.

Keywords: Chagas Parasite · Object Detection · Deep learning.

1 Introduction

Chagas disease, or American trypanosomiasis, caused by Trypanosoma Cruzi (T. cruzi), is a major health concern, especially in Latin America, affecting millions [1]. Chagas disease has different phases. Having a direct examination of blood smears for the acute phase would offer several benefits, such as: facilitating timely intervention and treatment, including the use of anti-parasitic medications and monitoring of treatment effectiveness, and preventive measures for Chagas disease by informing public health care. Therefore, automatic image analysis techniques for this phase are required to avoid the laborious task for specialists. Moreover, the proposed techniques should work within reasonable financial costs without the need for a specific high-resolution optical microscope.

This research explores the use of deep learning for detecting T. cruzi in blood smear samples from infection in mice smear samples, Fig. 1, meticulously prepared at the Zoonoses Laboratory, part of Centro de Investigaciones Regionales (CIR) in Mexico.

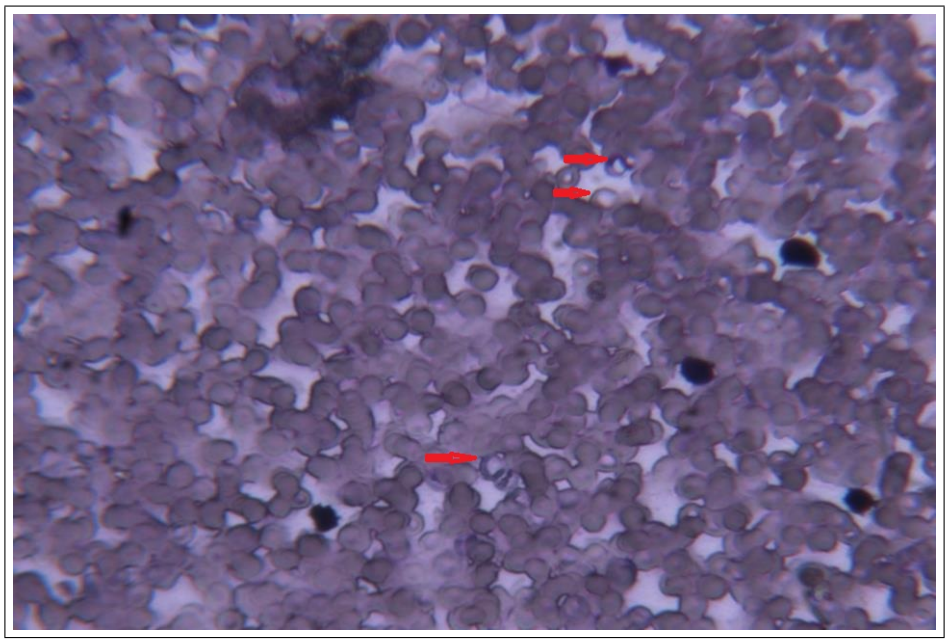

Fig. 1. Original image with Trypanosoma Cruzi presence.

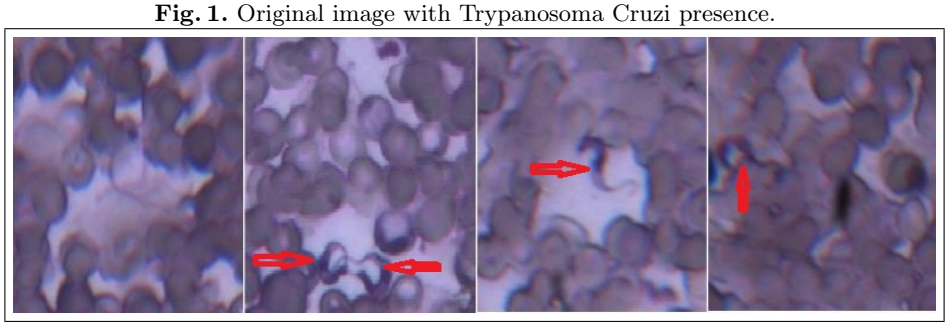

Fig. 2. Patches of images from Fig. 1 where the parasite are present.

Recently, there has been a surge in applying machine learning for the detection and classification of various diseases. These methods typically require a considerable amount of data for effective implementation of supervised learning techniques. Specialists in the field often prepare this data, labelling a large number of images to train algorithms like k-nearest neighbors [2], SVM [3], neural networks [4], and AdaBoost [5]. These models excel in identifying new, unseen

images as either infected or not, utilising the feature information from the labelled data. This automation greatly enhances the speed and reliability of disease detection, while reducing effort and costs. However, in the context of Chagas disease [6], a critical global health concern, the development of machine learning models for its detection or segmentation remains limited.

For Chagas detection there has been early efforts in 2013 by Soberanis-Mukul et al. [7] utilised basic machine learning techniques to classify the disease with notable sensitivity and precision, albeit with limitations due to false positives. By 2015, Uc-Cetina et al. [8] enhanced detection accuracy through combined image processing and data augmentation techniques, demonstrating the potential of integrating multiple machine learning approaches. Gaussian Discriminant Analysis has been employing in order to extract green channels from images, known to provide more significant information about the parasite. This refines detection capabilities by emphasising colour channel analysis, achieving impressive true positive rates.

The shift towards deep learning models marked a new era in Chagas disease identification. A. da Silva Pereira et al.'s 2019 study on MobileNetV2 indicated high validation accuracy [9], though it faced challenges with over fitting. The development of a U-Net-based convolutional neural network in 2020 by Allan Ojeda-Pat et al. [10] showed promise with an F1 score of 80%, despite facing hurdles such as dataset imbalance and architectural simplicity. Diogo Silva et al. [11] adopting a random forest algorithm approach, achieving commendable precision and sensitivity, though limited by image resolution from mobile cameras. The latest, 2022 research by Allan Ojeda-Pat et al. [12] on the Res2Unet model, achieved notable results, signalling the ongoing evolution of machine learning methodologies in combating Chagas disease, with a strong focus on image-based datasets. This work is inspired from Heun's method in solving differential equations, this model amalgamated active contour loss with advanced residual connections, enhancing its segmentation capability. Trained on a dataset comprising 626 images and evaluated on 207, Res2Unet demonstrated remarkable performance metrics: an 84% Dice coefficient, 85% precision, and 82% recall, surpassing the efficacy of prior models such as U-Net and ResUnet. Nevertheless, the study faced certain constraints, notably a scarcity of positive instances in the dataset and a constrained test set size, which cast doubts on its generalisability to broader real-world scenarios.

Given the escalating challenges associated with accurate diagnostics and the growing utilisation of Deep Learning (DL) as a supportive instrument, it becomes increasingly imperative to delve into this potential avenue. Convolutional Neural Networks (CNNs) [13] have been extensively used in assisting several, disease a diagnosis and improving clinical decisions [14]. Such works encourage researchers on applying CNN based architecture or any other deep-learning-based architectures to effectively classify diseases (caused by a parasite or not) and give diagnosis in endemic or non-endemic areas.

Tools development that does not require specialised people is of crucial importance in non-endemic areas where the illness has risen as a result of migra-

tory movement [1] or in endemic areas as a straightforward help in blood donor triage [15]. Developing such tools for Chagas disease would be of great help in non-endemic and endemic areas.

In this paper, we propose and compare various deep learning models for solving the problem of T. cruzi detection. All the experiments of the proposed models work on low-resolution data and are easily adapted to new similar datasets. Table 1 compares four object detection models: RetinaNet, Faster-RCNN, FCOS, and YOLOv8, using two key performance metrics: mAP50 and mAP50-95. These metrics show the precision of object detection at different Intersection over Union (IoU) thresholds. YOLOv8 achieves the highest mAP50 score (0.951), indicating superior performance in accurately detecting Chagas parasites with a moderate overlap criterion. However, across all models, mAP50-95 scores are moderate, reflecting the challenge of accurate localization over a range of IoU thresholds. Despite this, YOLOv8 leads in this metric as well, demonstrating its robustness in accurate detection and localization of chagas parasites. We remark that such a good performance was obtained for low-resolution images where edges of the objects of interest may appear blurred.

The rest of this paper is structured as follows: Section 2 details the type of the data and the methods used for detection comparison. Section 3 details the experimental results of the comparative architectures.

2 Material and Methods

2.1 Analysed Samples and Used Tools

In this study we used blood smear images prepared at the Zoonoses Laboratory, a crucial division of the Centro de Investigaciones Regionales (CIR) in Mexico, taken from mice in laboratory in the acute phase of T. cruzi infection. Blood smear samples were collected and subsequently imaged or recorded using a Nikon Eclipse E600 optical microscope. All images were captured with an attached 75 dpi camera, each with dimensions of 2560×1920 pixels. From these images, patches measuring 512×512 pixels were extracted to construct the image database. Within this database, 576 regions of interest (ROIs) were identified as containing parasites, while 624 ROIs were deemed parasite-free, leaving some images unlabelled. A visual representation of this dataset is provided in Fig. 1. The compilation of these 512×512 resolution image patches, depicted in Fig. 2, was further divided into training, validation, and test sets.

All the models were trained and tested on a Windows 10 system with Intel Xeon, 2 : 3-GHz processor, 16 GB RAM, 1 TB HDD, a CUDA-enabled Nvidia graphical processing unit (GPU), Python with Keras and Tensorflow as the backend, and CUDA and cuDNN dependencies were installed for GPU acceleration.

2.2 RetinaNet

RetinaNet is a single-stage object detection model proposed by the Tsung-Yi Lin et al. [16]. It is designed to address the extreme foreground-background class

imbalance encountered during object detection training. The key innovations in RetinaNet are:

- **Focal Loss**: A novel loss function that focuses training on hard, misclassified examples by down-weighting the loss from easy, well-classified examples. This helps mitigate the class imbalance issue.
- **Feature Pyramid Network (FPN)**: A top-down architecture with lateral connections that combines low-resolution, semantically strong features with high-resolution, semantically weak features. This allows RetinaNet to efficiently detect objects at different scales.

RetinaNet Architecture The overall architecture of RetinaNet, as shown in Fig. 3, consists of a backbone network followed by two sub-networks: the Feature Pyramid Network (FPN) and the detection head. The backbone network extracts features at different scales from the input image. The FPN combines these multi-scale features in a top-down manner with lateral connections, enabling efficient detection of objects at different scales. The detection head comprises two sub-networks: the classification sub-network and the box regression sub-network. As depicted in (c) and (d) part of Fig. 3, the classification sub-network predicts the probability of an object being present at each spatial location, while the box regression sub-network simultaneously predicts the coordinates of the bounding boxes enclosing the objects. This seamless integration of the FPN and the detection head allows RetinaNet to achieve state-of-the-art performance in a single, unified network.

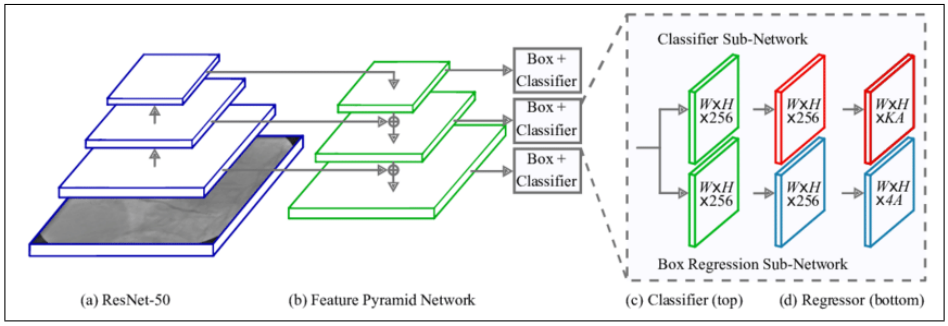

Fig. 3. RetinaNet Architecture.

RetinaNet eliminates the need for separate region proposal and feature resampling steps, making it a simple and efficient single-stage detector.

2.3 Faster R-CNN

Faster R-CNN is a two-stage object detection model proposed by Ross Girshick et al. [17]. It builds upon its predecessors, R-CNN [18] and Fast R-CNN [19], by introducing two key innovations:

- **Region Proposal Network (RPN)**: A dedicated fully convolutional network that efficiently generates region proposals (bounding boxes) that may contain objects. This replaces the slow selective search algorithm used in R-CNN.
- **End-to-end Training**: Unlike R-CNN which trained the region proposal and object detection components separately, Faster R-CNN trains the RPN and the Fast R-CNN object detection network jointly using a multi-task loss function.

Faster R-CNN Architecture As illustrated in the Fig. 4, the Faster R-CNN architecture consists of convolutional layers that extract feature maps from the input image. These feature maps are then fed into two sub-networks: the Region Proposal Network (RPN) and the Region-of-Interest (RoI) pooling layer. The RPN generates region proposals, which are then processed by the RoI pooling layer to extract region-specific features. These RoI features are finally passed through fully connected layers to produce the final class predictions and bounding box refinements. The RPN and the Fast R-CNN network share convolutional features, enabling nearly cost-free region proposals. This architecture allows Faster R-CNN to achieve state-of-the-art object detection accuracy while being several times faster than its predecessors. Despite its efficiency, the two-stage design can limit performance on small objects and predict redundant computations.

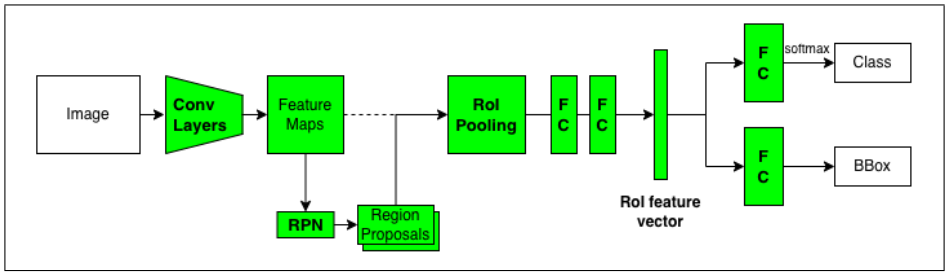

Fig. 4. Faster R-CNN Architecture.

2.4 FCOS (Fully Convolutional One-Stage Object Detection)

FCOS, proposed by Tian et al. [20], is a simple yet effective one-stage object detector that aims to solve the issues associated with the popular anchor-based detection methods. It introduces the following key ideas:

- **Fully Convolutional Design**: FCOS is a simple and anchor-free fully convolutional network that avoids the need for anchor boxes or region proposals. This reduces computational complexity and memory footprint.

– **Centreness**: A novel centreness branch is added to the standard classification and regression branches. This centerness branch aims to detect the centralised and complete objects, helping to address the issue of overlapping bounding box predictions.
– **Multi-level Prediction**: Like feature pyramid networks, FCOS makes predictions at multiple levels of the feature pyramid, allowing it to efficiently detect objects across various scales.

FCOS Architecture: The architecture of FCOS, as depicted in the Fig. 5, consists of a backbone network and a feature pyramid structure that extracts multi-scale features from the input image. The feature pyramid is constructed from different levels of the backbone's feature maps, with each level having a specific resolution.

For each level of the feature pyramid, FCOS employs a shared head module that performs three parallel tasks: classification, centreness prediction, and bounding box regression. The classification branch predicts the probability of an object being present at each spatial location, while the centreness branch aims to detect centralised and complete objects by estimating the "centreness" score. Simultaneously, the regression branch predicts the coordinates of the bounding boxes enclosing the objects.

By leveraging the multi-level predictions from the feature pyramid, FCOS can effectively detect objects at various scales. Its fully convolutional and anchor-free design allows for a simple and efficient architecture, making it well-suited for computationally constrained environments while achieving competitive performance compared to state-of-the-art detectors.

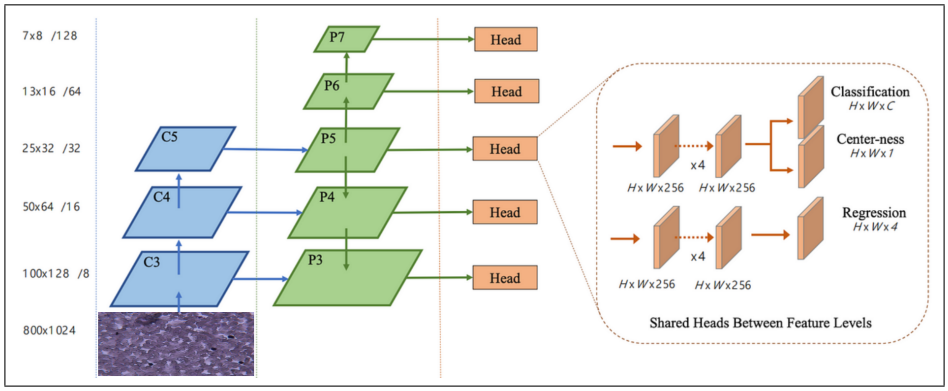

Fig. 5. FCOS Architecture.

2.5 YOLOv8

YOLOv8 is the latest iteration of the popular YOLO (You Only Look Once) [21] family of real-time object detection systems. It builds upon the previous versions, incorporating several improvements and new features:

- **Efficient Architecture**: YOLOv8 adopts a new architecture called YOLO-Pose, which incorporates keypoint estimation capabilities in addition to object detection and classification. This allows for simultaneous detection and pose estimation of objects.
- **Improved Backbone**: YOLOv8 utilises an improved version of the EfficientNet backbone, called EfficientRep, which is more accurate and efficient than previous backbones used in YOLO models.
- **Data Augmentation**: YOLOv8 employs advanced data augmentation techniques, such as Mosaic data augmentation and self-supervised learning, to improve the model's performance and generalisation capabilities.
- **Ensemble Learning**: YOLOv8 introduces ensemble learning, where multiple models are combined to improve accuracy, particularly for small object detection.

YOLOv8 **Architecture**: The YOLOv8 architecture introduces numerous improvements compared to the earlier versions of YOLO, such as a new neural network architecture that utilises both Feature Pyramid Network (FPN) and Path Aggregation Network (PAN). This modular design consists of two main components: the backbone network and the detection head, refer to Fig. 6.

The backbone network is responsible for extracting multi-scale features from the input image. It generates feature maps at different scales (P5, P4, P3) which are subsequently utilised by the detection head. The detection head receives these feature maps and processes them through a series of operations including concatenation (C), upsampling (U), and convolutional layers (Conv, C2f) to fuse and process the multi-scale information. These processed features are then fed into the Detect layers, where the final predictions are made. The Detect layers generate bounding boxes, object class probabilities, and optionally, keypoint predictions.

In terms of performance enhancement, the YOLOv8 architecture incorporates additional components such as CIoU (Complete Intersection over Union) loss for bounding box regression and DFL (Distribution Focal Loss) for classification. These enhancements contribute to improved accuracy and robustness in object detection tasks.

3 Experiments and Results

Key metrics for this evaluation are Intersection over Union (IoU) and Mean Average Precision (MAP), essential for assessing the accuracy of object detection models. Additionally, the chapter examines performance graphs of models providing insights into their learning effectiveness and generalisation capabilities.

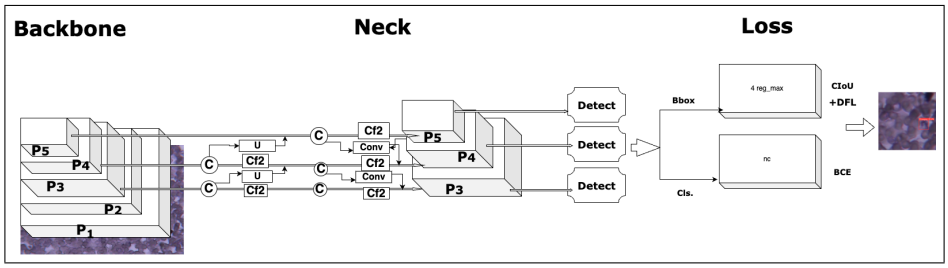

Fig. 6. YOLOv8 Architecture.

Intersection over Union (IoU) is a vital metric in object detection, used for evaluating the accuracy of bounding box predictions. It measures the degree of overlap between the predicted and ground truth (GT) boxes. IoU is calculated by dividing the area of overlap between the predicted and GT boxes by the area of their union. It is integral during training for refining bounding box predictions and during evaluation to assess model performance. Typically, a threshold, such as 0.5, is set for IoU, above which predictions are considered accurate. This makes IoU a fundamental standard for assessing the precision of object detection models.

$$\text{IoU} = \frac{\text{Area of Overlap between Predicted Box and Ground Truth Box}}{\text{Area of Union between Predicted Box and Ground Truth Box}}$$

Mean Average Precision (mAP) is a key metric in object detection, offering a comprehensive model performance evaluation by considering precision and recall across various IoU thresholds. Precision is the ratio of correct positive predictions to all positive predictions made by the model, while recall is the proportion of actual positives correctly identified.

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

where, TP is true positive, FP is false positive, and FN is false negative.

For object detection tasks, the precision and recall is calculated using the IoU value for a given IoU threshold. For example, when dealing with an IoU threshold of 0.5, a prediction with an IoU value of 0.7 is designated as a true positive (TP). Conversely, if the IoU falls to 0.3, we categorise the prediction as a false positive (FP).

The general definition of average precision (AP) involves calculating the area under the precision-recall curve, where as mAP is computed as the average of these individual AP values. mAP calculation involves ranking detection, calculating precision and recall at each threshold, plotting Precision-Recall curves, and averaging the area under these curves across classes. mAP is crucial in scenarios requiring a balance between precision and recall, and its interpretation varies with different IoU thresholds.

3.1 Results for all models and Comparative Analysis of Detection Models Performance

In this section, we show the comparison of four object detection models: RetinaNet, Faster-RCNN, FCOS, and YOLOv8, focusing on their mAP scores at different IoU thresholds. From our experiments we notice that:

RetinaNet scores an mAP of 0.797 at a 50% IoU threshold and 0.360 on average from 50% to 95%. This show a good performance, but there is room for improvement, especially at higher overlaps.

Faster-RCNN does better compared to RetinaNet. It reaches 0.913 at the 50% IoU and 0.498 on average up to 95%. This means it is quite accurate for moderate overlaps.

FCOS scores even higher in a range of 0.930 at 50% IoU and 0.577 on average to 95%. It shows great accuracy for basic overlaps and holds up well even as the overlap standards get stronger.

YOLOv8 leads to 0.951 at 50% IoU and 0.594 for 50-95%. This value shows that YOLOv8 outperforms, both at the basic 50% overlap and across a range of stronger overlaps.

All the results above are summarised in Table 1 and Fig. 7.

Table 1. Comparative Analysis of Detection Models Performance. Last row shows an outperformance of YOLOv8 compared to RetinaNet, FasterRCNN, and FCOS.

MODELS	**Mean Average Precision (mAP50)**	**mAP50-95**
RetinaNet	0.797	0.360
Faster-RCNN	0.913	0.498
FCOS	0.930	0.577
YOLOv8	**0.951**	**0.594**

The grouped bar chart compares the performance of four object detection models: RetinaNet, Faster-RCNN, FCOS, and YOLOv8, using Mean Average Precision (mAP) scores at two different thresholds. The blue bars represent the mAP at a 50% Intersection over Union (IoU) threshold (mAP50), while the green bars show the average mAP calculated at IoU thresholds ranging from 50% to 95% (mAP50-95).

From the chart, we see that YOLOv8 outperforms other models with the highest mAP scores at both thresholds, followed by FCOS, Faster-RCNN, and RetinaNet. The chart clearly indicates that as the models progress from RetinaNet to YOLOv8, there is an apparent improvement in their ability to accurately detect and classify objects in images, both at the baseline 50% IoU and across a spectrum of stronger IoU thresholds. Fig. 8 shows an example where RetinaNet, Faster-RCNN, FCOS models would fail to detect the Chagas parasite where as YOLOv8 successfully captures them.

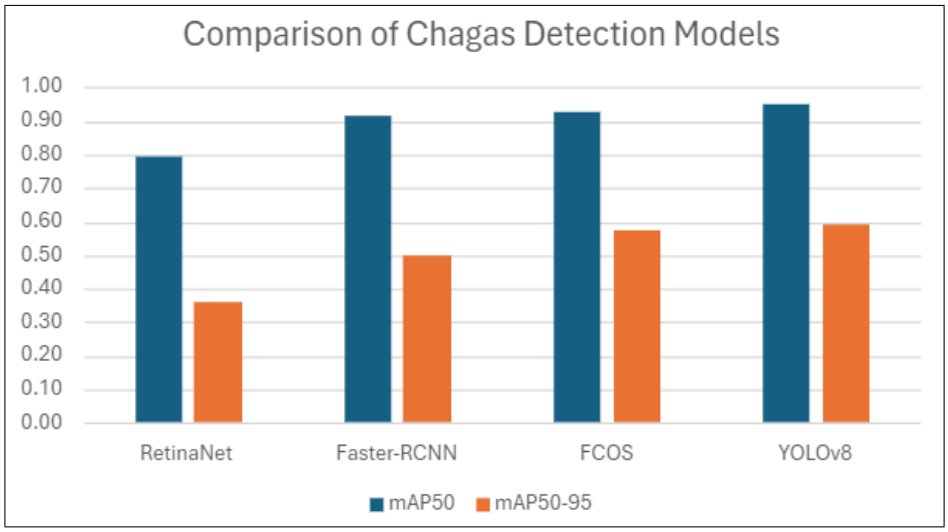

Fig. 7. Comparative Analysis of Detection Models Performance.

4 Conclusion

Our analysis encompassed a range of state-of-the-art models, including RetinaNet, Faster-RCNN, FCOS, and YOLOv8. The performance of these models was evaluated using key metrics such as Mean Average Precision (mAP) at different Intersection over Union (IoU) thresholds. The results demonstrated that while all models showed a high degree of performance in detecting Chagas parasites, the YOLOv8 model outperformed others, especially in terms of mAP50 score of 0.951. This indicates its superior ability to accurately detect and localise the parasites in medical images.

References

1. Bern C. Chagas' disease. *N Engl J Med.*, 30(373):456–66, 2015.
2. Beatriz S. L. Pires de Lima Marcelo Beckmann, Nelson F. F. Ebecken. A knn undersampling approach for data balancing. *Journal of Intelligent Learning Systems and Applications*, 7(4), 1951.
3. V. Vapnik, S. E. Golowich, and Smola A. J. Support vector method for function approximation, regression estimation, and signal processing. *In Advances in Neural Information Processing Systems*, pages 281–287, 1997.
4. F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.
5. Schapire R. E. Freund Y. A decision-theoretic generalization of on-line learning and an application to boosting. *In Proceedings of the Second European Conference on Computational Learning Theory*, 65(6):23–37, 1995.
6. Centers for Disease Control and Prevention. Parasites - American Trypanosomiasis (also known as Chagas Disease).

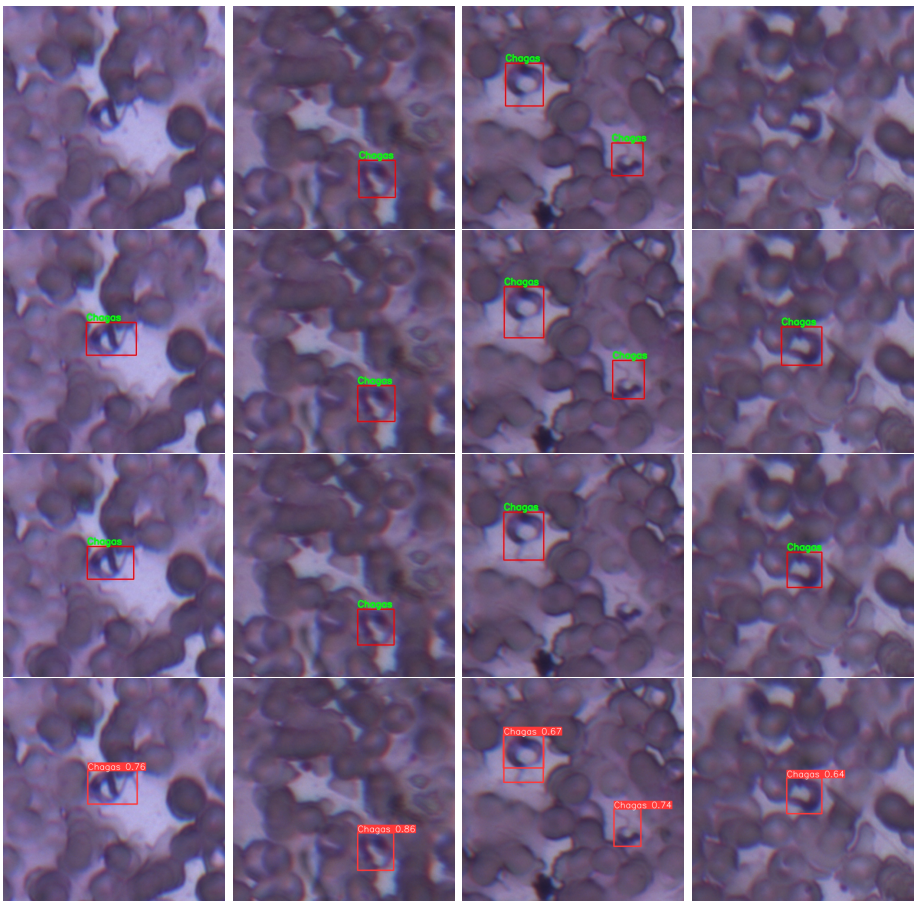

Fig. 8. First row RetinaNet output, second row Faster RCNN, third row FCOS, and forth row YOLOv8 Chagas detection.

https://www.cdc.gov/parasites/chagas/index.html, 2021. [Online; accessed 14-July-2022].
7. Roger Soberanis-Mukul, Víctor Uc-Cetina, Carlos Brito-Loeza, and Hugo Ruiz-Piña. An automatic algorithm for the detection of Trypanosoma cruzi parasites in blood sample images. *Computer methods and programs in biomedicine*, 112(3):633–639, 2013.
8. Víctor Uc-Cetina, Carlos Brito-Loeza, and Hugo Ruiz-Piña. Chagas parasite detection in blood images using Adaboost. *Computational and mathematical methods in medicine*, 2015, 2015.
9. André Pereira, Alexandre Pyrrho, Daniel Vanzan, Leonardo Mazza, and José Gomes. Deep convolutional neural network applied to chagas disease parasitemia assessment. In *Congresso Brasileiro de Inteligência Computacional*, pages 1–8, 01 2020.
10. Allan Ojeda-Pat, Anabel Martin-Gonzalez, and Roger Soberanis-Mukul. Convolutional neural network u-net for trypanosoma cruzi segmentation. In *International Symposium on Intelligent Computing Systems*, pages 118–131. Springer, 2020.
11. D. Morais, M. C. C.and Silva, M. M. Milagre, M. T. d Oliveira, T. Pereira, J. S. Silva, L. D. F. Costa, P. Minoprio, R. M. C. Junior, R. Gazzinelli, M. de Lana, and H. I. Nakaya. Automatic detection of the parasite Trypanosoma cruzi in blood smears using a machine learning approach applied to mobile phone images. *PeerJ*, 2022.
12. Allan Ojeda-Pat, Anabel Martin-Gonzalez, Carlos Brito-Loeza, Hugo Ruiz-Piña, and Daniel Ruz-Suarez. Effective residual convolutional neural network for chagas disease parasite segmentation. *Medical & Biological Engineering & Computing*, 60:1099–1110, 2022.
13. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
14. World Health Organization. Chagas disease (also known as American trypanosomiasis). https://www.who.int/news-room/fact-sheets/detail/chagas-disease-(american-trypanosomiasis), 2022. [Online; accessed 14-July-2022].
15. FX. Lescure, G. Le Loup, H. Freilij, M. Develoux, L. Paris, L. Brutus, and G. Pialoux. Chagas disease: changes in knowledge and management. *Lancet Infect Dis.*, 10(8):556*70, 2010.
16. Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.
17. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.
18. Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
19. Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
20. Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.
21. Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.