UAV Imagery-Based Potential Forest Fire Detection Using YOLOv10

Jasman Pardede[1,*], Muhamad Rifki Pratama[2], Rizka Milandga Milenio[3]

[1,2,3]*Department of Informatics, Institut Teknologi Nasional Bandung, Jl. PHH Mustofa No. 23, Bandung 40124, Indonesia*

(Received: August 1, 2025; Revised: October 1, 2025; Accepted: December 24, 2025; Available online: January 31, 2026)

Abstract

Forest fire mitigation requires an early detection system that is both fast and reliable. This study presents a real-time potential forest fire detection system based on UAV imagery using the YOLOv10 object detection model. The main objective is to enhance the accuracy of detecting fire and smoke in aerial imagery and to minimize false alarms through hyperparameter optimization and data balancing strategies. The dataset used was compiled from Roboflow Universe and Kaggle, consisting of two object classes: fire and smoke, with a slight class imbalance (1329 fire and 1024 smoke). In total, 1,691 annotated images were used, covering various lighting conditions, smoke densities, camera angles, and geographic backgrounds, and were divided into training, validation, and test sets with a ratio of approximately 75:15:10. To address the class imbalance and visual variability, data augmentation techniques such as rotation, flipping, brightness adjustment, and noise addition were applied, along with loss weighting to improve learning performance for the minority smoke class. Model training was conducted using 24 hyperparameter configurations combining six optimizers, two batch sizes, and two learning rates. The best hyperparameters are NAdam optimizer, batch_size 24, and learning_rate 0.001. The best performance of accuracy, precision, recall, F1-score, mean IoU, and mAP were achieved at 0.879, 0.8705, 0.8575, 0.863, 0.7373, and 0.870, respectively. Real-time testing using a DJI Mini 4 Pro UAV with RTMP livestream input demonstrated stable and responsive detection, displaying bounding boxes, class labels, confidence scores, and a “POTENTIAL FOREST FIRE” indicator when both fire and smoke were detected simultaneously. These findings confirm that integrating UAV and YOLOv10 technologies provides an effective and adaptive approach for real-time early detection of potential forest fires.

Keywords: YOLOv10, Object Detection, Potential Forest Fire, UAV, Aerial Imagery, Real-Time Detection, Hyperparameter

1. Introduction

Forest fires are one of the most serious environmental disasters, causing extensive forest loss, ecosystem damage, and increasing carbon monoxide (CO) emissions by up to 30% in the atmosphere [1], [2]. CO binds to hemoglobin about 240 times more strongly than oxygen, forming carboxyhemoglobin (COHb) that inhibits oxygen transport and leads to tissue hypoxia, which can cause permanent cell damage or even death [3]. Considering these severe environmental and health impacts, an early and accurate forest fire detection system is urgently needed. Unfortunately, conventional monitoring methods such as manual patrols and watchtowers still face major limitations in coverage, efficiency, and real-time responsiveness, making them less reliable for early detection and rapid response [4].

Recent advances in technology have introduced Unmanned Aerial Vehicles (UAVs) as a more efficient and flexible solution for forest monitoring [1]. UAVs, commonly known as drones, can capture high-resolution images and transmit live video in real-time, enabling surveillance across large and hard-to-reach forest areas [5], [6]. Their mobility, low operational cost, and ability to adapt to changing conditions make UAVs well suited for use in forest fire detection systems, especially for early warning and mitigation purposes.

To improve the accuracy and efficiency of UAV-based detection systems, deep learning methods such as You Only Look Once (YOLO) have gained widespread use in recent years [7]. YOLO is known for its ability to perform real-time object detection with a strong balance between speed and accuracy. Previous studies comparing YOLO with other architectures, such as Faster-RCNN, R-FCN, and SSD have shown that YOLOv3 achieves high detection precision of 83.7% and operates at 28 FPS, meeting real-time detection requirements [7], [8]. However, earlier YOLO versions still

***Corresponding author: Jasman Pardede (jasman@itenas.ac.id)**

DOI: https://doi.org/10.47738/jads.v7i1.1052

struggle with challenges related to high-resolution aerial imagery and variations in smoke density, lighting, and fire intensity, which are often encountered in UAV-based applications.

To overcome these limitations, the latest YOLOv10 architecture introduces several improvements in accuracy, latency, and detection efficiency [9], [10]. YOLOv10 implements a Dual Label Assignment mechanism consisting of a One-to-Many approach during training and a One-to-One approach during inference. This mechanism helps the model learn more comprehensively during training while producing cleaner and more efficient detections during inference without the need for Non-Maximum Suppression (NMS). In addition, hyperparameter tuning plays a key role in optimizing the model's performance parameters such as optimizer type, batch size, and learning rate have been shown to significantly affect detection results [11].

Based on these advancements, this study proposes a real-time UAV-based forest fire potential detection system using the YOLOv10 model. In this system, fire potential is defined when both fire and smoke are detected simultaneously. By integrating UAV imagery and the optimized YOLOv10 model, the proposed system aims to enhance detection accuracy, responsiveness, and reliability for early forest fire warning and prevention in real-world field conditions.

2. Method

2.1. Forest Fire

Forest fires are a phenomenon of uncontrolled burning of natural vegetation, triggered by a combination of natural factors such as lightning and human activities such as land clearing and changes in land use [12]. One of the main consequences of forest fires is the release of large amounts of carbon emissions including carbon dioxide (CO_2), methane (CH_4), and nitrogen oxides (NO_x) which have a significant contribution to increasing the greenhouse effect and accelerating global climate change through direct effects on atmospheric radiation and warming of the earth's surface [13]. In addition, forest fires have ecological impacts such as loss of biodiversity, degradation of soil structure, and disruption of local and global hydrological cycles [14].

Given their broad impacts, it is important to understand how wildfires are classified and defined in a global context. Classified forest fire as part of a wildfire that is unplanned, difficult to control, and has a major impact on ecosystems and human life [18]. In [19] emphasized that standardization of terms such as wildfire, forest fire, and bushfire in global policy is essential, given the different connotations and legal implications across countries, where harmonization of these definitions is key to strengthening mitigation, early detection systems, and disaster response internationally, especially amidst the increase in extreme fires due to climate change.

2.2. Unmanned Aerial Vehicle

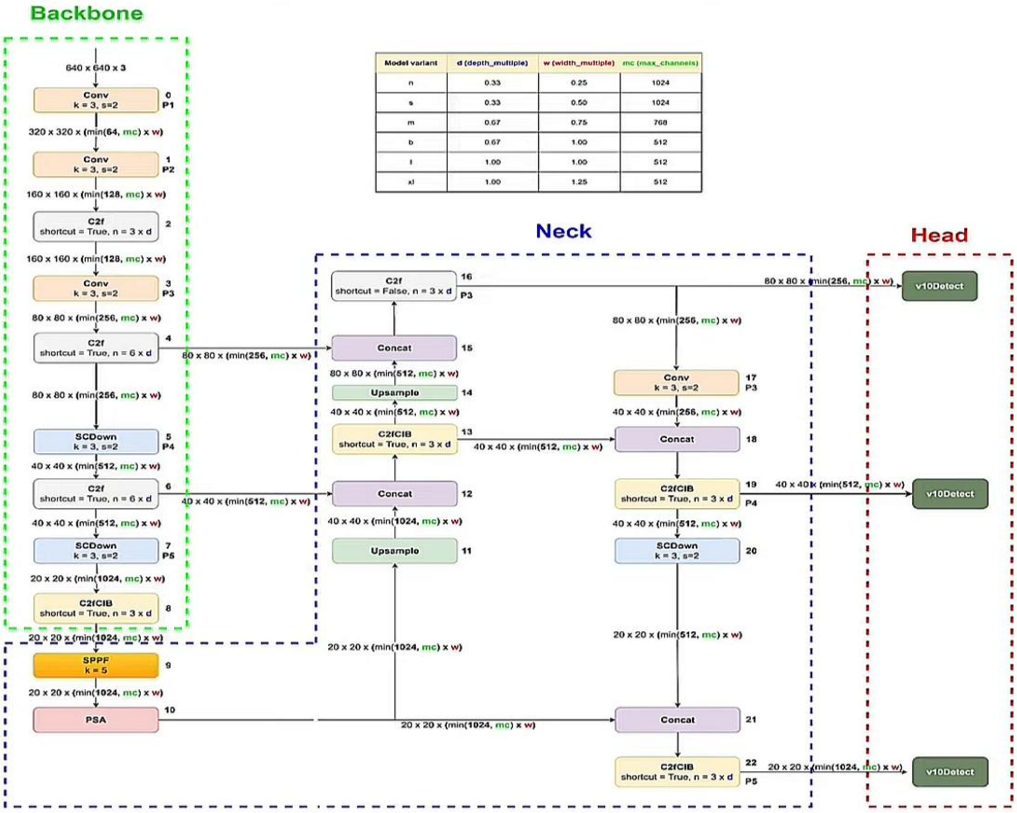
UAV or drone is an aerial-vehicles that can remain in flight without requiring a human operator on board, controlled remotely or autonomously, and used in a variety of scenarios such as surveillance, mapping, and military and civilian applications [17]. UAV generally consist of main components such as an airframe, propulsion system, navigation module, and communication unit that allow autonomous operation or remote control without the presence of a pilot on board [18]. The main advantages of UAVs lie in their high mobility capabilities, maneuverability in narrow areas, and efficiency in collecting real-time visual and thermal data [19].

Furthermore, several studies highlight the advantages of UAVs in terms of design and technical capabilities in various operational conditions. UAVs are designed to be able to operate in challenging and extreme environments by utilizing aerodynamic efficiency, functional flexibility, and advanced automatic control systems that play an important role in maintaining stability and performance during flight missions [20]. The rotorcraft-type UAVs have been widely used in civil engineering as a survey and monitoring tool for construction projects that are efficient in terms of time and cost [21]. UAV technology is capable of producing high-resolution aerial imagery that supports accurate spatial analysis, thus providing an important basis for data-based decision-making processes, especially in the fields of precision agriculture and environmental monitoring [22]. Along with the advancement of technologies such as automatic navigation systems, Global Positioning System (GPS), and Inertial Measurement Units (IMU), UAVs are evolving into key components in the integration of artificial intelligence and Internet of Things (IoT)-based systems, which increase efficiency, autonomy, and intelligence in various industrial applications [23].

2.3. YOLOv10

YOLOv10 (You Only Look Once version 10) is one of the latest object detection models in the YOLO series, designed to improve the accuracy and efficiency of real-time object recognition. This model is an evolution of the previous version, focusing on architectural innovation and significant performance improvements. YOLOv10 is equipped with various new features that are crucial for improving detection performance [12].

Key innovative features of YOLOv10 include Dual Label Assignments that enable training without NMS and improve prediction efficiency; Partial Self-Attention (PSA) for efficient global representation learning; use of Large Kernel Convolution that expands the receptive field to support small object detection; Spatial-Channel Decoupled Downsampling for efficiency without losing detail; and Rank-Guided Block Design with Compact Inverted Block (CIB) to reduce parameter redundancy and improve computational efficiency. An illustration of the YOLOv10 architecture is shown in figure 1.



Model variant	d (depth_multiple)	w (width_multiple)	mc (max_channels)
n	0.33	0.25	1024
s	0.33	0.50	1024
m	0.67	0.75	768
b	0.67	1.00	512
l	1.00	1.00	512
xl	1.00	1.25	512

Backbone
Neck
Head
640 x 640 x 3
Conv k = 3, s=2 0 P1
320 x 320 x (min(64, mc) x w)
Conv k = 3, s=2 1 P2
160 x 160 x (min(128, mc) x w)
C2f shortcut = True, n = 3 x d 2
Conv k = 3, s=2 3 P3
80 x 80 x (min(256, mc) x w)
C2f shortcut = True, n = 6 x d 4
SCDown k = 3, s=2 5 P4
40 x 40 x (min(512, mc) x w)
C2f shortcut = True, n = 6 x d 6
SCDown k = 3, s=2 7 P5
20 x 20 x (min(1024, mc) x w)
C2fCIB shortcut = True, n = 3 x d 8
SPPF k = 5 9
PSA 10
Upsample 11
Concat 12
C2fCIB shortcut = True, n = 3 x d 13
Upsample 14
Concat 15
C2f shortcut = False, n = 3 x d 16 P3
Conv k = 3, s=2 17 P3
Concat 18
C2fCIB shortcut = True, n = 3 x d 19 P4
SCDown k = 3, s=2 20
Concat 21
C2fCIB shortcut = True, n = 3 x d 22 P5
v10Detect

Figure 1. YOLOv10 Architecture.

The YOLOv10 architecture is fundamentally composed of three main components: backbone, neck, and head. The backbone is responsible for the main feature extraction using various convolution blocks (such as Conv, C2f, C2fCIB, and SCDown) to process and extract important features. Next, the neck receives and combines features from the backbone through blocks such as SPPF (for multi-scale feature representation) and PSA (for efficient global modeling). Finally, the head is responsible for the final prediction, including object marking with bounding boxes and class determination, involving the v10Detect main block and the Dual Label Assignment strategy for efficient training and inference. The overall architecture is designed to achieve an optimal balance between detection accuracy and real-time processing efficiency. Despite the promising improvements of YOLOv10, its application in potential forest fire detection remains limited, especially in real-time UAV-based scenarios. Most previous studies relied on earlier YOLO versions (e.g., YOLOv3–YOLOv9) or static image datasets without considering real-time video input and hyperparameter optimization for aerial imagery. Therefore, this research explores the implementation of YOLOv10 for fire and smoke detection using UAV livestream data, focusing on achieving high accuracy and real-time responsiveness through systematic hyperparameter tuning.

2.4. Evaluation Metrics

To evaluate the performance of the YOLOv10 object detection model in this study, several standard evaluation metrics were used including Confusion Matrix, Intersection over Union (IoU), and Mean Average Precision (mAP). These metrics assess both the classification accuracy and the spatial precision of the model in detecting fire and smoke objects.

Confusion matrix is a table used to evaluate the performance of classification algorithms in machine learning [25]. This matrix compares the labels predicted by the model with the actual labels and generally consists of four main components in the case of classification, namely True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) [25]. From this matrix, other evaluation metrics such as accuracy, recall, precision, and F1-score can be calculated [25]. Equations (1) to (4) are used for performance measurement using confusion matrix.

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \tag{1}$$

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \tag{2}$$

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{3}$$

$$\text{F1 Score} = 2 \times (\text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall}) \tag{4}$$

Intersection over Union (IoU) is a key evaluation metric in object detection tasks that is used to measure how well the model's bounding box predictions match the ground truth in the image [26]. The IoU value is calculated by dividing the area of overlap between the prediction and the ground truth (Area of Intersection) by the combined total area of both (Area of Union) as shown in Equation (5).

$$\text{IoU} = (\text{Area of Intersection})/(\text{Area of Union}) \tag{5}$$

Figure 2 shows an illustration of the IoU concept, measuring how close the prediction of an object detection model is to the actual position (ground truth). In this figure, the green box represents the ground truth and the red box represents the bounding box of the prediction result. The yellow area shows the Area of Intersection, which is the overlapping part between the prediction and the ground truth. While the light blue area is the Area of Union, which is the total combination of the two boxes.

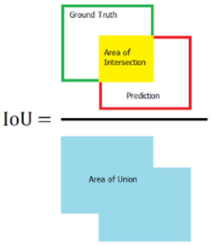


Figure 2. Illustration of IoU calculation between prediction bounding box and ground truth.

The IoU value is in the range of 0 to 1, where the higher the IoU value (closer to 1), the better the model is at predicting the location of an object accurately, conversely a low IoU value indicates that the model's prediction is far from the ground truth [27]. In this study, the Intersection over Union (IoU) calculation was performed using a Python script on 254 validation data. The prediction bounding box was obtained from the inference results of the YOLOv10 model, while the ground truth was taken from the annotation file in YOLO format and converted to absolute coordinates. For each ground truth, the highest IoU value was searched for from the available predictions, then averaged per image. The IoU value per image was then averaged to obtain the overall mean IoU, which reflects the spatial accuracy of the model on the validation dataset.

Mean Average Precision (mAP) is a key evaluation metric in object detection tasks used to assess how well a model like YOLO (You Only Look Once) can detect and classify objects [28]. mAP is calculated by first determining the Average Precision (AP) for each class based on the Precision-Recall Curve, and then averaging all of those APs. mAP takes into account the positional accuracy (bounding box) as well as the classification of the detected object. In the context of YOLO, mAP is very important because YOLO prioritizes speed and efficiency, so mAP is the main measure of prediction quality [28]. In this study, the mAP value was calculated automatically by the YOLOv10 training script during the training and validation process according to the built-in evaluation standards of the Ultralytics framework.

To find out the Average Precision (AP) value of a class, it is necessary to know the precision and recall values. After that, the precision value is plotted against the recall value to obtain a zigzag pattern. To calculate the AP value, the zigzag plot results will be smoothed, where the precision value (p) at a recall (r) will be equated with the maximum precision value at the next recall (r'), or written using Equation (6).

$$P_{inter}(r_{n+1}) = \max p(r'); r' \geq r_{n+1} \tag{6}$$

Next, to get the AP value is obtained by calculating the curve area at the unique recall value, namely when the maximum precision value falls. So that the calculated area is the area where the zigzag is smoothed, which is called the Area Under Curve (AUC). Furthermore, the AP value and the mAP are obtained using Equation (7) and Equation (8).

$$AP = \sum (r_{n+1} - r_n) p_{inter}(r_{n+1}) \tag{7}$$

$$mAP = \sum_{i=1}^{N} \frac{AP(i)}{N} \times 100\% \tag{8}$$

2.5. Dataset

The dataset used in this study was compiled from multiple open sources, including The Wildfire Dataset on Kaggle [31], and additional collections from Roboflow Universe, resulting in a total of 1,691 labeled images. The data were divided into 1,268 training, 254 validate, and 169 testing (approximately 75:15:10 ratio). There are two object classes: fire (1329 annotations) and smoke (1024 annotations) indicating a moderate class imbalance between the two categories. This imbalance was addressed through data augmentation and balanced training strategies to ensure that the model could generalize well across both classes.

To ensure better representation and model generalization, the dataset includes diverse visual conditions such as various illumination levels, smoke densities, viewing angles, and environmental backgrounds (forest, grassland, urban edges). Since the number of smoke samples was relatively lower than fire samples, data augmentation techniques such as random rotation, horizontal and vertical flipping, brightness and contrast adjustments, and Gaussian noise addition were applied. In addition, loss weighting was employed to reduce bias toward majority classes during training. This combination of multi-source data, augmentation, and balancing strategies enhances the robustness of the YOLOv10 model in detecting smoke-only or early-stage fire scenarios which are critical for effective early warning and wildfire prevention in real-world applications.

2.6. System Design

The system proposed for UAV image-based potential forest fire detection using YOLOv10 is shown in figure 3. The first process is data preparation, including collecting UAV image datasets and public datasets with various conditions, including the presence of fire spots, smoke, and areas without fires. Each image is then manually labeled using Roboflow to mark fire and smoke objects with bounding boxes according to the YOLO format. After that, the data goes through a preprocessing stage, which includes adjusting the image resolution, normalization, and data augmentation to increase the variation and ability of the model to recognize patterns.

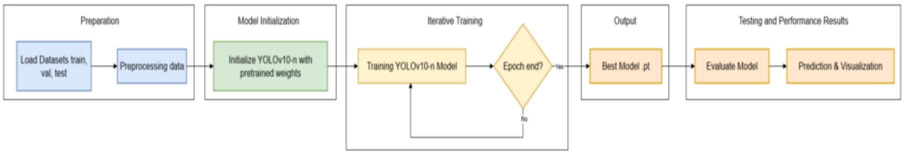


Figure 3. System Block Diagram

The YOLOv10 model training is carried out by utilizing its three main components: Backbone for feature extraction, Neck for multi-scale feature fusion, and Head for bounding box prediction, confidence score, and object classification. This training process takes place iteratively while monitoring the training loss and validation loss to prevent overfitting or underfitting. Finally, after training is complete, the model performance is measured using test data with evaluation metrics such as Confusion Matrix, IoU, and mAP.

3. Result

3.1. Graphical User Interface Application Configuration

The Graphical User Interface (GUI) of the YOLOv10-based potential fire detection system is designed using the Tkinter library to ensure ease of use for operators in the field. At the top of the GUI, users can input the RTMP stream address through a dedicated field, which by default is pre-filled with a local server address commonly used for initial testing. In addition, the detection model in .pt format can be selected through the Model Path (.pt) field, which is equipped with a Browse button for convenient file selection. Directly below these fields are the Start and Stop buttons, designed in a simple white style with black fonts to maintain clarity and consistency. Pressing the Start button prompts the system to load the YOLOv10 model, open the UAV livestream from the specified RTMP address, and display the detection results in real time through 854×480 pixel video window. Each incoming frame is processed, annotated with bounding boxes, class labels (fire and smoke), and confidence scores, then immediately displayed on the application screen.

A unique feature of this GUI is the status box, placed above the video feed and below the control buttons. By default, it shows the symbol “-”. However, when the system detects both fire and smoke simultaneously, the text changes to “POTENTIAL FOREST FIRE”, serving as an early indicator of fire risk. If only fire or smoke is detected individually, the status remains “-”, minimizing false alarms and ensuring that warnings are only given when both indicators are present together. When the Stop button is pressed, the streaming process is halted and the video feed resets to black, indicating that the detection process has ended.

3.2.Model Performance Evaluation

In this study, training the YOLOv10 model resulted in 24 different configurations derived from a combination of six optimizers (Adam, AdamW, NAdam, RAdam, RMSprop, and SGD), two learning rate values (0.001 and 0.0001), and two batch sizes (16 and 24). All models were trained using an input image size of 640×640 pixels. The training was conducted on an NVIDIA RTX 3060 Laptop GPU for 100 epochs with early stopping applied to prevent overfitting.

Table 1. YOLOv10 Model Hyperparameter Combination Details and Evaluation Results

No.	Optimizer	Batch Size	Learning Rate	Accuracy	Precision	Recall	F1 Score	Mean IoU	mAP
1	Adam	16	0.001	0.8485	0.8365	0.806	0.8205	0.7259	0.862
2	Adam	16	0.0001	0.8515	0.8405	0.812	0.8255	0.7004	0.835
3	Adam	24	0.001	0.867	0.8555	0.836	0.845	0.7256	0.876
4	Adam	24	0.0001	0.8415	0.8255	0.797	0.81	0.6957	0.841
5	AdamW	16	0.001	0.8685	0.8735	0.824	0.8475	0.7245	0.868
6	AdamW	16	0.0001	0.834	0.838	0.7665	0.7975	0.7019	0.838
7	AdamW	24	0.001	0.849	0.8195	0.8275	0.8225	0.7114	0.857
8	AdamW	24	0.0001	0.8445	0.819	0.812	0.8145	0.7095	0.832
9	NAdam	16	0.001	0.862	0.869	0.812	0.8385	0.7376	0.867
10	NAdam	16	0.0001	0.8535	0.8435	0.812	0.8265	0.6987	0.836
11	**NAdam**	**24**	**0.001**	**0.879**	**0.8705**	**0.8575**	**0.863**	**0.7373**	**0.87**
12	NAdam	24	0.0001	0.8445	0.817	0.815	0.8145	0.708	0.844
13	RAdam	16	0.001	0.863	0.8465	0.836	0.8405	0.7155	0.867
14	RAdam	16	0.0001	0.843	0.8395	0.788	0.812	0.7005	0.83
15	RAdam	24	0.001	0.863	**0.8735**	0.8512	0.841	0.71	**0.87**
16	RAdam	24	0.0001	0.836	0.8475	0.764	0.803	0.6709	0.83
17	RMSprop	16	0.001	0.712	0.7345	0.5065	0.587	0.4324	0.574
18	RMSprop	16	0.0001	0.76	0.786	0.6065	0.684	0.5683	0.73
19	RMSprop	24	0.001	0.689	0.7815	0.422	0.5455	0.3816	0.573
20	RMSprop	24	0.0001	0.7525	0.7335	0.6185	0.668	0.563	0.714

21	SGD	16	0.001	0.851	0.843	0.806	0.8235	0.7189	0.851
22	SGD	16	0.0001	0.7495	0.7215	0.615	0.6585	0.5611	0.692
23	SGD	24	0.001	0.839	0.821	0.797	0.8085	0.695	0.847
24	SGD	24	0.0001	0.7405	0.6815	0.6185	0.642	0.5762	0.673

Based on the experiment, it is evident that the best configuration is achieved by Model-11, which utilizes a combination of the NAdam optimizer, the batch size of 24, and the learning rate of 0.001, as shown in table 1. The best performance of accuracy, recall, F1 score, and mean IoU are 0.879, 0.8575, 0.863, and 0.7373, respectively. Although the values of precision (0.8705) and mAP (0.870) were not the highest individually, they remained relatively high and contributed positively to the overall model performance. This model shows a strong balance between precision and sensitivity, as well as good spatial accuracy in detecting fire and smoke objects.

The results in table 1 also reveal that optimizer selection significantly influences model stability and final performance. Among the tested optimizers, RMSprop consistently produced the lowest accuracy, recall, and mAP values. This can be attributed to its tendency to accumulate squared gradients in non-stationary training conditions, causing unstable convergence when applied to complex aerial imagery with high background variability. In contrast, adaptive optimizers such as NAdam and RAdam demonstrated smoother loss reduction and faster convergence, leading to more consistent detection of smoke regions that exhibit subtle pixel intensity differences. The proposed YOLOv10 model with NAdam optimization achieved a recall of 0.8575, indicating its robustness in minimizing undetected fire events while maintaining competitive precision (0.8705). This balance ensures the model remains both responsive and reliable for real-time early warning applications.

Table 2. Comparison of YOLOv10 Performance with Previous Forest Fire Detection Studies

Model	mAP	Precision	Recall
Improved YOLOv5 [30]	0.788	0.734	0.761
YOLOv3 [31]	0.789	0.840	0.780
LD-YOLO (YOLOv8) [4]	0.863	0.801	0.798
Ours (the best model proposed)	0.870	0.8705	0.8575

The proposed YOLOv10 model outperforms previous YOLO-based approaches in forest fire detection tasks, as shown in table 2. The results clearly indicate that YOLOv10 achieves superior overall performance across all evaluation metrics. Compared with earlier models such as YOLOv3 and the improved YOLOv5, the proposed approach yields notable gains in mAP, precision, and recall. Specifically, YOLOv10 recorded mAP of 0.87, which surpasses YOLOv3 (0.789) and improved YOLOv5 (0.788) by approximately 8–10%. The recall value of 0.8575 also demonstrates the model's robustness in minimizing false negatives, a critical aspect in early wildfire detection where missed smoke or flame events can lead to severe consequences.

Furthermore, compared with the LD-YOLO (YOLOv8-based) model, YOLOv10 still achieved slightly higher accuracy while maintaining a better balance between precision and recall. This improvement is largely attributed to YOLOv10's architectural enhancements, including the use of larger convolutional kernels and the Partial Self-Attention (PSA) mechanism, which collectively enhance feature extraction for small and low-contrast smoke regions. These results confirm that the proposed YOLOv10 framework provides a more reliable and responsive solution for UAV-based real-time forest fire detection applications. These findings further emphasize YOLOv10's suitability for early warning systems, especially in resource-limited UAV deployments where both accuracy and inference speed are critical.

3.3. Drone Configuration

In this study, the DJI Mini 4 Pro drone was used as an aerial image acquisition device. The input source came from a live feed sent via the RTMP-livestream feature on the DJI RC2 remote controller. Once the server was configured and running, the stream was sent from the remote controller to the address rtmp://<IP-Computer>/live/stream via the custom RTMP-livestream feature. The video was then forwarded to a local laptop via a WiFi network using the NGINX server with an RTMP module as a receiver and real-time data provider; the NGINX server in our setup ran on a

Windows device using a precompiled build specifically built for Windows platforms. RTMP configuration was carried out on the standard port 1935 with live = on and record = off settings to reduce latency.

The received stream was captured using the OpenCV library in a Python script, where the video stream was accessed using the RTMP Pull protocol. Incoming frames were decoded and resized to 640×640 pixels before being processed frame-by-frame by the YOLOv10 model to perform live fire and smoke detection. During testing, the drone was flown at a height of 5–10 meters from the ground and the video resolution was automatically configured at 720p with 30 FPS, which is the maximum limit of the DJI Mini 4 Pro's built-in RTMP system. Figure 4 presents a system block diagram that illustrates the flow of video delivery from the drone camera to processing on the local device.

While this configuration yields low-latency, end-to-end real-time inference suitable for proof-of-concept experiments, we acknowledge its practical limitations in terms of flight duration, altitude coverage, and reliance on local WiFi; scalability to higher altitudes (e.g., 50–100 m) and alternative communication channels (LTE/5G/satellite or multi-UAV mesh) is discussed in the Limitations and Future Work section.

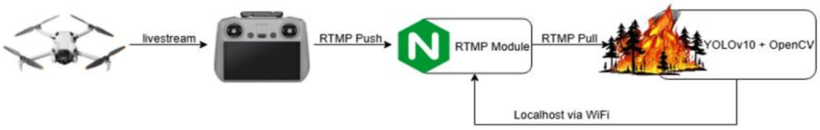
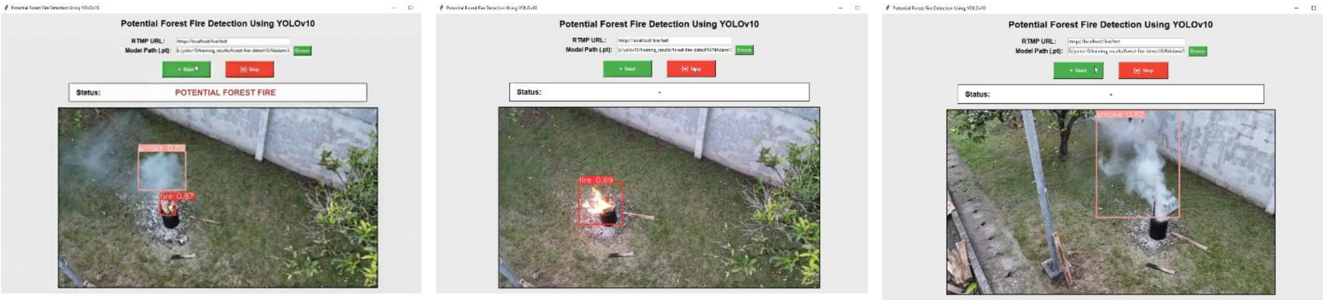


Figure 4. Block diagram of real-time fire detection system

3.4. Real-time Potential Forest Fire Detection Model Testing and External Dataset

The next test evaluates the performance of the YOLOv10 model in real-time using a livestream video from a DJI Mini 4 Pro drone that is processed frame-by-frame. The test was conducted to detect fire and smoke at a drone height of 5–10 meters with a livestream resolution of 720p. The results of the real-time test are shown in figure 5(a) where the system successfully identified fire and smoke objects as well as detecting potential fire status.



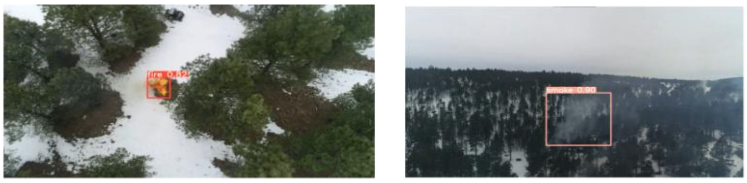
(a) (b) (c)

Figure 5. Real-time detection (a) POTENTIAL FOREST FIRE (b) fire only (c) smoke only

The result shown in figure 5(a) shows the system can accurately detect fire and smoke with bounding boxes, class labels, and confidence scores consistently displayed. Detection is stable, responsive, and fast without visual disturbances. The test was simulated using a fire source producing smoke in an open area to represent forest fire conditions. The status box displays “POTENTIAL FOREST FIRE” only when both fire and smoke are detected simultaneously providing an early indication of wildfire potential while minimizing false alarms. For example, figure 5(b) and figure 5(c) shows the system will not classify the status as “POTENTIAL FOREST FIRE” if the system detects fire only or smoke only.

Figure 5(b) and figure 5(c)further emphasize that the system does not trigger the warning status unless both fire and smoke are detected simultaneously. When only fire or only smoke is detected, the status box remains “-” indicating no potential forest fire classification. This design minimizes false alarms and ensures that the warning is issued only when both critical indicators of a forest fire are detected together. Therefore, the system provides not only accurate object detection but also a more reliable decision-support mechanism for early forest fire mitigation. These findings confirm the success of the YOLOv10 implementation with hyperparameter optimization in a real-time UAV-based potential forest fire detection system.

To test generalization, the model was tested on external videos from [32] which recorded forest fires using a DJI Matrice 200 drone and a Zenmuse X4S camera with a resolution of 1280×720 pixels at 29 FPS. The results showed successful fire and smoke detection with a confidence score greater than 0.8 on a snowy forest background different from the training data, confirming the generalization and adaptability of the YOLOv10 model. Visualization of the YOLOv10 model testing results against external videos is shown in figure 6.



(a) (b)

Figure 6. Fire detection results by YOLOv10 on external video (a) Fire detection (b) Smoke detection

4. Limitations and Future Work

Although the proposed YOLOv10-based framework demonstrated strong performance in real-time forest fire detection using UAV imagery, several limitations remain. First, the experiments were conducted at a relatively low flight altitude (5–10 m), which restricts the system's scalability for wide-area forest surveillance. Second, the current implementation relies on a WiFi-based RTMP connection between the UAV and the ground station, which may be impractical in remote forest regions with limited network coverage. Additionally, the DJI Mini 4 Pro drone used in this study has limited flight duration and weather resistance, affecting long-term monitoring capability under harsh outdoor conditions.

Future work will focus on extending the system for higher-altitude detection (50–100 m) using industrial-grade UAVs and integrating long-range communication technologies such as LTE, 5G, or satellite networks to ensure stable data transmission. Incorporating multimodal data (e.g., infrared and thermal imagery) and optimizing model quantization for on-board deployment are also promising directions. Finally, real-world testing under various environmental conditions and collaborative multi-UAV operation will be explored to validate system robustness and scalability for operational early-warning systems.

It is important to note that the reported metrics in this study were obtained from a single complete training run using the best model checkpoint. Due to time and computational constraints during the revision stage, multiple independent runs with different random seeds were not performed to estimate the variance of these results. Future work will include repeated experiments to assess the reproducibility and stability of the YOLOv10 model's performance, following best practices in deep learning evaluation.

5. Ethical and Regulatory Considerations

The use of UAVs for wildfire monitoring involves several ethical and legal considerations. All UAV operations in this study complied with local civil aviation and research regulations, including restrictions on flight altitude, airspace boundaries, and non-commercial usage. The experiments were conducted in controlled and uninhabited areas to prevent interference with public safety and privacy. No Personally Identifiable Information (PII) or human subjects were captured during data collection.

From an ethical standpoint, the deployment of UAV-based surveillance systems must balance the need for environmental safety with concerns related to data protection and public privacy. Future implementations should include onboard data encryption and anonymization protocols to prevent misuse of collected imagery. Additionally, collaboration with local forestry and disaster management agencies will be essential to ensure responsible, transparent, and lawful deployment of UAV-assisted early warning systems.

6. Discussion

The evaluation results demonstrate that the hyperparameter configuration had a significant impact on both classification and spatial detection performance. The best-performing model was achieved using the NAdam optimizer with a batch size of 24 and a learning rate of 0.001. This configuration produced the highest values in accuracy, precision, recall, F1 score, mean IoU, and mAP, which are 0.879, 0.8705, 0.8575, 0.863, 0.7373, and 0.870, respectively. These results indicate that the model achieves a strong balance between classification performance and spatial accuracy.

It is important to note that these metrics were obtained from the best-performing checkpoint of a single training run. Multiple independent training repetitions with different random seeds were not performed due to computational and time constraints. Nevertheless, the model exhibited stable convergence behavior during training, as shown by consistent validation trends across epochs.

Real-time testing was carried out using video livestreams from the DJI Mini 4 Pro drone via the RTMP protocol. The system ran stably and responsively under various visual conditions, including different smoke densities and fire intensities. It also demonstrated strong generalization capabilities when tested on external datasets beyond the original training data. These findings confirm the system's ability to operate effectively under dynamic field conditions and validate the feasibility of UAV-based real-time forest fire detection using deep learning.

When compared with previous YOLO-based wildfire detection studies, the proposed YOLOv10 model demonstrates competitive performance across all key evaluation metrics. Its mAP of 0.87 is slightly lower than the transformer-enhanced YOLOv9 model (0.962) but notably higher than earlier models such as YOLOv3 (0.789) and improved YOLOv5 (≈0.82). Unlike heavier architectures that require high-end GPUs, the proposed YOLOv10 achieves a balance between accuracy and real-time efficiency, operating smoothly on standard hardware while maintaining reliable detection of both flame and smoke. These findings indicate that YOLOv10 provides a practical and efficient trade-off between performance and computational cost, making it suitable for UAV-based early fire detection applications.

7. Conclusion

The implementation of the YOLOv10 with appropriate hyperparameter optimization by using UAV imagery captured by the DJI Mini 4 Pro proved to be effective for real-time potential forest fire detection. The optimal model performance was obtained using the NAdam optimizer with a batch size of 24 and a learning rate of 0.001. This configuration achieved the highest values in accuracy, precision, recall, F1 score, mean IoU, and mAP, which are 0.879, 0.8705, 0.8575, 0.863, 0.7373, and 0.870, respectively. The model demonstrated high performance in both classification and spatial detection tasks and successfully operated in real-time during field testing with livestream video from the DJI Mini 4 Pro.

The reported metrics were derived from the best checkpoint of a single training run, which showed stable convergence across epochs. These findings highlight the capability of YOLOv10 to provide reliable detection of both fire and smoke features in dynamic environments. Furthermore, this research demonstrates the potential of integrating UAV and deep learning technologies to support early fire warning systems. Future developments will focus on deployment through cloud-based or IoT-integrated monitoring systems, enabling scalable and automated forest fire detection in real-world operations. Overall, YOLOv10 demonstrates a promising balance between accuracy, speed, and computational efficiency for real-time forest fire detection using UAV imagery.

8. Declarations

8.1. Author Contributions

Conceptualization: J.P., M.R.P, and R.M.M.; Methodology: J.P and M.R.P.; Software: M.R.P and R.M.M.; Validation: J.P., M.R.P., and M.P.; Formal Analysis: J.P. and M.R.P.; Investigation: J.P. and M.R.P.; Resources: J.P and M.R.P. ; Data Curation: M.R.P., and R.M.M.; Writing Original Draft Preparation: J.P. and M.R.P.; Writing Review and Editing: J.P. and M.R.P.; Visualization: R.M.M. and J.P.; All authors have read and agreed to the published version of the manuscript.

8.2. Data Availability Statement

The dataset used in this study was obtained from publicly available sources on Roboflow Universe and Kaggle.

8.3. Funding

The authors received financial support from LPPM Institut Teknologi Nasional Bandung, Indonesia.

8.4. Institutional Review Board Statement

Not applicable.

8.5. Informed Consent Statement

Not applicable.

8.6. Declaration of Competing Interest

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

References

[1] Z. Jiao , "A Deep Learning Based Forest Fire Detection Approach Using UAV and YOLOv3," 2*019 1st International Conference on Industrial Artificial Intelligence (IAI), Shenyang, China*, 2019, pp. 1-5, doi: 10.1109/ICIAI.2019.8850815.

[2] Mr. S. Goyal, Mr. M. Shagill, Ms. A. Dhillon, Dr. H. Vohra, and Dr. A. Singh, "A YOLO based Technique for Early Forest Fire Detection," *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 6, pp. 1357–1362, Apr. 2020, doi: 10.35940/ijitee.F4106.049620.

[3] Rose JJ, Wang L, Xu Q, McTiernan CF, Shiva S, Tejero J, Gladwin MT. "Carbon Monoxide Poisoning: Pathogenesis, Management, and Future Directions of Therapy," *Am J Respir Crit Care Med*. 2017 Mar 1;195(5):596-606. doi: 10.1164/rccm.201606-1275CI.

[4] Z. Lin, B. Yun, and Y. Zheng, "LD-YOLO: A Lightweight Dynamic Forest Fire and Smoke Detection Model with Dysample and Spatial Context Awareness Module," *Forests*, vol. 15, no. 9,p.1-12, Sep. 2024, doi: 10.3390/f15091630.

[5] V. M. Becerra, "Autonomous control of unmanned aerial vehicles," *Electronics*, vol. 8, no. 4, p.1-12, Apr. 2019, doi: 10.3390/electronics8040452.

[6] Y. Chen , "UAV Image-based Forest Fire Detection Approach Using Convolutional Neural Network," 2019 14th *IEEE Conference on Industrial Electronics and Applications (ICIEA),* Xi'an, China, 2019, pp. 2118-2123, doi: 10.1109/ICIEA.2019.8833958.

[7] M. Yandouzi , "Investigation of Combining Deep Learning Object Recognition with Drones for Forest Fire Detection and Monitoring," *International Journal of Advanced Computer Science and Applications (IJACSA),* vol. 14, no. 3, p1-12, 2023, doi: 10.14569/IJACSA.2023.0140342.

[8] P. Li and W. Zhao, "Image fire detection algorithms based on convolutional neural networks," *Case Studies in Thermal Engineering*, vol. 19, no. 1, pp-12, Jun. 2020, doi: 10.1016/j.csite.2020.100625.

[9] X. Zheng, F. Chen, L. Lou, P. Cheng, and Y. Huang, "Real-Time Detection of Full-Scale Forest Fire Smoke Based on Deep Convolution Neural Network," *Remote Sensing*, vol. 14, no. 3, pp1-12, Feb. 2022, doi: 10.3390/rs14030536.

[10] M. Mao, A. Lee, and M. Hong, "Efficient Fabric Classification and Object Detection Using YOLOv10," *Electronics*, vol. 13, no. 19, p.1-12, Oct. 2024, doi: 10.3390/electronics13193840.

[11] A. S. Geetha, M. A. R. Alif, M. Hussain, and P. Allen, "Comparative Analysis of YOLOv8 and YOLOv10 in Vehicle Detection: Performance Metrics and Model Efficacy," *Vehicles,* vol. 6, no. 3, pp. 1364–1382, Aug. 2024, doi: 10.3390/vehicles6030065.

[12] F. Agarwal, I. Chakraborty, S. Banerjee, and A. P. J. Abdul, "Impact of Forest Fire on Climate Change," *International Journal of Advances in Engineering and Management (IJAEM),* vol. 6, no.1, p. 391-398, 2024, doi: 10.35629/5252-0601391398.

[13] S. Urbanski, "Wildland fire emissions, carbon, and climate: Emission factors," *Forest Ecology and Management*, vol. 317, no.1, pp. 51–60, Apr. 2014, doi: 10.1016/j.foreco.2013.05.045.

[14] S. Sannigrahi , "Examining the effects of forest fire on terrestrial carbon emission and ecosystem production in India using remote sensing approaches," *Science of the Total Environment*, vol. 725, no.1, pp.1-12, Jul. 2020, doi:

10.1016/j.scitotenv.2020.138331.

[15] F. Tedim , “Defining extreme wildfire events: Difficulties, challenges, and impacts,” *Fire,* vol. 1, no. 1, pp. 1–28, Jun. 2018, doi: 10.3390/fire1010009.

[16] G. Huidobro, L. Giessen, and S. L. Burns, “And it burns, burns, burns, the ring-of-fire: Reviewing and harmonizing terminology on wildfire management and policy,” *Environmental Science & Policy*, vol. 157, no.1, pp.1-12, Jul. 01, 2024, doi: 10.1016/j.envsci.2024.103776.

[17] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif, “Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review,” *Drones*, vol. 6, no. 6, pp.1-12, Jun. 01, 2022, doi: 10.3390/drones6060147.

[18] Q. Song, Y. Zeng, J. Xu, and S. Jin, “A survey of prototype and experiment for UAV communications,” *Sci. China Inf. Sci*, vol. 64, no.1, pp.1-12, 2021, doi: 10.1007/s11432-020-3030-2.

[19] S. H. Alsamhi , “UAV Computing-Assisted Search and Rescue Mission Framework for Disaster and Harsh Environment Mitigation,” *Drones*, vol. 6, no. 7, pp.1-12, 2022, doi: 10.3390/drones6070154.

[20] J. Cai, H. Zhang, A. Khadakar, A. Mohammadzadeh, and C. Zhang, “Automatic control of UAVs: new adaptive rules and type-3 fuzzy stabilizer,” *Complex and Intelligent Systems,* vol. 10, no.1, pp. 7235–7248, Oct. 2024, doi: 10.1007/s40747-024-01434-y.

[21] P. Liu , “A review of rotorcraft unmanned aerial vehicle (UAV) developments and applications in civil engineering,” *Smart Structures and Systems, Techno Press*, vol. 13, no. 6, p.1-12, 2014, doi: 10.12989/sss.2014.13.6.1065.

[22] E. Dritsas and M. Trigka, “Remote Sensing and Geospatial Analysis in the Big Data Era: A Survey,” *Remote Sensing*, vol.17, no. 3, p.1-12, 2025, doi: 10.3390/rs17030550.

[23] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, and I. Moscholios, “A compilation of UAV applications for precision agriculture,” *Computer Networks*, vol. 172, no.1, pp.1-12, May 2020, doi: 10.1016/j.comnet.2020.107148.

[24] A. Wang , “YOLOv10: Real-Time End-to-End Object Detection,”, arXiv, May 2024, doi: 10.48550/arXiv.2405.14458.

[25] S. Sathyanarayanan and B. R. Tantri, “Confusion Matrix-Based Performance Evaluation Metrics,” *African Journal of Biomedical Research,* vol. 27, no. 45, pp. 4023–4031, Nov. 2024, doi: 10.53555/AJBR.v27i4S.4345.

[26] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression,” *arXiv*, 2019, doi: 10.48550/arXiv.1902.09630.

[27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” *arXiv*, 2015, doi: 10.48550/arXiv.1506.02640.

[28] R. Padilla, S. L. Netto, and E. A. B. Da Silva, “A Survey on Performance Metrics for Object-Detection Algorithms,” 2020 *International Conference on Systems, Signals and Image Processing (IWSSIP), Niteroi, Brazil*, 2020, pp. 237-242, doi: 10.1109/IWSSIP48289.2020.9145130.

[29] I. El-Madafri, M. Peña, and N. Olmedo-Torre, “The Wildfire Dataset: Enhancing Deep Learning-Based Forest Fire Detection with a Diverse Evolving Open-Source Dataset Focused on Data Representativeness and a Novel Multi-Task Learning Approach,” *Forests,* vol. 14, no. 9, p.1-12, 2023, doi: 10.3390/f14091697.

[30] L. Cao, Z. Shen, and S. Xu, “Efficient forest fire detection based on an improved YOLO model,” *Visual Intelligence*, vol. 2, no. 20, p.1-12, 2024, doi: 10.1007/s44267-024-00053-y.

[31] Z. Jiao , “A YOLOv3-based Learning Strategy for Real-time UAV-based Forest Fire Detection,” 2020 *Chinese Control And Decision Conference (CCDC), Hefei, China*, 2020, pp. 4963-4967, doi: 10.1109/CCDC49329.2020.9163816.

[32] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Z. Fulé, and E. Blasch, “Aerial imagery pile burn detection using deep learning: The FLAME dataset,” *Computer Networks*, vol. 193,no.1, pp.1-12, Jul. 2021, doi: 10.1016/j.comnet.2021.108001.