

Smart Camera for Volcano Eruption Early Warning System Based on Faster R-CNN and YOLO

Hasanur Mohammad Firdausi^{1*}, Satrio Budi Utomo¹, Widjonarko¹, Gamma Aditya Rahardi¹

¹Department of Electrical Engineering Faculty of Engineering Universitas Jember

Jl. Kalimantan Tegalboto No 37 Sumbersari Jember Jawa Timur 68121

*E-mail Korespondensi : hasanur.firdausi@gmail.com

DOI: <https://doi.org/10.21107/rekayasa.v18i1.27372>

Submitted September 9th 2024, Accepted March 3rd 2025, Published April 7th 2025

Abstrak

Penelitian ini menggunakan dua algoritma deteksi objek, *Faster R-CNN* dengan tulang punggung ResNet50 dan YOLOv5, untuk mengembangkan sistem kamera cerdas untuk memantau aktivitas vulkanik. Model-model tersebut dilatih dan dievaluasi menggunakan rekaman CCTV dari Gunung Semeru, wilayah yang rawan letusan gunung berapi. Metrik kinerja utama seperti *Precision*, *Recall*, dan *mean Average Precision* (mAP) digunakan untuk mengevaluasi kinerja kedua model. Angka presisi tinggi untuk YOLOv5 dan *Faster R-CNN* menunjukkan bahwa keduanya bagus dalam menghindari positif palsu, yang penting untuk pemantauan gunung berapi. YOLOv5 memiliki presisi 83,2%, sedangkan *Faster R-CNN* adalah 84%. Namun, *recall* menunjukkan perbedaan yang lebih signifikan antara kedua model tersebut. *Faster R-CNN* memiliki *recall* sebesar 82%, yang berarti lebih baik dalam mendeteksi semua aktivitas gunung berapi yang relevan, meskipun itu berarti menangkap beberapa positif palsu. Variasi dalam kinerja dapat dikaitkan dengan desain masing-masing. YOLOv5 dirancang untuk mencapai deteksi cepat dan waktu nyata dengan memprediksi kotak pembatas dan probabilitas kelas secara bersamaan. Pendekatan ini meningkatkan kecepatan tetapi mungkin sedikit mengurangi ingatan. *Faster R-CNN* menggunakan proses dua tahap, cenderung lebih akurat tetapi bisa lebih lambat dan kurang fleksibel di berbagai ambang batas IoU. Ingatannya yang lebih tinggi berarti ia menangkap lebih banyak objek, berkontribusi pada mAP@50-95 yang lebih rendah karena ia dapat berjuang dengan objek yang tumpang tindih atau berukuran bervariasi.

Kata Kunci: deep learning, faster R-CNN, kebencanaan, smart camera, YOLO

Abstract

This research uses two object detection algorithms, *Faster R-CNN* with ResNet50 backbone and YOLOv5, to develop an intelligent camera system for monitoring volcanic activities. The models were trained and evaluated using CCTV footage from Mount Semeru, a region prone to volcanic eruptions. Key performance metrics such as *Precision*, *Recall*, and *mean Average Precision* (mAP) were used to evaluate the performance of both models. The high precision numbers for YOLOv5 and *Faster R-CNN* show they are good at avoiding false positives, which is essential for volcanic monitoring. YOLOv5 has a precision of 83.2%, while *Faster R-CNN* is 84%. However, *recall* shows a more significant difference between the two models. *Faster R-CNN* has a recall of 82%, meaning it is better at detecting all relevant volcanic activities, even if that means catching a few false positives. The variations in performance can be attributed to their respective designs. YOLOv5 is designed to achieve rapid, real-time detection by simultaneously predicting bounding boxes and class probabilities. This approach enhances speed but may slightly reduce recall. *Faster R-CNN* uses a two-stage process, tending to be more accurate but can be slower and less flexible across different IoU thresholds. Its higher recall means it catches more objects, contributing to its lower mAP@50-95 since it could struggle with overlapping or varying-sized objects.

Key words: deep learning, faster R-CNN, disaster, smart camera, YOLO

INTRODUCTION

Indonesia, located at the confluence of three major tectonic plates and consisting of several islands, is known for its dynamic metavolcanic belt, which causes periodic earthquakes and volcanic eruptions. The "Ring of Fire" refers to a chain of 129 currently active volcanoes along a subduction zone that stretches from western Sumatra to southern Java, Bali, NTB, NTT, Sulawesi, and Papua (Kusumasari, 2019). The Indonesian island of Java is primarily characterized by its significant volcanic activity, with 23 A-type volcanoes. The country has experienced 470 volcanic eruptions, accounting for 47% of all eruptions worldwide. Most of the activity is focused on the southern flank of Mount Merapi, a stratovolcano (Toar et al., 2021). In 2010, a significant eruption severely damaged 2,682 houses in the Special Region of Yogyakarta and 174 houses in Central Java. The volcanic eruption caused substantial disruption to public activities and

services in the surrounding areas, including residential areas, infrastructure, social structures, and many economic sectors. The chain of volcanoes in Java extends horizontally from west to east (Martinez & Hudayana, 2022).

Indonesia's vulnerability to natural disasters is a worrying issue, prompting many government and commercial institutions to study disasters in various regions. Advances in technology and research have been utilized to develop mitigation strategies or detect natural disasters. Rescue operations are carried out to minimize material and non-material losses, including damage to physical resources, public infrastructure, and human lives. Satellite imagery is used for cartography, assessing affected areas, and early warning systems to predict natural disasters such as volcanic eruptions (Cummins, 2017).

One of the prevention efforts made by the government is to identify the presence of cold Lahar heading toward the residents' environment using CCTV installed on the path often passed by cold Lahar from volcanic eruptions. In this study, we used the cold Lahar path of Mount Semeru. The installed CCTV has implemented image processing technology to detect the potential danger of cold Lahar along the path (Awaludin et al., 2012)..

Research that leads to computer vision for cold Lahar detection is still scarce. Therefore, researchers will compare several methods that have been proven reliable for application in CCTV-based early warning systems to detect cold Lahar to find a process suitable method. Based on the journal (Liu, 2018), from this study using Faster R-CNN for object detection, a precision value of 88.39% was obtained, compared to RFCN with a precision value of 83.54%. The journal (Hua & Tong, 2020) uses faster r-cnn for face detection and produces an mAP value of 81.33%. The journal (Wang et al., 2021; Yang et al., 2021) uses YOLO for underwater object detection; the results of model testing using video data produce an accuracy value of 97.59%, with a precision value of 0.88, a recall of 0.95, an F-1 score of 0.92, and an average IoU of 69.28%.

Based on the description above, the Volcanic Eruption Early Warning System requires a smart camera. This study will analyze the innovative camera system on the Volcanic Eruption Early Warning System tool to detect the presence of cold Lahar and hot clouds. This study will utilize the Faster Region-based Convolutional Neural Network (Faster R-CNN) and You Only Look Once (YOLO) approaches. This project aims to develop an intelligent camera system that adds the Faster R-CNN and YOLO methods to accurately identify and categorize the occurrence of hot clouds and extraordinary lahar floods. This study aims to reduce the loss of human life, especially in disaster-prone areas (KRB) vulnerable to the impacts of cold Lahar flows and hot clouds.

RESEARCH METHODS

Faster R-CNN

A CNN, short for Convolutional Neural Network, is a sophisticated type of neural network that employs convolutional structures, which are essential components of deep learning, to extract features from data (Kido et al., 2018; Priyadharshini & Judie Dolly, 2023; Yanagisawa et al., 2018; Yuhana et al., 2023). The Faster Region-based Convolutional Neural Network (Faster R-CNN) consists of two components. The first module of the system is a comprehensive convolutional network that produces regions (Ahmed et al., 2021; Ren et al., 2017). The Fast R-CNN detector utilizes these regions as its second module. The system functions as an item detection network. The Region Proposal Network (RPN) is an essential element of Faster R-CNN that effectively produces region proposals and allows for real-time model execution (Bappy & Roy-Chowdhury, 2016; Dong & Wang, 2016; Zamanidoost et al., 2023). The RPN method employs a "image-centric" approach to sample regions inside each image and classify them as positive or negative (Lin et al., 2019; Mansoor et al., 2019).

ResNet, also known as Residual Network, is a specialized neural network architecture used in deep learning applications, namely in the field of computer vision. This is a sophisticated convolutional neural network specifically created to tackle the difficulties encountered when training deep neural networks, such as problems with degradation and loss of gradient. To tackle this problem, ResNet provides a technique called deep residual learning (Al-Jawahry et al., 2023; Prasad et al., 2023; Tahir et al., 2021). Instead of assuming that each layer will precisely correspond to the desired base mapping, this network is specifically designed to acquire a residual mapping. This strategy simplifies the optimization of the residual mapping for the network compared to the original mapping that has not been referred to.

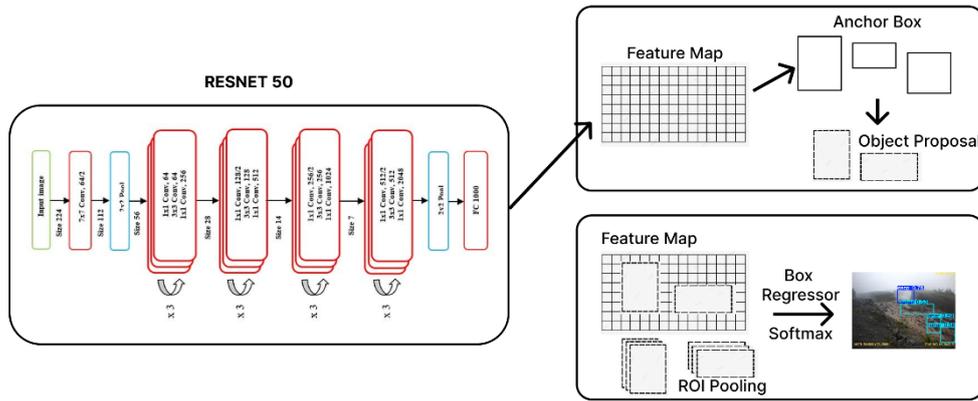


Figure 1. Architecture of Faster R-CNN Model with Resnet-50

The fundamental component of ResNet is a residual block comprising two or three layers. Each residual block combines its output with the output of the stacked layers and includes a shortcut connection for identity mapping (Anslam Sibi et al., 2024; Praveen et al., 2023). This shortcut connection does not introduce additional variables or increase computational complexity, enabling a fair comparison between simple and residual networks. ResNet architectures are commonly denoted by the number of layers they contain, such as ResNet-18, ResNet-50, ResNet-101, and ResNet-152 (M et al., 2022; Vuyyuru et al., 2023). The author employs ResNet50 in this research. ResNet50 utilizes a pioneering learning technique called residual learning. The ResNet50 architecture incorporates skip connections, often called shortcut connections, at each building block. These skip connections enable the network to acquire knowledge of identity mappings, enabling it to have a residual function that aids in the training process.

YOLOv5

You Only Look Once (YOLO) version 5 is a framework featuring an object detection algorithm created in 2020 by Glenn Jocher, a researcher and CEO of Ultralytics LLC. Yolov5 utilizes the PyTorch framework, which is implemented in the Python programming language. Yolov5 is a product of the Yolov3 implementation in the PyTorch framework, created by Glenn Jocher, as stated on the Roboflow website. Yolov5 consists of five pre-trained models that vary in size: YOLOv5s (the smallest), YOLOv5m, YOLOv5l, and YOLOv5x (the largest). YOLO is an exceptionally efficient object detection method, with YOLOv5 being one of its advanced versions (Jiang et al., 2022; Redmon et al., 2016). According to another study, the YOLO algorithm works by examining pixel blocks for their color and shape to identify objects or targets. It is specially trained to detect and classify damage to buildings (Biswas et al., 2024).

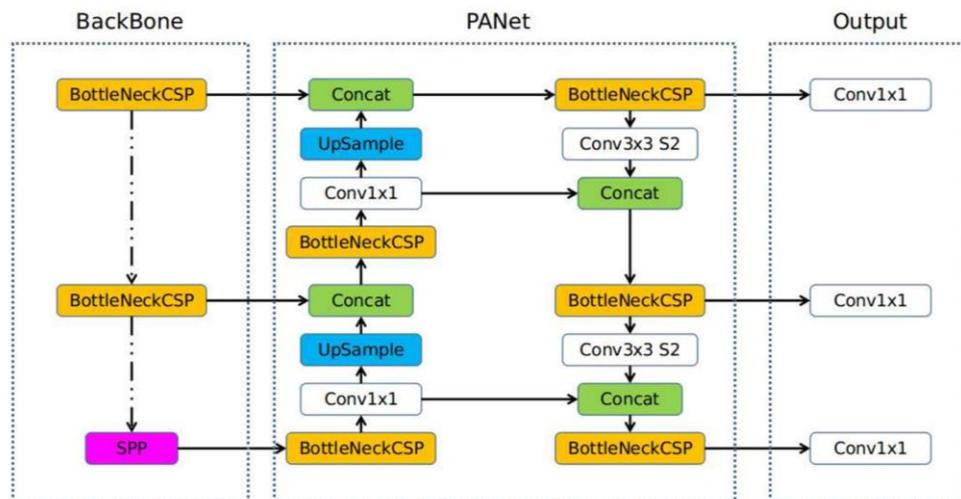


Figure 2. Architecture of YOLOv5

This approach employs a solitary neural network to concurrently manage the classification and localization of items in a picture. The technique employs bounding boxes for these tasks and offers significantly quicker processing performance than alternative methods, ranging from around 45 to 155 frames per second (Upadhyay et al., 2024). Unlike region-based convolutional neural networks, which analyze several regions of an image separately, YOLO uses a Fully Convolutional Neural Network (FCN) to process the entire image at once (Lu et al., 2023). This approach allows for a comprehensive analysis of the image features, thereby improving the algorithm's performance in object detection. YOLO (You Only Look Once) is well-regarded for its efficiency in detecting objects in natural images and managing new and unexpected scenarios by predicting bounding boxes based on a comprehensive assessment of the entire image. The YOLOv5 architecture seen in Figure 2 demonstrates that the process does not require substantial data. Thus, Yolo is the most advanced iteration that enhances the speed of object detection. The graphic contains information about cross-stage partial networks (CSP), partial pyramid pooling (SPP), convolutional layers (Conv), and concatenate functions (Concat).

Dataset

The datasets of cold Lahar and hot cloud were acquired from CCTV footage donated by the proprietor of the YouTube channel "CCTV SEMERU". This footage captures the Lahar flow from an elevated perspective, encompassing various elements beyond just the cold Lahar and Asap, such as surrounding terrain and vegetation. To ensure accurate representation of the cold Lahar's impact, frames are extracted at a rate of one frame per second, resulting in a dataset consisting of 292 images with a resolution of 1280x736 pixels. This temporal resolution is adequate as the cold Lahar's movement is relatively slow and changes in the terrain are minimal over short periods. The dataset is divided into training and testing subsets using a 90:10 split, yielding 266 images for training and 26 images for testing. Each image is annotated with regions of interest (ROI) for two classes: Asap and Lahar, facilitating comprehensive analysis and model training. This dataset provides a valuable resource for developing and evaluating object detection models, particularly in monitoring and analyzing volcanic activities and their environmental impact.

Annotation

In object detection-based segmentation, accurate annotations are essential as they define the areas of interest for both classification and localization tasks. We utilized the Roboflow Annotation tool to draw bounding boxes around Smoke and cold Lahar objects in a dataset of 292 images. The annotations follow the format $[x, y, h, w]$, where x and y indicate the center of the bounding box, while h and w represent its height and width. This process resulted in 838 total annotations, split into 269 for Asap and 569 for cold Lahar. The dataset was then divided, with 748 annotations (226 for Asap and 522 for cold Lahar) designated for training and validation, and 90 annotations (23 for Asap and 47 for cold Lahar) reserved for testing. Annotations are saved in YOLO Darknet format for YOLO model training and in COCO JSON format for Faster R-CNN model training. This systematic approach ensures comprehensive and effective object detection across diverse scenarios.



Figure 3. Example of annotated image

Training

The models are trained on a high-performance computing system equipped with an Nvidia GeForce 3090 GPU and 24GB of RAM. For object detection tasks, the YOLOv5s model is employed, which is a streamlined version of the YOLOv5 architecture, offering a balance between speed and accuracy. Simultaneously, Faster R-CNN employs a ResNet50 backbone, augmented by including the Region Proposal Network (RPN) in its architecture. Training is conducted with consistent hyperparameters across models: a learning rate of 0.001, a batch size of 6, and a training duration of 15 epochs. During the training process, validation is performed using a set of 41 images to assess model performance. The evaluation focuses on key metrics, including the loss function, which provides insights into the model's error minimization and predictive accuracy. Additionally, performance is monitored at each epoch to identify improvements and detect potential issues such as overfitting or underfitting. This rigorous approach ensures a thorough assessment of both YOLOv5s and Faster R-CNN models in terms of object detection accuracy and their generalization capabilities on new, unseen data. The insights derived from this analysis are instrumental in refining model performance and optimizing detection accuracy.

Metric Performance

In object detection, the fundamental metrics are Confidence and Intersection over Union (IoU). Confidence refers to the probability that an anchor box contains an object, as predicted by the classification component of the detection method. However, this metric is not utilized for performance evaluation, as it primarily determines the presence or absence of an object within an anchor box. IoU, however, quantifies the extent to which the anticipated bounding box and the ground truth bounding box overlap. The percentage is calculated by dividing the intersection of the predicted and ground truth boxes by their union and is given as a percentage. A similarity criteria of 0.5 is established, where bounding boxes with an Intersection over Union (IoU) greater than 0.5 are deemed valid detections.

For performance comparison, the following metrics are employed: True Positive (TP), False Positive (FP), Precision, Recall, Average Precision (AP), and mean Average Precision (mAP). TP refers to ground truth regions of interest (ROIs) that are accurately detected, typically with an IoU exceeding 0.5. FP denotes detected objects that are not part of the ground truth, also with an IoU greater than 0.5. True Negative (TN) is not used in this context because it does not pertain to the discussion of undetected ROIs, and Recall can be evaluated without considering False Negatives (FN). TN is excluded as it represents objects that should not be detected. Precision is a metric that quantifies the accuracy of a system in generating correct detections. It is computed by dividing the number of true positives by the sum of true and false positives and expressing the result as a percentage. Recall evaluates the system's capacity to correctly detect all pertinent items, measured as the ratio of true positives to the total number of ground truth objects (the sum of true positives and false negatives). The utilization of the following formulas achieves the quantification of these measures:

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots (1)$$

$$Recall = \frac{TP}{TP+TN} \dots\dots\dots (2)$$

Object detection performance is commonly evaluated using the universally recognized Average Precision (AP) metric [18]. Its comprehensive nature makes it preferable over precision and recall when comparing detectors. Average Precision (AP) is computed by taking the mean precision at each individual recall level. Interpolation is employed at various recall levels to minimize oscillations in the precision-recall curve.

$$P_{interp}(r) = \max_{r' \geq r} (r') \dots\dots\dots (3)$$

The average precision (AP) is calculated from the precision-recall curve, where the values range from 0 to 1. It is obtained by applying the integral formula.

$$AP = \int_0^1 p(r) dr \dots\dots\dots (4)$$

The approximation of this integral involves the summation of precision values at different threshold levels, with each value weighted by the corresponding change in recall.

$$AP = \sum_{i=1}^n (r_{i+1} - r_i) P_{interp}(r_{i+1}) \dots \dots \dots (5)$$

Ultimately, the mean Average Precision (mAP) is determined by taking the average of the Average Precision (AP) values for all K classes. This consolidated metric provides a measure of overall performance.

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

RESULTS AND DISCUSSION

This work utilized two advanced object detection algorithms, Faster R-CNN with ResNet50 backbone and YOLOv5, to create an intelligent camera system for monitoring volcanic activity. The primary objective was to detect cool Lahar and heated clouds. The models underwent training and evaluation using a dataset from CCTV footage near Mount Semeru, an area susceptible to volcanic eruptions.

Performance Evaluation of YOLOv5 Model

To get the most out of this model, we must use the correct parameter values, and these values cannot be measured with certainty, so we have to try them individually. Of course, if we change one by one, it will be very time-consuming. Therefore, researchers use a library called Optuna to process it with just one press, and the results can be seen in the following table.

Table 1. Hypertoptuna YOLOv5

Subject	mAP50-95	Batch size	Image size	Learning rate	Num epochs
1	0.313	16	512	0.0001	33
2	0.33	4	512	0.01	38
3	0.323	8	512	0.001	23
4	0.316	16	416	0.001	30
5	0.364	32	416	0.1	30
6	0.357	16	320	0.1	26
7	0.345	4	512	0.1	24
8	0.332	4	416	0.001	26
9	0.357	32	416	0.00001	23
10	0.346	16	416	0.1	21

From the optional hyperparameter process, the highest value is in the 5th test with an mAP (50-95) value of 0.365, so the parameters used are 32 for batch size, 416 for image size, learning rate 0.1, and epoch of 30 so that the overall value of the model test with these parameter values is as follows.

Table 2. YOLOv5 Model Testing

Class	Images	Instances	P	R	mAP50	mAP50-95
all	44	182	0.832	0.778	0.84	0.365

In this test, the Asap class has a very high accuracy value compared to the Lahar class, and this is due to the many backgrounds that have similarities with soil features and textures and the presence of Lahar images that are not correctly annotated so that the system can read them but are not included in the ground truth by researchers because Lahar images are not included in the substantial Lahar criteria.

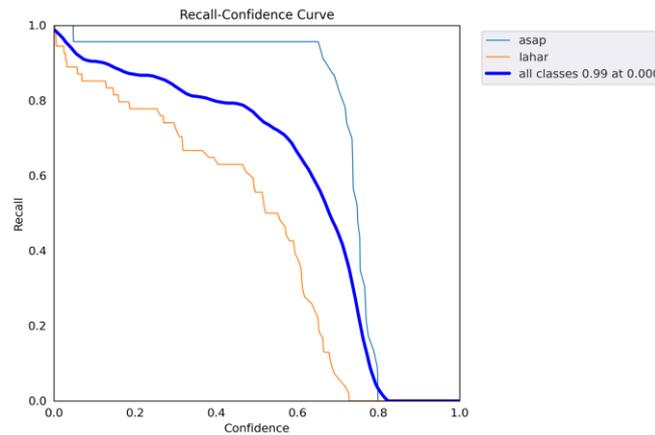


Figure 4. YOLOv5 Recall-Confidence Graph

This Recall-Confidence graph shows the relationship between recall and the model's confidence. Recall measures the ability of the model to find all relevant instances. The curve for the "Asap" class performs better than the "Lahar" class, as seen from the higher recall at various confidence levels. At high confidence (>0.8), the recall for the "Lahar" class drops off sharply, indicating that the model has a more challenging time detecting this class at high confidence levels. The "all classes" curve shows the model's overall performance, which is close to the performance of the "Asap" class due to its dominant performance. The model detects the "Asap" class more effectively than the "Lava." Performance drops significantly for the "Lahar" class at high confidence levels.

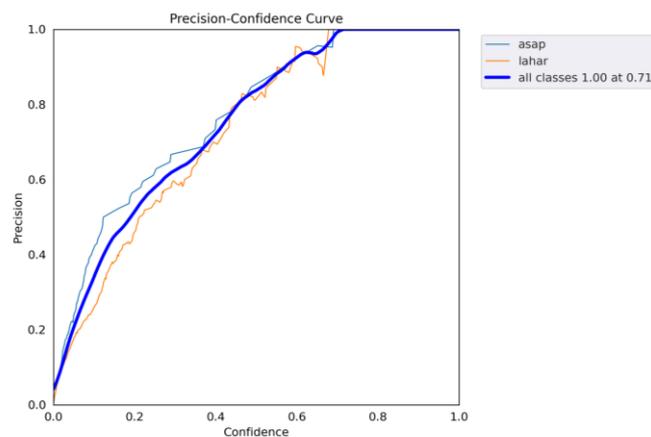


Figure 5. YOLOv5 Precision-Confidence Graph

This Precision-Confidence graph illustrates the relationship between precision and confidence. Precision measures the accuracy of the model's predictions of actual instances. The "Asap" class has more consistent precision than "Lava." At high confidence, the precision for all classes is nearly at its maximum (close to 1.0). The model has difficulty maintaining precision for the "Lahar" class at lower confidence. The model is more accurate in predicting the "Asap" class. Higher confidence levels help improve overall precision.

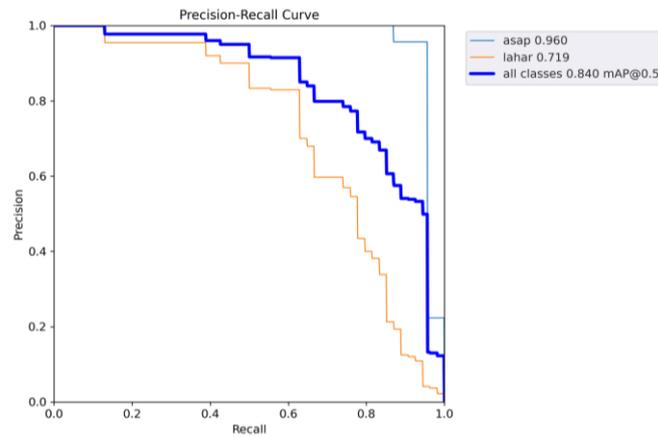


Figure 6. YOLOv5 Precision-Recall Graph

This Precision-Recall graph illustrates the relationship between precision and recall for each class. It shows the model's trade-off between finding all instances (recall) and making accurate predictions (precision). The "Asap" class shows near-perfect performance (high precision and recall, approaching 1.0). The "Lahar" class performs poorly, with lower precision and recall. The highest mAP (mean average precision) is achieved by the "Asap" class (0.960), while the "Lahar" class only reaches 0.719. The model performs very well in the "Asap" class but struggles with the "Lahar" class. The difference in performance between classes suggests possible data imbalance or difficulty detecting certain class features.

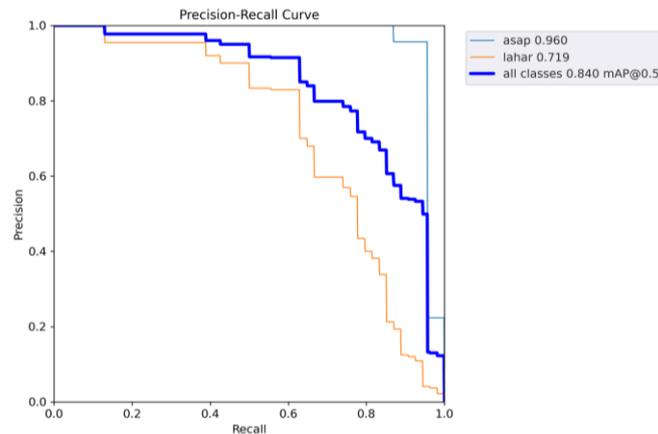


Figure 7. YOLOv5 F1-Confidence Graph

This F1-Confidence graph shows the relationship between F1 score (harmonization of precision and recall) and confidence level. The F1 score reflects the balance between precision and recall. The "Asap" class has a high F1 score at almost all confidence levels, indicating a good balance. The "Lahar" class has a lower F1 score, especially at higher confidence levels, suggesting an imbalance between precision and recall. The F1 score for "all classes" shows good average performance but is more affected by the "Asap" class. The model can balance precision and recall for the "Asap" class. The low F1 score for the "Lahar" class indicates areas for improvement, especially at high confidence. By class performance from all graphs, The model performs very well in the "Asap" class, with high precision, recall, and F1 score. Performance on the "Lahar" class is lower, indicating the model has difficulty detecting and predicting this class. By Confidence Level Effect from all graphs High confidence levels result in better precision, but can reduce recall, especially for lower-performing classes such as "Lahar." The model performs well overall, but there is scope for improving accuracy in certain classes.

Performance Evaluation of Faster R-CNN Model

The following model is Faster-RCNN, the model is superior with its high level of accuracy but has slower processing compared to Yolo. However, Faster R-CNN is the fastest model compared to its previous models, namely Fast R-CNN and R-CNN. Testing of this model must also be maximized by selecting the proper parameters like the previous Yolo test. Therefore the Optuna library is also used in the performance evaluation of this model.

Table 3. HypertOptuna Faster R-CNN

Subject	mAP50-95	Batch size	Learning rate	Num epochs
1	0.328	2	0.01	37
2	0.297	4	0.001	31
3	0.302	4	0.001	33
4	0.316	2	0.01	36
5	0.287	8	0.00001	33
6	0.252	8	0.001	30
7	0.158	2	0.0001	15
8	0.189	2	0.001	15
9	0.251	16	0.00001	20
10	0.288	4	0.0001	30

The optimal parameter value is obtained by searching for hyperparameter values using the Optuna library, reaching a mAP50-95 value of 0.328. More detailed results can be seen in the following table.

Table 4. Faster R-CNN Model Testing

Model	P	R	mAP50	mAP50-95
Faster RCNN	0.84	0.82	0.843	0.328

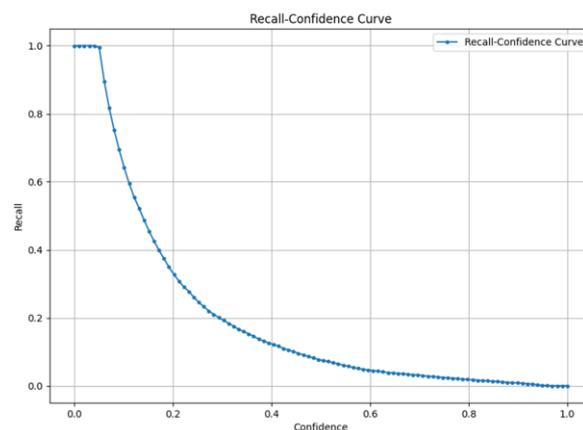


Figure 8. Faster R-CNN Recall-Confidence Graph

This Recall-Confidence graph shows the relationship between recall and confidence. At low confidence (approaching 0), the recall value approaches 1. This indicates that the model can detect almost all instances but with lower accuracy. As confidence increases (approaching 1), the recall value decreases drastically to near 0. This means that at high confidence levels, the model only detects a small portion of instances, but an increase in precision usually accompanies this. This graph indicates that there is a trade-off between recall and confidence. To achieve an optimal balance, choosing the right confidence threshold is necessary.

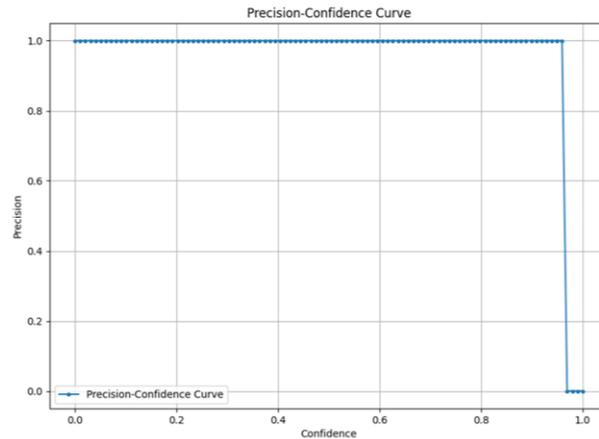


Figure 9. Faster R-CNN Precision-Confidence Graph

This Precision-Confidence graph shows the relationship between precision and confidence. The precision value is close to 1 in almost the entire confidence range. This indicates that the model almost always makes correct predictions at a certain confidence level. The sharp drop at the right end of the graph (confidence approaching 1) indicates that some instances are incorrectly predicted at very high confidence levels. The model shows an excellent ability to maintain precision at various confidence levels. The drop at the right end of the graph may occur due to the threshold being too tight.

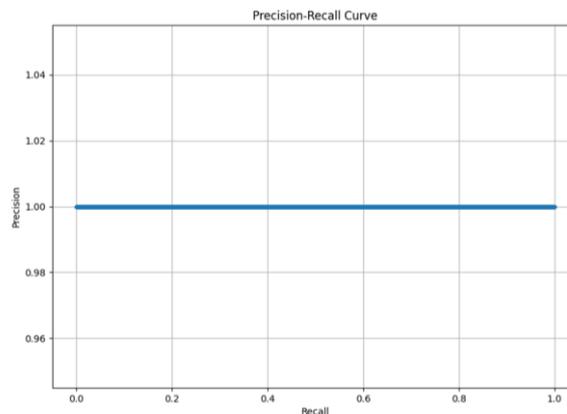


Figure 10. Fastre R-CNN Precision-Recall Graph

This Precision-Recall graph illustrates the relationship between precision and recall. The precision value remains constant throughout the recall range, close to 1. This indicates that the model has a consistent level of accuracy even though the number of instances detected increases. A flat graph suggests that the model is not affected much by the trade-off between precision and recall, indicating a highly reliable model. This graph shows excellent detection performance, where the model can maintain high precision without sacrificing recall. This F1-Confidence graph illustrates the relationship between the F1 score (harmony between precision and recall) and the confidence level. The highest F1 score is achieved at low to medium confidence, where the trade-off between precision and recall is balanced. The decrease in the F1 score at higher confidence indicates that despite the increase in precision, a significant reduction in recall affects the overall F1 score. To maintain an optimal F1 score, the selection of the confidence threshold must consider the balance between precision and recall.

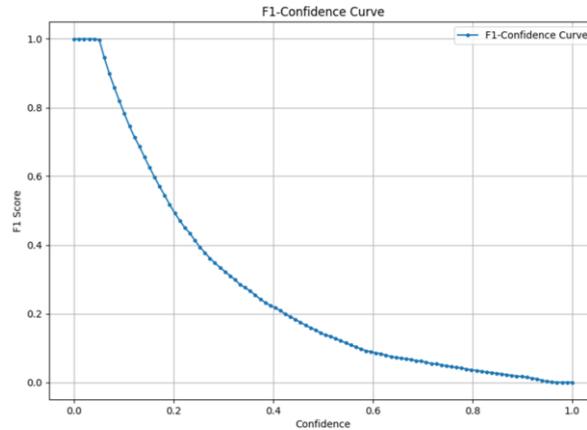


Figure 11. Fastre R-CNN F1-Confidence Graph

The model shows excellent performance in terms of precision, which is consistently high across the confidence range. However, the recall value tends to decrease drastically at high confidence levels, which impacts the F1 score. It is essential to adjust the confidence threshold for practical implementation to achieve the optimal balance between precision and recall.

Model Performance Comparison

Based on the data in the table, the YOLOv5 model with a Precision value of 83.2% shows that this model produces accurate positive predictions with a relatively low false positive rate. The Recall value of 77.8% is slightly lower when compared to the precision value, indicating that positive instances are not detected. The mAP50 value of 84% shows that YOLOv5's performance in detecting objects at the 50% IoU threshold is relatively high, indicating reliable detection accuracy. The mAP50-95 value of 36.5% suggests that the model's accuracy decreases when the evaluation is carried out at a tighter IoU. The Faster R-CNN model, with a Precision value of 84%, higher than the YOLOv5 model's precision value, shows that this model's ability is slightly better at minimizing false positives. The Recall value of 82% indicates that this model can detect more positive instances. The mAP50 value of 84.3% suggests that the detection performance at the 50% IoU threshold is excellent, even slightly higher when compared to the YOLOv5 model. The mAP50-95 value of 32.8% indicates that the average mAP value at a stricter IoU threshold is lower than the YOLOv5 model, indicating a decrease in accuracy for more complex scenarios.

Table 5. Comparison Data of Each Model

Methods	Test Set			
	Precision	Recall	mAP 50	mAP50-95
Yolo v5	0.832	0.778	0.84	0.365
Faster R-CNN	0.84	0.82	0.843	0.328



Figure 12. YOLOv5 detection results

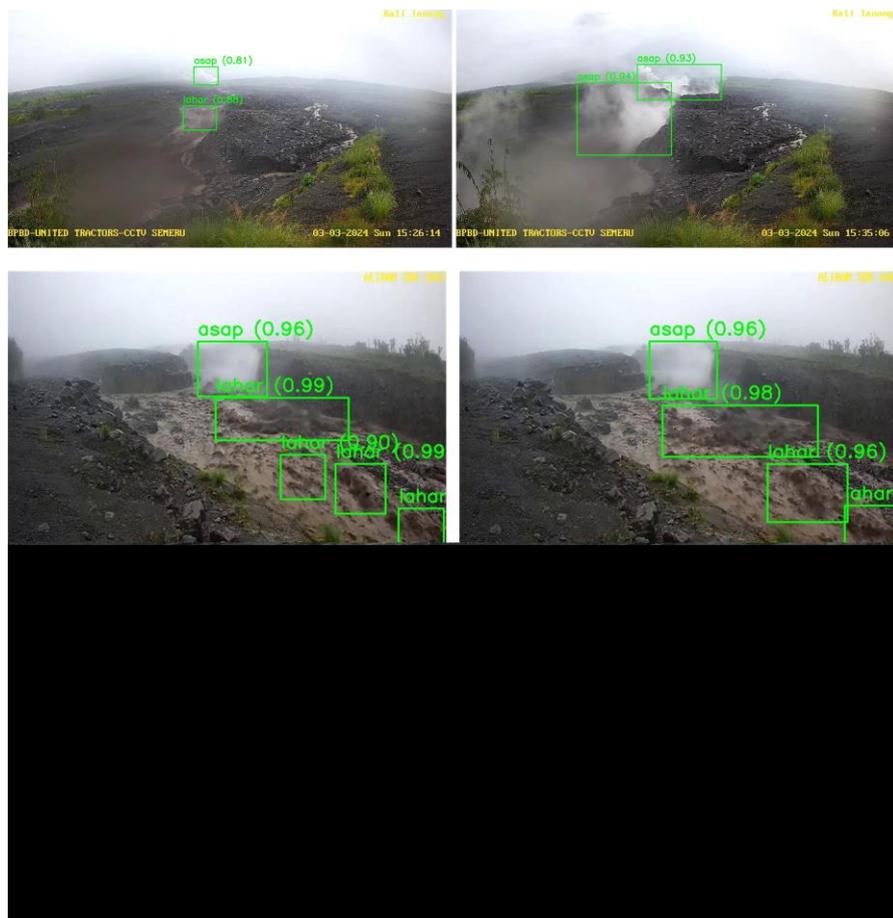


Figure 13. Faster R-CNN detection result

CONCLUSIONS

YOLOv5 and Faster R-CNN have their respective advantages in detecting objects. YOLOv5 shows stable performance with a mAP50-95 value of 36.5%, which is higher than Faster R-CNN thus, it is superior in detection at various IoU levels. However, Faster R-CNN has higher precision (84%) and recall (82%) values, indicating more accurate and comprehensive detection capabilities at an IoU threshold of 50%. However, Faster R-CNN's performance decreases in evaluations with stricter IoUs, as shown by the mAP50-95 value of 32.8%. Therefore, YOLOv5 is more suitable for applications that balance speed, stability, and good performance at various IoU levels. Meanwhile, Faster R-CNN is more recommended for applications that prioritize high accuracy in detecting objects with a lower IoU threshold, although at the cost of lower performance in stricter evaluations. Model selection should be adjusted to the specific needs and data characteristics of the application being implemented.

REFERENCES

- Ahmed, K., Ghareh Mohammadi, F., Matus, M., Shenavarmasouleh, F., Pereira, L., Zisis, I., & Amini, M. H. (2021). Towards Real-Time House Detection in Aerial Images Using Faster Region-Based Convolutional Neural Network. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.3994191>
- Al-Jawahry, H. M., Hussein, A. H. A., Sunil, G., Alkhafaji, M. A., & Pareek, P. K. (2023). Faster Region-Convolutional Neural Network with ResNet50 for Video Stream Object Detection. *2023 3rd International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, 1–6. <https://doi.org/10.1109/ICMNWC60182.2023.10435864>
- Anslam Sibi, S., Sivamohan, S., Prabhakaran, S., & Albin, S. B. (2024). The Resnet-50 Revolution: Leveraging Transfer Learning for Malaria Diagnosis. *2024 2nd International Conference on Artificial Intelligence and Machine Learning Applications Theme: Healthcare and Internet of Things (AIMLA)*, 1–5. <https://doi.org/10.1109/AIMLA59606.2024.10531528>
- Awaludin, L., Harjoko, A., & Sumiharto Raden. (2012). Pemrosesan Video Pendeteksi Kecepatan dan Ketinggian Aliran Lahar Dingin Pendukung Sistem Peringatan Dini. *IJEIS*, 2, 187–198.
- Bappy, J. H., & Roy-Chowdhury, A. K. (2016). CNN based region proposals for efficient object detection. *2016 IEEE International Conference on Image Processing (ICIP)*, 3658–3662. <https://doi.org/10.1109/ICIP.2016.7533042>
- Biswas, S., Nandy, A., Naskar, A. K., & Saw, R. (2024). Real time Gesture Recognition using Improved YOLOv5 Model. *2024 11th International Conference on Signal Processing and Integrated Networks (SPIN)*, 328–333. <https://doi.org/10.1109/SPIN60856.2024.10511787>
- Cummins, P. R. (2017). Geohazards in Indonesia: Earth science for disaster risk reduction – introduction. *Geological Society, London, Special Publications*, 441(1), 1–7. <https://doi.org/10.1144/SP441.11>
- Dong, P., & Wang, W. (2016). Better region proposals for pedestrian detection with R-CNN. *2016 Visual Communications and Image Processing (VCIP)*, 1–4. <https://doi.org/10.1109/VCIP.2016.7805452>
- Hua, W., & Tong, Q. (2020). Research on Face Expression Detection Based on Improved Faster R-CNN. *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, 1189–1193. <https://doi.org/10.1109/ICAICA50127.2020.9182525>
- Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A Review of Yolo Algorithm Developments. *Procedia Computer Science*, 199, 1066–1073. <https://doi.org/https://doi.org/10.1016/j.procs.2022.01.135>
- Kido, S., Hirano, Y., & Hashimoto, N. (2018). Detection and classification of lung abnormalities by use of convolutional neural network (CNN) and regions with CNN features (R-CNN). *2018 International Workshop on Advanced Image Technology (IWAIT)*, 1–4. <https://doi.org/10.1109/IWAIT.2018.8369798>
- Kusumasari, B. (2019). Natural Hazards Governance in Indonesia. In *Oxford Research Encyclopedia of Natural Hazard Science*. Oxford University Press. <https://doi.org/10.1093/acrefore/9780199389407.013.234>

- Lin, Q., Zhao, J., Tong, Q., Zhang, G., Yuan, Z., & Fu, G. (2019). Cropping Region Proposal Network Based Framework for Efficient Object Detection on Large Scale Remote Sensing Images. *2019 IEEE International Conference on Multimedia and Expo (ICME)*, 1534–1539. <https://doi.org/10.1109/ICME.2019.00265>
- Liu, Y. (2018). An Improved Faster R-CNN for Object Detection. *2018 11th International Symposium on Computational Intelligence and Design (ISCID), 02*, 119–123. <https://doi.org/10.1109/ISCID.2018.10128>
- Lu, Z., Ding, L., Wang, Z., Dong, L., & Guo, Z. (2023). Road Condition Detection Based on Deep Learning YOLOv5 Network. *2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI)*, 497–501. <https://doi.org/10.1109/ICETCI57876.2023.10176545>
- M, P., Sreekumar, S., & S, A. (2022). Detection of Covid-19 from the Chest X-Ray Images: A Comparison Study between CNN and Resnet-50. *2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)*, 1–7. <https://doi.org/10.1109/MysuruCon55714.2022.9972488>
- Mansoor, A., Porras, A. R., & Linguraru, M. G. (2019). Region Proposal Networks with Contextual Selective Attention for Real-Time Organ Detection. *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, 1193–1196. <https://doi.org/10.1109/ISBI.2019.8759480>
- Martinez, A., & Hidayana, B. (2022). From Transmigrasi to Relokasi. *Archipel*, 104, 57–74. <https://doi.org/10.4000/archipel.3004>
- Prasad, J. V. D., Chandra, M. K., Akash, J., & Vivek, K. (2023). Coconut Tree Detection in Coastal Areas with Fast-RCNN Using Resnet-50. *2023 Global Conference on Information Technologies and Communications (GCITC)*, 1–4. <https://doi.org/10.1109/GCITC60406.2023.10426612>
- Praveen, T. N. V. S., Sivathmika, D., Jahnvi, G., & Bolledu, J. (2023). An In-depth Exploration of ResNet-50 for Complex Emotion Recognition to Unraveling Emotional States. *2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, 1–5. <https://doi.org/10.1109/InCACCT57535.2023.10141774>
- Priyadharshini, G., & Judie Dolly, D. R. (2023). Comparative Investigations on Tomato Leaf Disease Detection and Classification Using CNN, R-CNN, Fast R-CNN and Faster R-CNN. *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 1, 1540–1545. <https://doi.org/10.1109/ICACCS57279.2023.10112860>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788. <https://doi.org/10.1109/CVPR.2016.91>
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- Tahir, H., Shahbaz Khan, M., & Owais Tariq, M. (2021). Performance Analysis and Comparison of Faster R-CNN, Mask R-CNN and ResNet50 for the Detection and Counting of Vehicles. *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, 587–594. <https://doi.org/10.1109/ICCCIS51004.2021.9397079>
- Upadhyay, D., Manwal, M., Kukreja, V., & Sharma, R. (2024). A Fine-Tuned YOLOv5 and Exception Model for Oral Cancer Detection. *2024 5th International Conference for Emerging Technology (INCET)*, 1–5. <https://doi.org/10.1109/INCET61516.2024.10592942>
- Vuyuru, L., Nalluri, S., Vempatapu, J., & Thopuri, R. babu. (2023). Air Pollution Detection using Resnet-50. *2023 7th International Conference on Computing Methodologies and Communication (ICCMC)*, 845–849. <https://doi.org/10.1109/ICCMC56507.2023.10083837>

- Wang, Y., Liu, J., Yu, S., Wang, K., Han, Z., & Tang, Y. (2021). Underwater Object Detection based on YOLO-v3 network. *2021 IEEE International Conference on Unmanned Systems (ICUS)*, 571–575. <https://doi.org/10.1109/ICUS52573.2021.9641489>
- Yanagisawa, H., Yamashita, T., & Watanabe, H. (2018). A study on object detection method from manga images using CNN. *2018 International Workshop on Advanced Image Technology (IWAIT)*, 1–4. <https://doi.org/10.1109/IWAIT.2018.8369633>
- Yang, H., Liu, P., Hu, Y., & Fu, J. (2021). Research on underwater object recognition based on YOLOv3. *Microsystem Technologies*, 27. <https://doi.org/10.1007/s00542-019-04694-8>
- Yuhana, U. L., Edo, G., & Syarif, H. (2023). Enhancement of Blurred Indonesian License Plate Number Identification Using Multi-Scale Information CNN. *2023 3rd International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON)*, 1–6. <https://doi.org/10.1109/SMARTGENCON60755.2023.10442912>
- Zamandoost, Y., Alami-Chentoufi, N., Ould-Bachir, T., & Martel, S. (2023). Efficient Region Proposal Extraction of Small Lung Nodules Using Enhanced VGG16 Network Model. *2023 IEEE 36th International Symposium on Computer-Based Medical Systems (CBMS)*, 483–488. <https://doi.org/10.1109/CBMS58004.2023.00266>