TELKOMNIKA Telecommunication Computing Electronics and Control
Vol. 24, No. 6, December 2026, pp. 2170~2182
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v24i6.28086

Instance segmentation of parasitic eggs in microscopic images using Cascade Mask R-CNN

Danny Matthew Saputra[1], Muhammad Azka Faridi[2], Rizki Kurniati[2]
[1]Department of Computer Systems, Faculty of Computer Science, Sriwijaya University, Palembang, Indonesia
[2]Department of Informatics Engineering, Faculty of Computer Science, Sriwijaya University, Palembang, Indonesia

Article Info

Article history:

Received May 31, 2026
Revised Sep 11, 2026
Accepted Oct 11, 2026

Keywords:

Cascade Mask R-CNN
Deep learning
Helminthiasis
Instance segmentation
Microscopic images
Parasitic eggs

ABSTRACT

Helminthiasis is recognized by the World Health Organization (WHO) as one of the neglected tropical diseases (NTDs) due to its widespread prevalence, yet it is often neglected in global health priorities. Conventional microscopic examination remains the primary approach for parasite egg detection, but it requires specialized expertise and is limited by low reproducibility and observer-dependent interpretation. This study proposes an instance segmentation framework using the Cascade Mask region-based convolutional neural networks (R-CNN) architecture to enable precise localization and pixel-level segmentation of individual parasitic eggs. The model was trained using a dataset of parasitic egg images that were re-annotated from bounding box annotations into instance masks. This study employed several backbone variants, namely ResNet-50, ResNet-101, ResNeXt-101, and ConvNeXt-T. The ConvNeXt-T backbone achieved the highest F1-score of 0.989. These findings demonstrate the potential of Cascade Mask R-CNN for accurate parasitic egg separation and quantification. Furthermore, a desktop-based application was developed for parasitic helminth egg segmentation.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Muhammad Azka Faridi
Department of Informatics Engineering, Faculty of Computer Science, Sriwijaya University
Palembang, Indonesia
Email: azkafaridi@gmail.com

1. INTRODUCTION

Helminthiasis is one of the most prevalent global health problems, particularly in developing countries, yet it often receives insufficient attention compared with other infectious diseases [1]. This infection is specifically classified among the "great neglected tropical diseases (NTDs)" due to the limited priority given to its global control despite its widespread impact. The World Health Organization (WHO) estimates that approximately 1.5 billion people, or 24% of the world's population, are infected with soil-transmitted helminths, with the highest prevalence reported in Sub-Saharan Africa, the Americas, China, and East Asia [2].

Effective control of helminthiasis largely depends on accurate diagnosis. Among the available detection methods, visual microscopic examination remains the most commonly used approach. The Kato-Katz technique applied to fecal samples is recognized as the gold standard method for identifying helminth eggs in cases of intestinal infection [3]. Environmental surveillance, including the collection and examination of flies in high-risk areas, may also support the early detection of environmental contamination [4].

However, manual egg monitoring requires specialized expertise and a high level of concentration, thereby imposing an additional economic burden, particularly for communities in remote rural areas [5]. This method also has several technical limitations, including low sensitivity and reproducibility, as well as susceptibility to interpretation errors caused by manual examination [6]. Furthermore, the complexity of

microscopic images characterized by variations in shape, staining intensity, and parasite density often complicates the identification process even for experienced observers [7].

To address these challenges, artificial intelligence-based digital image processing technologies have been developed to automate the identification of parasites and their eggs in microscopic images. Historically, the development of detection systems can be categorized into two stages, including semi-automated approaches utilizing traditional machine learning algorithms and fully automated approaches based on convolutional neural network (CNN) architectures [8]. Nevertheless, most previous studies have focused on bounding box-based object detection without performing precise object segmentation in microscopic images [8]–[25]. Such approaches tend to include excessive background regions, resulting in background noise that may affect the quality of features learned by the model [26].

On the other hand, previous research employed a staged approach combining block-matching and 3D filtering (BM3D) for denoising, U-shaped network (U-Net) for segmentation, and convolution and attention network (CoAtNet) for classification [27]. Although this approach achieved high accuracy, it still has limitations because it relies on semantic segmentation, which cannot distinguish individual egg objects separately [28]. This limitation may affect the egg quantification process because eggs per gram (EPG) in fecal samples is commonly used as an indicator for estimating disease burden in parasitic infection diagnostics [29], [30]. Another segmentation approach applies instance segmentation through the DT4PEIS method, which integrates detection transformer (DETR) with the segment anything model (SAM) [31]. However, this approach also has limitations because it is not fully integrated in an end-to-end framework and requires substantial computational resources due to the use of SAM.

To address these limitations, this study adopts Cascade Mask region-based convolutional neural networks (R-CNN) as an end-to-end instance segmentation approach. Unlike conventional Mask R-CNN, which employs only a single detection stage, this method utilizes a cascade detection mechanism with progressively stricter intersection over union (IoU) thresholds at each stage [32]. Several studies have demonstrated that cascade structures can improve detection quality and reduce identification errors, particularly in images with complex backgrounds [33], [34]. In medical image segmentation involving complex object boundaries and overlapping structures, a comparison study reported that Cascade R-CNN achieved a higher mean average precision (mAP) of 0.6245 compared with Mask R-CNN, which obtained 0.5909 [35]. Through this approach, it is expected that a system capable of performing precise parasite egg instance segmentation within a single integrated process while maintaining high performance can be developed.

2. METHOD

This study adopted a quantitative approach focusing on parasitic egg instance segmentation using the Cascade Mask R-CNN architecture. Several backbone variants, namely ResNet-50, ResNet-101, ResNeXt-101, and ConvNeXt-T, were evaluated and compared to determine the optimal configuration. Model training was conducted on the Kaggle platform using an NVIDIA Tesla P100 graphics processing unit (GPU) with 16 GB of video random access memory (VRAM). The overall research workflow is illustrated in Figure 1.

2.1. Data acquisition

The data source used in this study is secondary data in the form of the public microscopic image dataset Chula-ParasiteEgg-11 [36]. This dataset was first introduced in the ICIP 2022 challenge on parasitic egg detection with bounding box annotation format. It consists of 11 classes of parasitic eggs, namely *Ascaris lumbricoides*, *Capillaria philippinensis*, *Enterobius vermicularis*, *Fasciolopsis buski*, *Hookworm egg*, *Hymenolepis diminuta*, *Hymenolepis nana*, *Opisthorchis viverrine*, *Paragonimus spp.*, *Taenia spp.*, and *Trichuris trichiura*. The images in this dataset were acquired using several different devices, including a Canon EOS 70D camera with an Olympus BX53 microscope, a Nikon DS-Fi2 camera with a Nikon Eclipse Ni microscope, as well as smartphone devices such as the Samsung Galaxy J7 Prime, iPhone 12, and iPhone 13 using a 10× eyepiece lens on either a Nikon Eclipse Ni or Olympus BX53 microscope. This resulted in variations in resolution, illumination, and acquisition conditions. In addition, some images exhibit diverse quality characteristics, such as out-of-focus regions, noise, and motion blur caused by the use of a microscope equipped with a motorized stage. The wide variation in image quality and appearance provides a broad range of visual characteristics, which is expected to improve the robustness of the developed detection model. The dataset distribution is presented in Table 1.

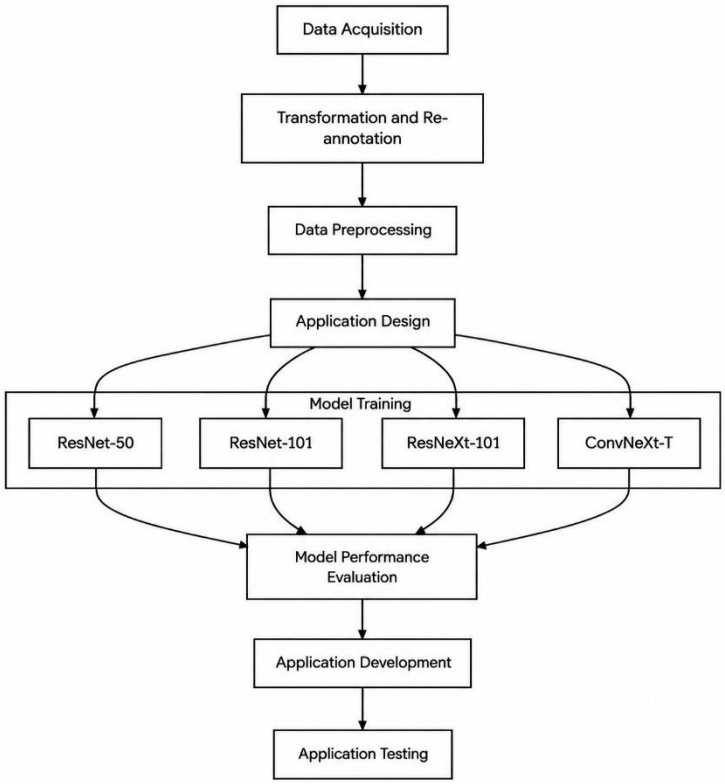


Figure 1. Research methodology

Table 1. Dataset class distribution

Label	Number of images
Ascaris lumbricoides	200
Capillaria philippinensis	200
Enterobius vermicularis	200
Fasciolopsis buski	200
Hookworm egg	200
Hymenolepis diminuta	200
Hymenolepis nana	200
Opisthorchis viverrine	200
Paragonimus spp	200
Taenia spp	200
Trichuris trichiura	200

2.2. Transformation and Re-annotation

The original dataset annotations, which were provided in bounding box format, were converted into instance segmentation annotations using Roboflow platform. The re-annotation process generated individual object masks, enabling the model to distinguish between multiple parasitic egg objects within the same class [37]. This conversion was performed because bounding box annotations provide limited object boundary information and may include surrounding background regions, artifacts, and debris, which can affect segmentation accuracy, while ensuring compatibility with the instance mask annotation format required by the Cascade Mask R-CNN architecture [26], [32].

2.3. Data preprocessing

Before being used in the training process, the image data underwent a preprocessing stage. The preprocessing steps consisted of:

- Applying auto-orientation and resizing the images into a fixed dimension of 800×800 pixels to ensure uniform input dimensions [38].
- Applying a letterboxing technique by adding black padding to the image borders in order to preserve the object proportions without introducing distortion.

- Splitting the dataset into three subsets using the stratified sampling method with a ratio of 80:10:10. This division produced a training set comprising 80% used for model weight training, a validation set comprising 10% used for evaluation during the training process, and a testing set comprising 10% used as unseen data for final model performance evaluation after training [39].
- Exporting the dataset into Microsoft common objects in context (MS COCO) segmentation JavaScript object notation (.json) format to adopt a standardized annotation structure widely used in instance segmentation tasks [40].

2.4. Model architecture and training configuration

The main stage of this study was the training of a parasitic helminth egg segmentation model using the Cascade Mask R-CNN architecture. During the training process, several parameters were kept constant. The detailed fixed parameter settings are presented in Table 2.

Table 2. Fixed parameter

Parameter name	Parameter value
Image size	800×800
Classes	11
Batch size	4
Optimizer	AdamW
Learning rate	0.0001
Weight decay	0.05
Scheduler	LinearLR + CosineAnnealingLR
Epoch	50
Loss function (bbox)	Smooth L1 loss
Loss function (classification)	Cross-entropy loss
Loss function (mask)	Cross-entropy loss (mask)

During the training stage, data augmentation techniques consisting of geometric and photometric transformations were applied. The geometric transformation-based augmentations included random flipping and random rotation to improve model performance when trained on datasets with limited data availability [41]. In addition, image visual property manipulation was also applied through photometric distortion, which included adjustments to brightness, contrast, saturation, and hue, preventing the model from becoming biased toward variations in lighting conditions during image acquisition [42]. These augmentation techniques were applied exclusively to the training data with the aim of improving the model's generalization capability, reducing overfitting, and producing a more robust model [43]. In this study, several backbone architecture variants were employed, including ResNet-50, ResNet-101, ResNeXt-101, and ConvNeXt-T [44]-[46]. The use of multiple backbone architectures was intended to compare the feature extraction capabilities of each architecture in improving model performance.

2.5. Model evaluation

The model evaluation in this study was conducted using recall, precision, and F1-score metrics [47]. In addition, mAP50–95 for both bounding boxes and segmentation masks was employed as an evaluation metric, representing the mAP across IoU thresholds ranging from 0.50 to 0.95 with an interval of 0.05 [48]. The formulas used to compute these metrics are presented as:

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (1)$$

$$Recall = \frac{TP}{allground - truths} \quad (2)$$

$$Precision = \frac{TP}{all\ detections} \quad (3)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

$$AP = \int_0^1 Precison(recall) \times dRecall \quad (5)$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \quad (6)$$

2.6. Application design

2.6.1. Requirement analysis

Software requirements are generally divided into two categories, namely functional requirements and non-functional requirements. Functional requirements represent the core requirements that must be fulfilled to ensure that the software operates according to its intended objectives. Meanwhile, non-functional requirements refer to supporting requirements that contribute to improving software quality, usability, and overall system performance [49]. The functional and non-functional requirements of the software developed in this study are presented in Tables 3 and 4.

Table 3. Functional requirements

No	Functional requirements
1	The software provides a login feature
2	The software allows users to create and manage projects
3	The software enables users to upload microscopic images into a project
4	The software is capable of performing automatic segmentation using the Cascade Mask R-CNN model
5	The software displays segmentation results in the form of segmentation masks, bounding boxes, labels, and confidence scores
6	The software can calculate the number of parasitic eggs per image and per project

Table 4. Non-functional requirements

No	Non-functional requirements
1	The software is capable of handling multiple image data within a single project
2	The software provides a user-friendly interface for users

2.6.2. Entity relationship diagram (ERD)

The ERD is used to model entities, attributes, and relationships among entities within a database system, thereby providing a clear representation of the data structure and its relationships [50]. The ERD used in this study is presented in Figure 2.

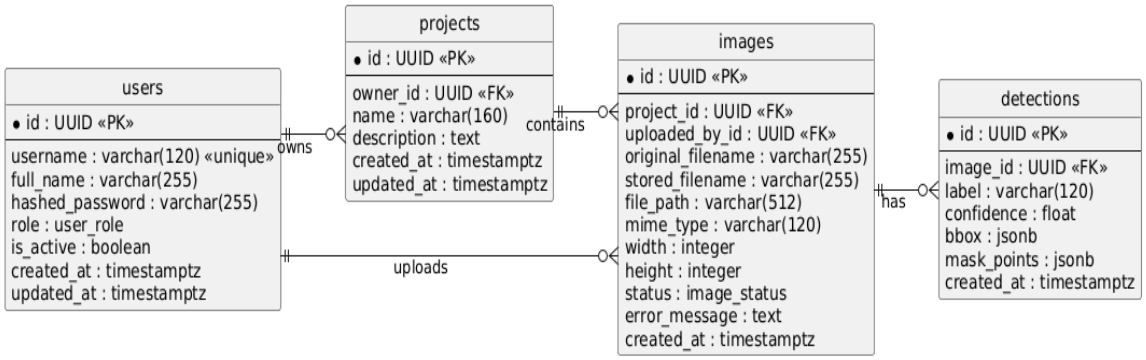


Figure 2. ERD

2.6.3. User interface

Before the software interface development process, an interface design was initially created in the form of a low-fidelity prototype. This design was represented through a simple wireframe to provide an initial overview of the interface layout and structure prior to the implementation stage [51]. The resulting low-fidelity interface design used in this study is presented in Figures 3(a) to (e).

2.6.4. Application testing

Software testing in this study was conducted using the black-box testing method to ensure that all developed functionalities operated according to the specified requirements without considering the internal structure or implementation of the system [52]. The testing scenarios are presented in Table 5.



Figure 3. Low-fidelity wireframes: (a) login page, (b) dashboard page, (c) project page, (d) detail project page, and (e) detail image page

Table 5. Software testing scenarios

ID	Testing	Testing level
T-01-1	Displaying the login page	Interface testing
T-01-2	Displaying the dashboard page after successful login	Interface testing
T-01-3	Displaying error message when login fails	Interface testing
T-02-1	Displaying the project creation form	Interface testing
T-02-2	Displaying the newly created project	Interface testing
T-03-1	Selecting images into a project	Interface testing
T-03-2	Displaying the image processing status	Interface testing
T-03-3	Displaying the successful image processing status	Interface testing
T-03-4	Displaying the failed image processing status	Interface testing
T-04-1	Displaying the project list	Interface testing
T-04-2	Displaying the image list within a project	Interface testing
T-04-3	Displaying segmentation results in the form of masks, bounding boxes, labels, and confidence scores	Interface testing

3. RESULTS AND DISCUSSION

The original bounding box annotations were converted into segmentation annotations. After the re-annotation process, the dataset underwent image preprocessing, which included auto-orientation, resizing the images to a fixed dimension of 800×800 pixels, and applying a letterboxing technique by adding black padding to the image borders to preserve object proportions without distortion. Examples of the re-annotated and preprocessed images are shown in Figure 4. Subsequently, the dataset was divided into three subsets using the stratified sampling method with an 80:10:10 ratio. The dataset split distribution is presented in Table 6. The training visualizations for each backbone architecture variation are presented in Figures 5(a) and (b). Based on the validation results, ResNet-50 achieved a bbox mAP50–95 of 0.918 and a segm mAP50–95 of 0.939. ResNet-101 obtained a bbox mAP50–95 of 0.914 and a segm mAP50–95 of 0.945. ResNeXt-101 produced a bbox mAP50–95 of 0.918 and a segm mAP50–95 of 0.946. Meanwhile, ConvNeXt-T demonstrated the best performance with a bbox mAP50–95 of 0.935 and a segm mAP50–95 of 0.954 on the validation data. The trained model was then evaluated on the test data to assess its performance based on recall, precision, F1-score, as well as mAP50–95 for both bounding boxes and segmentation masks, as presented in Table 7.

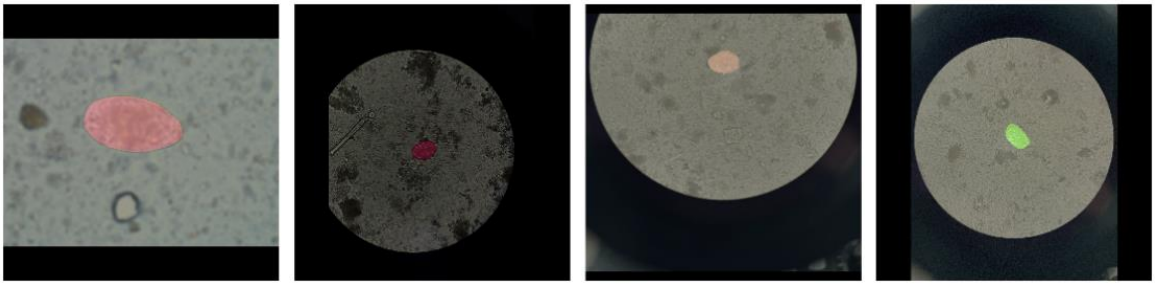
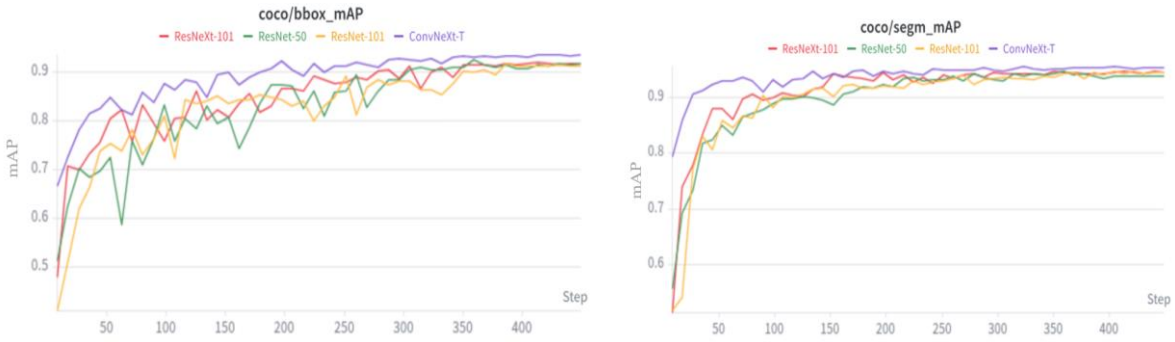


Figure 4. Sample images after re-annotation and preprocessing

Table 6. Dataset split distribution

Label	Training set	Validation set	Testing set
Ascaris lumbricoides	160	20	20
Capillaria philippinensis	160	20	20
Enterobius vermicularis	160	20	20
Fasciolopsis buski	160	20	20
Hookworm egg	160	20	20
Hymenolepis diminuta	160	20	20
Hymenolepis nana	160	20	20
Opisthorchis viverrine	160	20	20
Paragonimus spp	160	20	20
Taenia spp	160	20	20
Trichuris trichiura	160	20	20



(a) (b)

Figure 5. Training performance: (a) Bbox mAP50-95 and (b) Segm mAP50-95

Table 7. Model performance evaluation results

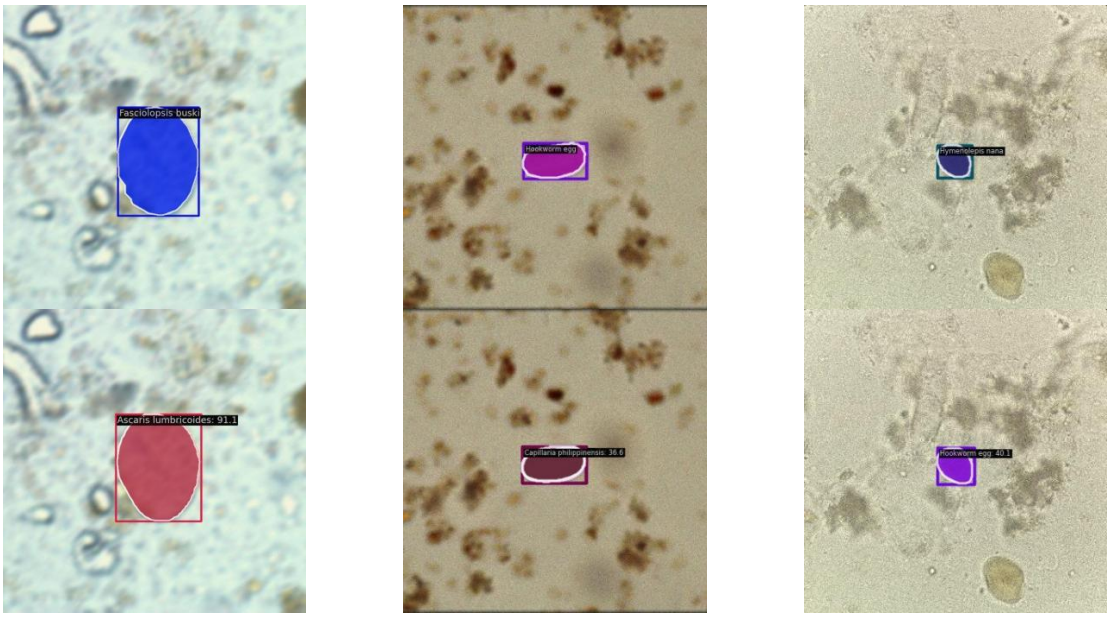
Model	Recall	Precision	F1-score	bbox mAP50-95	segm mAP50-95
Cascade Mask R-CNN + ResNet-50	0.978	0.971	0.974	0.887	0.908
Cascade Mask R-CNN + ResNet-101	0.987	0.977	0.982	0.889	0.910
Cascade Mask R-CNN + ResNeXt-101	0.987	0.980	0.983	0.901	0.918
Cascade Mask R-CNN + ConvNeXt-T	0.991	0.987	0.989	0.916	0.936

Based on the experimental results across all backbone variations, Cascade Mask R-CNN demonstrated excellent performance in segmenting parasitic eggs in microscopic images, with recall, precision, and F1-score values exceeding 97% across all evaluated models. These results demonstrate the models' ability to accurately distinguish parasitic egg instances from the background. The high precision values indicate a low number of false-positive detections, whereas the high recall values reflect the models' effectiveness in detecting parasitic eggs. Among the evaluated configurations, ConvNeXt-T achieved the best performance and was therefore selected as the proposed model in this study, obtaining a recall of 0.991, precision of 0.987, F1-score of 0.989, bbox mAP50–95 of 0.916, and segm mAP50–95 of 0.936. It was

followed by ResNeXt-101, ResNet-101, and ResNet-50. The per-class performance and representative misclassification cases of the selected model are presented in Table 8 and Figures 6(a) to (c).

Table 8. Per-class performance of Cascade Mask R-CNN with ConvNeXt-T

Label	Recall	Precision	F1-score	bbox mAP50-95	segm mAP50-95
Ascaris lumbricoides	1.00	0.997	0.999	0.96	0.975
Capillaria philippinensis	1.00	1.00	1.00	0.875	0.901
Enterobius vermicularis	1.00	1.00	1.00	0.891	0.913
Fasciolopsis buski	0.952	0.95	0.951	0.892	0.91
Hookworm egg	0.95	0.926	0.938	0.815	0.826
Hymenolepis diminuta	1.00	1.00	1.00	0.981	0.994
Hymenolepis nana	1.00	0.997	0.999	0.937	0.949
Opisthorchis viverrine	1.00	1.00	1.00	0.916	0.936
Paragonimus spp	1.00	1.00	1.00	0.97	0.993
Taenia spp	1.00	1.00	1.00	0.931	0.959
Trichuris trichiura	1.00	0.987	0.993	0.909	0.945



(a) (b) (c)

Figure 6. Misclassification results of Cascade Mask R-CNN with ConvNeXt-T: (a) *Fasciolopsis buski* misclassified as *Ascaris lumbricoides*, (b) *hookworm egg* misclassified as *Capillaria philippinensis*, and (c) *Hymenolepis diminuta* misclassified as hookworm egg

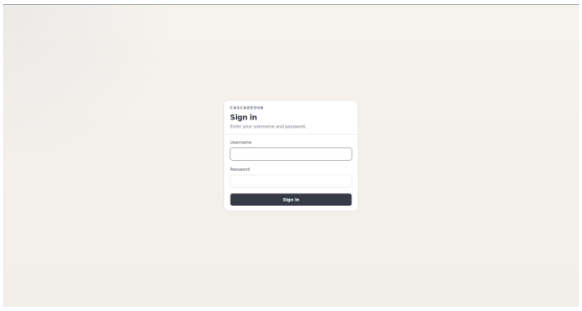
The Cascade Mask R-CNN model with ConvNeXt-T demonstrated high and consistent performance across nearly all parasitic egg classes. Most classes achieved perfect or near-perfect recall, precision, and F1-score values. However, two classes obtained relatively lower F1-scores, namely Fasciolopsis buski with 0.951 and hookworm egg with 0.938. The bbox mAP50–95 and segm mAP50–95 values were also relatively similar for most classes. However, hookworm egg obtained the lowest values and showed a considerable difference compared with the other classes, recording a bbox mAP50–95 of 0.815 and a segm mAP50–95 of 0.826. Several misclassifications were observed, presumably due to morphological similarities among the classes, particularly in egg shape and internal structure. In addition, blurred microscopic images made the internal structures and visual characteristics of the eggs difficult to distinguish clearly. The combination of morphological similarities and image blurring may reduce the model's ability to extract discriminative features among classes, thereby resulting in misclassification. A comparison between the proposed model and previous studies is presented in Table 9.

The implemented software interface was developed as a desktop-based application. This software enables users to upload microscopic images, which are subsequently processed through an inference stage using the Cascade Mask R-CNN model. The resulting outputs include segmentation masks, bounding boxes, species classification results, and confidence scores. The implementation results of the software interface are

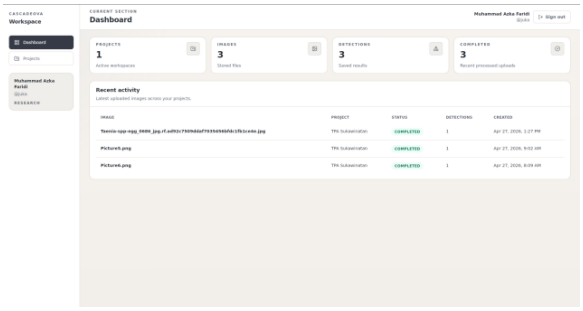
presented in Figures 7(a) to (e). In addition, the software was evaluated using the black-box testing method. The results of the software testing are presented in Table 10.

Table 9. Comparison of proposed method with related work

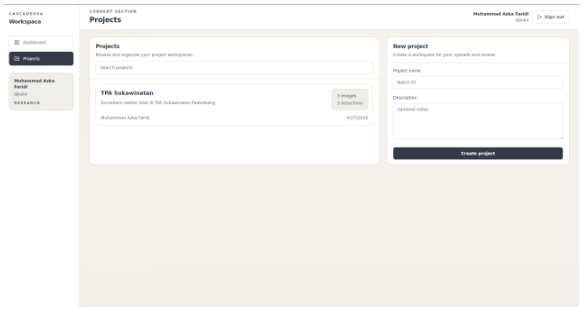
Paper	Model	Recall	Precision	F1-score	bbox mAP50-95	segm mAP50-95
[13]	Transformer-based	-	-	0.955	-	-
[14]	YOLOv5	-	-	0.864	-	-
[15]	Ensemble model	-	-	0.974	-	-
[16]	TOOD	-	-	0.964	-	-
[17]	RetinaNet	0.870	0.990	0.926	0.820	-
[18]	EfficientNet-V2 + EfficientNet-B7 + SVM	0.930	0.940	0.930	-	-
[19]	EfficientNet-B2 + YOLOv5	0.908	0.793	0.846	-	-
[20]	ResNet-50 + SE	0.940	0.940	0.940	-	-
[8]	YOLOv5n + AFPN + C2f	0.977	0.978	0.977	-	-
[21]	Cacade R-CNN + TOOD + RetinaNet	0.944	0.951	0.947	0.882	-
[22]	EfficientNetB0	0.954	0.958	0.955	-	-
[23]	RT-DETR + ResNet-50	-	-	0.978	0.908	-
[27]	BM3D + Wiener + CLAHE + U-Net + CNN	0.976	0.971	0.977	-	-
[24]	YOLOv11n + DropBlock	0.902	0.943	0.922	-	-
Ours	Cascade Mask R-CNN + ConvNeXt-T	0.991	0.987	0.989	0.916	0.936



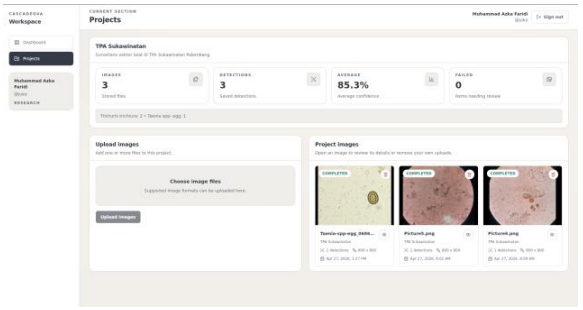
(a)



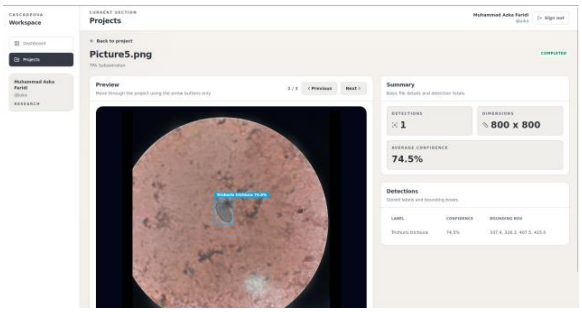
(b)



(c)



(d)



(e)

Figure 7. User interface: (a) login page, (b) dashboard page, (c) project page, (d) detail project page, and (e) detail image page

Table 10. Software testing results

ID	Description	Testing procedure	Expected output	Actual output	Conclusion
T-01-1	Displaying the login page	Open the application	The login page is displayed	The system displays the login page	Passed
T-01-2	Displaying the dashboard page after successful login	Enter a valid email and password, then click the login button	The dashboard page is displayed	The system displays the dashboard page after successful login	Passed
T-01-3	Displaying error message when login fails	Enter an incorrect email and password, then click the login button	A login failure message is displayed	The system displays a login failure message	Passed
T-02-1	Displaying the project creation form	Select the create project menu	The project creation form is displayed	The system displays the project creation form	Passed
T-02-2	Displaying the newly created project	Enter the project name and description, then click the create button	The new project appears on the project page	The system displays the newly created project	Passed
T-03-1	Selecting images into a project	Click the upload button and select images	The selected images appear in the application	The system displays the selected images	Passed
T-03-2	Displaying the image processing status	Click the upload button after selecting images	The processing status is displayed	The system displays the processing status	Passed
T-03-3	Displaying the successful image processing status	Wait until the segmentation process is completed	The successful status is displayed	The system displays the successful status	Passed
T-03-4	Displaying the failed image processing status	Upload a corrupted image	The failed status is displayed	The system displays the failed status	Passed
T-04-1	Displaying the project list	Open the project page	The project list is displayed	The system displays the project list	Passed
T-04-2	Displaying the image list within a project	Open one of the projects	The image list is displayed	The system displays the image list within the project	Passed
T-04-3	Displaying segmentation results in the form of masks, bounding boxes, labels, and confidence scores	Select a processed image	Segmentation results consisting of masks, bounding boxes, labels, and confidence scores are displayed	The system displays segmentation results (masks, bounding boxes, labels, and confidence scores)	Passed

4. CONCLUSION

Based on the experimental results, the model with the ConvNeXt-T backbone was selected as the best configuration, achieving a recall of 0.991, precision of 0.987, F1-score of 0.989, bbox mAP50–95 of 0.916, and segm mAP50–95 of 0.936. These results demonstrate its ability to localize parasitic eggs and generate separate pixel-level masks, supporting more precise egg separation and quantification. The best model was also successfully integrated into a desktop application, whose functionality was verified through black-box testing. The proposed system shows promise as a computer-assisted tool for clinical laboratory analysis by minimizing observer dependency and supporting laboratory personnel in identifying and quantifying parasitic eggs. However, this study relied on data from a single source and a dataset with limited size and image diversity. The system was not directly integrated with a microscope. Future research should therefore focus on microscope integration to enable real-time diagnostic analysis. Lightweight models capable of instance segmentation should also be explored to improve computational efficiency and real-time performance. Furthermore, increasing the number of parasitic egg classes and using larger, more diverse datasets are necessary to support a broader range of diagnostic scenarios and strengthen the model's generalizability.

ACKNOWLEDGMENTS

The authors would like to express their sincere gratitude to Universitas Sriwijaya for supporting this research through the 2026 Beginner Lecturer Research Scheme.

FUNDING INFORMATION

This research received financial support from Universitas Sriwijaya in accordance with Rector's Decree No. 0014/UN9.R/SK.R3.LPPM/2026.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Danny Mathew Saputra	✓			✓			✓			✓		✓		
Muhammad Azka Faridi	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	
Rizki Kurniati	✓			✓				✓		✓		✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, M.A.F., upon reasonable request.

REFERENCES

[1] P. J. Hotez, P. J. Brindley, J. M. Bethony, C. H. King, E. J. Pearce, and J. Jacobson, "Helminth infections: The great neglected tropical diseases," *Journal of Clinical Investigation*, vol. 118, no. 4, pp. 1311–1321, 2008, doi: 10.1172/JCI34261.

[2] W. H. Organization, "Soil-transmitted helminth infections," World Health Organization. Accessed: May 14, 2026. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/soil-transmitted-helminth-infections

[3] A. S. M. Al Amin and R. Wadhwa, *Helminthiasis*. Treasure Island (FL): StatPearls Publishing, 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK560525/

[4] M. F. Ridho, D. Dalilah, D. Handayani, G. D. Prasasty, and Susilawati, "Contamination of Soil-Transmitted Helminth (Sth) Eggs on Fly Vectors At the Sukawinatan Final Disposal Site (Tpa) in Palembang City," *Jurnal Kedokteran dan Kesehatan : Publikasi Ilmiah Fakultas Kedokteran Universitas Sriwijaya*, vol. 12, no. 2, pp. 176–185, 2025, doi: 10.32539/jkk.v12i2.640.

[5] P. Oyibo *et al.*, "Deep learning-based automated detection and multiclass classification of soil-transmitted helminths and Schistosoma mansoni eggs in fecal smear images," *Scientific Reports*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-02755-9.

[6] P. Ward *et al.*, "Affordable artificial intelligence-based digital pathology for neglected tropical diseases: A proof-of-concept for the detection of soil-transmitted helminths and Schistosoma mansoni eggs in Kato-Katz stool thick smears," *PLoS Neglected Tropical Diseases*, vol. 16, no. 6, p. e0010500, 2022, doi: 10.1371/JOURNAL.PNTD.0010500.

[7] S. Kumar, T. Arif, G. Ahamad, A. A. Chaudhary, S. Khan, and M. A. M. Ali, "An Efficient and Effective Framework for Intestinal Parasite Egg Detection Using YOLOv5," *Diagnostics*, vol. 13, no. 18, p. 2978, 2023, doi: 10.3390/diagnostics13182978.

[8] W. Xu, Q. Zhai, J. Liu, X. Xu, and J. Hua, "A lightweight deep-learning model for parasite egg detection in microscopy images," *Parasites and Vectors* , vol. 17, no. 1, 2024, doi: 10.1186/s13071-024-06503-2.

[9] A. Kitvimonrat, N. Hongcharoen, S. Marukatat and S. Watcharabutsarakham, "Automatic Detection and Characterization of Parasite Eggs using Deep Learning Methods," *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Phuket, Thailand*, 2020, pp. 153-156, doi: 10.1109/ECTI-CON49241.2020.9158084.

[10] T. Suwannaphong, S. Chavana, S. Tongsom, D. Palasuwan, T. H. Chalidabhongse, and N. Anantrasirichai, "Parasitic Egg Detection and Classification in Low-Cost Microscopic Images Using Transfer Learning," *SN Computer Science*, vol. 5, no. 1, 2024, doi: 10.1007/s42979-023-02406-8.

[11] P. Mayo, N. Anantrasirichai, T. H. Chalidabhongse, D. Palasuwan, and A. Achim, "Detection of Parasitic Eggs from Microscopy Images and the emergence of a new dataset," 2022, *arXiv*. doi: 10.48550/ARXIV.2203.02940.

[12] N. AlDahoul, H. A. Karim, S. L. Kee and M. J. T. Tan, "Localization and Classification of Parasitic Eggs in Microscpic Images Using An Efficientdet Detector," *2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France*, 2022, pp. 4253-4257, doi: 10.1109/ICIP46576.2022.9897844.

[13] A. Pedraza, J. Ruiz-Santaquiteria, O. Deniz, and G. Bueno, "Parasitic Egg Detection and Classification With Transformer-Based Architectures," *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 4301–4305, Oct. 16, 2022. doi: 10.1109/icip46576.2022.9897846.

[14] Y. Pratama, Y. Fujimura, T. Funatomi, and Y. Mukaigawa, "Parasitic Egg Detection and Classification By Utilizing the Yolo Algorithm With Deep Latent Space Image Restoration and Grabcut Augmentation," *2022 IEEE International Conference on Image Processing (ICIP), pp. 4311–4315, Oct. 16, 2022*. doi: 10.1109/icip46576.2022.9897645.

[15] J. Ruiz-Santaquiteria, A. Pedraza, N. Vallez, and A. Velasco, "Parasitic Egg Detection With a Deep Learning Ensemble," *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 4283–4286, Oct. 16, 2022. doi: 10.1109/icip46576.2022.9897858.

[16] A. Tureckova, T. Turecek, and Z. K. Oplatkova, "ICIP 2022 Challenge: PEDCMI, TOOD Enhanced by Slicing-Aided Fine-Tuning and Inference," *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 4292–4295, Oct. 16, 2022. doi: 10.1109/icip46576.2022.9897826.
[17] K. Pho, H. Lam, T. Le, H. T. Nguyen, and A. Yoshitaka, "Attention-driven RetinaNet for Parasitic Egg Detection," *2022 IEEE International Symposium on Multimedia (ISM)*, pp. 265–272, Dec. 2022. doi: 10.1109/ism55400.2022.00060.
[18] N. AlDahoul, H. A. Karim, M. A. Momo, F. I. F. Escobar, V. A. Magallanes, and M. J. T. Tan, "Parasitic egg recognition using convolution and attention network," *Scientific Reports*, vol. 13, no. 1, 2023, doi: 10.1038/s41598-023-41711-3.
[19] N. Penpong, Y. Wanna, C. Kamjanlard, A. Techasen, and T. Intharah, "Attacking the out-of-domain problem of a parasite egg detection in-the-wild," *Heliyon*, vol. 10, no. 4, p. e26153, 2024, doi: 10.1016/j.heliyon.2024.e26153.
[20] M. Muthulakshm, K. Venkatesan, S. B. Rahayu, and K. Elangovan, "A Squeeze-Excitation ResNet Approach for Effective Classification of Parasitic Eggs," *2024 International Visualization, Informatics and Technology Conference (IVIT)*, pp. 95–100, Aug. 07, 2024. doi: 10.1109/ivit62102.2024.10692960.
[21] M. L. L. Bessa, G. B. Junior, and J. D. S. De Almeida, "Neural Network Ensemble for Detecting Parasite Eggs in Microscopic Images," *Procedia Computer Science*, vol. 256, pp. 739–746, 2025, doi: 10.1016/j.procs.2025.02.174.
[22] I. P. Sari, B. Warsito, and O. D. Nurhayati, "Automated Classification of Parasitic Worm Eggs Based on Transfer Learning and Fine-Tuned CNN Models," *International Journal of Advanced Computer Science and Applications*, vol. 16, no. 5, pp. 105–110, 2025, doi: 10.14569/IJACSA.2025.0160511.
[23] I.-V. Petrychko, "Automated Detection and Classification of Parasitic Infections," 2025, [Online]. Available: https://er.ucu.edu.ua/server/api/core/bitstreams/392cefb8-e5cd-4805-a2f7-3a8a038fd634/content
[24] G. Giri, A. Shrestha, and V. Sawad, "Lightweight Parasitic Egg Detection using Modified YOLOv11 with Ghost Convolutions, Hybrid SimAM-ECA Attention, and DropBlock Regularization," *International Journal on Engineering Technology*, vol. 3, no. 1, pp. 167–178, 2025, doi: 10.3126/injet.v3i1.87020.
[25] M. B. Zia, X. Zhou, R. Gururajan, and K. C. Chan, "ParaVisionNet: A multitask vision transformer framework for accurate detection and classification of parasitic eggs in microscopy images," *Computers and Electrical Engineering*, vol. 132, p. 110978, 2026, doi: 10.1016/j.compeleceng.2026.110978.
[26] B. R. Kang, H. Lee, K. Park, H. Ryu, and H. Y. Kim, "BshapeNet: Object detection and instance segmentation with bounding shape masks," *Pattern Recognition Letters*, vol. 131, pp. 449–455, 2020, doi: 10.1016/j.patrec.2020.01.024.
[27] S. K. Mondal, M. Islam, and M. S. U. Yusuf, "An artificial intelligence-based approach for human parasite egg segmentation and classification," *Healthcare Analytics*, vol. 8, p. 100432, 2025, doi: 10.1016/j.health.2025.100432.
[28] J. M. Molina, J. P. Llerena, L. Usero, and M. A. Patricio, "Advances in instance segmentation: Technologies, metrics and applications in computer vision," *Neurocomputing*, vol. 625, p. 129584, 2025, doi: 10.1016/j.neucom.2025.129584.
[29] C. C. Lee *et al.*, "Helminth egg analysis platform (HEAP): An opened platform for microscopic helminth egg identification and quantification based on the integration of deep learning architectures," *Journal of Microbiology, Immunology and Infection*, vol. 55, no. 3, pp. 395–404, 2022, doi: 10.1016/j.jmii.2021.07.014.
[30] J. von Bahr *et al.*, "AI-supported versus manual microscopy of Kato-Katz smears for diagnosis of soil-transmitted helminth infections in a primary healthcare setting," *Scientific Reports*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-07309-7.
[31] J. Ruiz-Santaquiteria, A. Pedraza, O. Deniz, and G. Bueno, "DT4PEIS: detection transformers for parasitic egg instance segmentation," *Applied Intelligence*, vol. 55, no. 4, 2025, doi: 10.1007/s10489-024-06199-y.
[32] Z. Cai and N. Vasconcelos, "Cascade R-CNN: High quality object detection and instance segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 5, pp. 1483–1498, 2021, doi: 10.1109/TPAMI.2019.2956516.
[33] B. Zhang and L. Xu, "Oriented Cascade Mask R-CNN for Biomedical Image Segmentation," *2023 35th Chinese Control and Decision Conference (CCDC)*, pp. 2372–2377, May 20, 2023. doi: 10.1109/ccdc58219.2023.10326634.
[34] F. Xu *et al.*, "Improved Cascade Mask RCNN Single Character Detection Algorithm for Cast Pipe," *2023 4th International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE), Nanjing, China, 2023*, pp. 162–167, Aug. 25, 2023. doi: 10.1109/icbase59196.2023.10303109.
[35] Y. Zhang *et al.*, "V-UNet: Medical Image Segmentation Based on Variational Attention Mechanism," *CAAI Transactions on Intelligence Technology*, vol. 10, no. 5, pp. 1350–1362, 2025, doi: 10.1049/cit2.70053.
[36] N. Anantrasirichai *et al.*, "ICIP 2022 Challenge on Parasitic Egg Detection and Classification in Microscopic Images: Dataset, Methods and Results," in *2022 IEEE International Conference on Image Processing (ICIP)*, Oct. 2022, pp. 4306–4310. doi: 10.1109/ICIP46576.2022.9897267.
[37] A. M. Hafiz and G. M. Bhat, "A survey on instance segmentation: state of the art," *International Journal of Multimedia Information Retrieval*, vol. 9, no. 3, pp. 171–189, 2020, doi: 10.1007/s13735-020-00195-x.
[38] R. Kurniawan, S. Samsuryadi, and F. S. Mohamad, "Improved Classification of Arsenic-Affected Skin Diseases through Image Processing and Transfer Learning," *CommIT Journal*, vol. 19, no. 1, pp. 71–88, 2025, doi: 10.21512/commit.v19i1.11891.
[39] V. R. Joseph, "Optimal ratio for data splitting," *Statistical Analysis and Data Mining*, vol. 15, no. 4, pp. 531–538, 2022, doi: 10.1002/sam.11583.
[40] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," *Lecture Notes in Computer Science. Springer International Publishing*, pp. 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48.
[41] L. Perez and J. Wang, "The Effectiveness of Data Augmentation in Image Classification using Deep Learning," 2017, *arXiv*. doi: 10.48550/ARXIV.1712.04621.
[42] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.
[43] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches," *Array*, vol. 16, p. 100258, 2022, doi: 10.1016/j.array.2022.100258.
[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, Jun. 2016. doi: 10.1109/cvpr.2016.90.
[45] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5987–5995, Jul. 2017. doi: 10.1109/cvpr.2017.634.
[46] Z. Liu, H. Mao, C. Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11966–11976, Jun. 2022. doi: 10.1109/cvpr52688.2022.01167.
[47] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: an Overview," 2020, *arXiv*. doi: 10.48550/ARXIV.2008.05756.
[48] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988, Oct. 2017. doi: 10.1109/iccv.2017.322.
[49] I. Sommerville, *Software Engineering*, 10th ed. Boston, MA, USA: Pearson, 2016.

[50] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY, USA: McGraw-Hill Education, 2019.
[51] M. K. Shah, B. Jactat, T. Yasui, and M. Ismailov, "Low-Fidelity Prototyping with Design Thinking in Higher Education Management in Japan: Impact on the Utility and Usability of a Student Exchange Program Brochure," *Education Sciences*, vol. 13, no. 1, p. 53, 2023, doi: 10.3390/educsci13010053.
[52] M. Hamburg and A. Roman, "Black-Box Testing for Practitioners: A Case of the New ISTQB Test Analyst Syllabus," *2025 IEEE Conference on Software Testing, Verification and Validation (ICST)*, pp. 634–645, Mar. 31, 2025. doi: 10.1109/icst62969.2025.10988991.

BIOGRAPHIES OF AUTHORS



Danny Matthew Saputra received the B.Eng. degree in Informatics from the Telkom University, Bandung, Indonesia, and the M.Sc. degree in Computer Science from the The University of Kentucky, Kentucky, United States. He is currently a permanent lecturer with the Department of Computer Systems, University Sriwijaya, Palembang, Indonesia. His current research interests include machine learning, data mining, image processing, computer architecture, and distributed systems. He can be contacted at email: danny.saputra@gmail.com.



Muhammad Azka Faridi is currently pursuing the B.Sc. degree in Informatics Engineering with the Sriwijaya University (UNSRI), Palembang, Indonesia. His current research interests include power artificial intelligence, computer architecture, parallel computing, and distributed systems. He can be contacted at email: azkafaridi@gmail.com.



Rizki Kurniati received the B.Sc. degree in Informatics from the Sriwijaya University, Palembang, Indonesia, and the M.Eng. degree Informatics in from the The Bandung Institute of Technology, Bandung, Indonesia. He is currently a permanent lecturer with the Department of Informatics Engineering, University Sriwijaya, Palembang, Indonesia. His current research interests include system information, machine learning, data mining, big data analytics, pattern recognition, image processing, and artificial intelligence. He can be contacted at email: rizkikurniatii@gmail.com.