مجلة جامعة فزان العلمية
Fezzan University scientific Journal
journal@fezzanu.edu.ly

Comparison of YOLOv5 and YOLOv8 Models' Performance in Early Fire and Smoke Detection Applications

*Emsaieb Geepalla and Karima Ahmed [1]
[1]School of Biomedical Engineering, Faculty of Engineering, Wadi Alshatti University, Alshatti, Libya

A B S T R A C T

Abstract: In this paper, we conducted a comparative study on the performance of the YOLOv5 and YOLOv8 models for fire and smoke detection across various scenarios. Utilizing a dataset of 9,756 images capturing diverse fire incidents under different environmental conditions, both models were trained using identical hyperparameters (learning rate = 0.001, batch size = 16, and 40 epochs). YOLOv8 consistently outperformed YOLOv5 in terms of accuracy, precision, and recall across various evaluation metrics, highlighting the architectural improvements introduced in YOLOv8. Notably, YOLOv8 performed better in detecting fires in complex scenarios, such as small fires and challenging lighting conditions, where YOLOv5 faced difficulties. However, both models faced challenges in detecting transparent smoke, particularly in daylight. The results suggest that while YOLOv8 holds promise for further improvements in fire detection, expanding the dataset and exploring more advanced configurations could lead to better performance in real-world applications. This research emphasizes the importance of model selection based on specific applications and the potential of the latest YOLO versions to enhance early fire detection.

Keywords: Fire and smoke detection, YOLO8, Computer Vision, Deep Learning

مقارنة بين أداء نموذجي YOLOv5 وYOLOv8 في تطبيقات الكشف المبكر عن الحرائق والدخان

*امسيب جيب الله، كريمة احمد [1]

قسم الهندسة الطبية الحيوية، كلية الهندسة، جامعة وادي الشاطئ، براك الشاطئ، ليبيا

الملخص

في هذه الورقة، أجرينا دراسة مقارنة لأداء نموذجي YOLOv5 وYOLOv8 للكشف عن الحرائق والدخان في سيناريوهات متنوعة. باستخدام مجموعة بيانات مكونة من 9756 صورة تلتقط حوادث حرائق مختلفة في ظل ظروف بيئية مختلفة، قمنا بتدريب كلا النموذجين تحت نفس المعلمات الفائقة (معدل التعلم = 0.001، وحجم الدفعة = 16، و40 حقبة). تفوق YOLOv8 باستمرار على YOLOv5 من حيث الدقة والضبط والتذكر عبر مقاييس التقييم المختلفة، مما يؤكد التحسينات المعمارية المقدمة في YOLOv8. والجدير بالذكر أن YOLOv8 أظهر أداءً متفوقًا في اكتشاف الحرائق في

:Corresponding author
E-mail: Ems.geepalla@wau.edu.ly Received in 10 April 2025 - revised form 19 May 2025
Accepted 22 May 2025 - Publication date 20 June 2025

السيناريوهات المعقدة، مثل الحرائق الصغيرة وظروف الإضاءة الصعبة، حيث واجه YOLOv5 صعوبة. ومع ذلك، واجه كلا النموذجين صعوبات في اكتشاف الدخان الشفاف، وخاصة أثناء النهار. تشير النتائج إلى أنه في حين أن YOLOv8 يحمل وعدًا بمزيد من التحسينات في اكتشاف الحرائق، فإن توسيع مجموعة البيانات واستكشاف تكوينات أكثر تقدمًا قد يؤدي إلى أداء أفضل في التطبيقات الواقعية. يسلط هذا البحث الضوء على أهمية اختيار النموذج بناءً على التطبيق المحدد وإمكانات أحدث إصدارات YOLO في تحسين اكتشاف الحرائق المبكر.

الكلمات المفتاحية: اكتشاف الحريق والدخان، YOLO8، الرؤية الحاسوبية، التعلم العميق

Introduction

Fires are among the most devastating natural disasters globally, posing threats to human lives and causing significant damage to property and infrastructure, along with severe environmental consequences [1, 2]. They contribute to the emission of large amounts of greenhouse gases, exacerbating climate change and negatively affecting air quality. Hence, early fire and smoke detection is crucial for mitigating these harmful effects and protecting both the environment and human resources [3].

In recent decades, artificial intelligence (AI) technologies, particularly neural networks, have seen remarkable advancements, enabling effective solutions to the challenges faced by traditional detection systems. Computer vision models have emerged as one of the most prominent solutions, offering high levels of accuracy and speed in object detection, making them suitable for applications such as early fire and smoke detection [3-5]. The "YOLO" (You Only Look Once) algorithm is one of the most efficient models in this field due to its real-time data processing capability and fast object detection [4].

Several previous studies have compared models from the YOLO series with other models. For instance, reference [6] presents a study on fire and smoke detection in forests that compared YOLOv5 with algorithms like VGG16 and VGG19. YOLOv5 outperformed the other models in object detection accuracy on the test set and demonstrated superior performance in recognizing fires in various scenarios. In contrast, reference [7] conducted a comparative study between different YOLO versions, but our study focuses specifically on a detailed comparison between YOLOv5 and YOLOv8. Both models are highly efficient and widely adopted, with YOLOv8 being the latest release from Ultralytics, which builds on the architectural success of YOLOv5 while incorporating additional improvements. Furthermore, reference [4] compared YOLOv5 and YOLOv8 in forest fire scenarios, whereas our study aims to evaluate the models using a diverse dataset that reflects multiple scenarios, thereby enhancing their performance in recognizing fires and smoke in complex environments.

YOLO models, which rely on convolutional neural networks (CNNs), represent a fundamental advancement in the field of computer vision, with several versions released over time. YOLOv5 is one of the most commonly used versions due to its accuracy and speed in image processing. In contrast, YOLOv8 was recently launched, offering significant performance improvements in both accuracy and speed, making it a subject of interest for researchers in fields that require real-time data processing, such as fire and smoke detection [4, 6].

The importance of this research paper lies in conducting a comprehensive comparison of the performance of the YOLOv5 and YOLOv8 models in early fire and smoke detection applications. A diverse dataset was used, including different scenarios, such as forest and building scenes, along with various lighting conditions. Various hyperparameters, such as batch size, number of training epochs, optimizers, and learning rate, were applied to assess the models' overall performance. The comparison includes different variants (m, l, x) of both YOLOv5 and YOLOv8, and performance will be analyzed using multiple metrics, including recall, precision, F1 score, and mean Average Precision (mAP).

The results of this study are expected to provide significant insights into developing more effective solutions for early fire and smoke detection and to promote the use of AI technologies in addressing growing environmental challenges. Additionally, the research will offer a detailed comparison of the latest available models, contributing to future efforts aimed at improving and developing intelligent detection systems based on computer vision.

Materials and Methods

Dataset Preparation

The dataset is a pivotal step in developing machine learning (ML) and deep learning (DL) models, as it forms the foundation for evaluating the models' efficiency and their ability to handle real-world scenarios [6, 8]. In this paper, data were collected from various sources such as Kaggle, Pixels, and public platforms like social media and news websites. The dataset includes images and videos documenting fire and smoke incidents in diverse scenarios, such as forests, vehicles, and buildings, both indoors and outdoors, under varying lighting conditions (day and night). The dataset was cleaned to remove duplicates and irrelevant data, resulting in a final set of 3,565 images, classified into 1,000 images containing fire only, 1,000 images with smoke only, and 1,565 images containing both fire and smoke.

The target objects (fire and smoke) were manually labeled. To enhance the dataset size, data augmentation techniques such as 90-degree rotation and flipping were applied, increasing the dataset to 9,756 images. The images were resized to 640×640 pixels, following the size recommended by model developers. Subsequently, the dataset was split into three subsets: 80% for training, 10% for validation, and 10% for testing. This division aims to optimize model performance. Data preprocessing and augmentation were conducted using the Roboflow platform to ensure a smooth and efficient workflow. Figure 1 illustrates a sample of the dataset.

Fig. 1. A sample of the dataset used.

Model Architecture under Study

- YOLOv5

YOLOv5 (short for You Only Look Once version 5) is an advanced object detection algorithm known for its simplicity, accuracy, and reliability. This model was released by the Ultralytics team on June 25, 2020 [4,9]. The model consists of four main components: input, backbone, neck, and head, as illustrated in Figure 2.

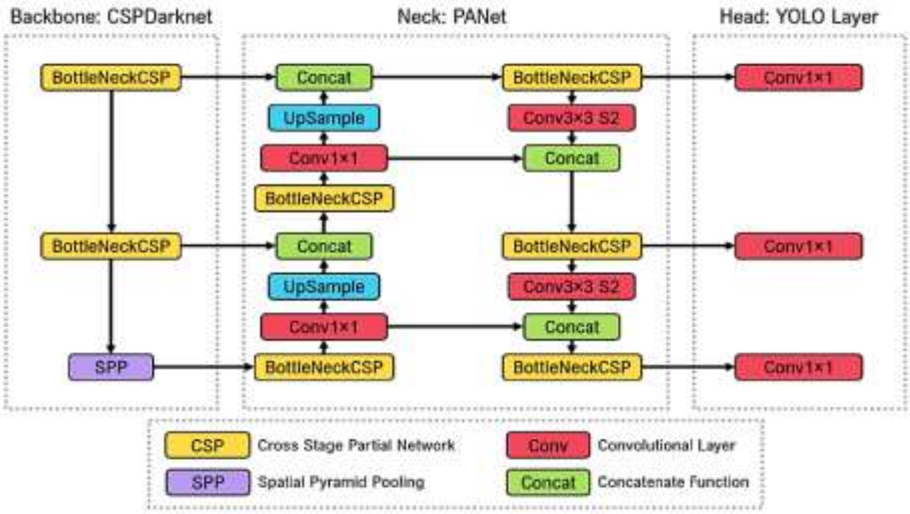

Fig. 2. The Architecture of the YOLOv5 Model [4].

The backbone of the YOLOv5 model is based on the CSP-Darknet53 convolutional network, which employs the Cross Stage Partial (CSP) strategy to facilitate the flow of information and alleviate the issues of repeated gradients and vanishing gradients [10]. Through this backbone, key features are extracted and analyzed from the input images.

The neck of YOLOv5 incorporates components derived from Spatial Pyramid Pooling (SPP), along with the integration of BottleNeckCSP into the Path Aggregation Network (PANet) [11]. This configuration enhances the receptive field and focuses on essential contextual features to improve detection accuracy. Additionally, PANet has been optimized using the CSPNet strategy to enhance precise pixel localization.

The head section follows the approach used in previous YOLO versions, consisting of three convolutional layers that predict bounding boxes, confidence scores, and object classes [10]. Despite ongoing challenges, such as feature map repetition and target errors in certain scenarios, YOLOv5 remains one of the leading models in object detection [4].
YOLOv5 is available in several versions, differing in depth, width, number of parameters, and floating-point operations. These variations provide multiple options that range from extremely fast inference speeds to high object detection accuracy [6, 12]. In this study, we will analyze three versions of the model, as shown in Table 1.

Table 1. Different Models of YOLOv5 [13].

Model	Size (pixels)	mAP (50-95)	Parameters (M)	FLOPs (B)
YOLOv5 m	640	45.4	21.2	49
YOLOv5l	640	49.0	46.5	109.1
YOLOv5 x	640	50.7	86.7	205.7

- YOLOv8

The YOLOv8 model was released on January 10, 2023, by Ultralytics as the latest version in the YOLO (You Only Look Once) series. It features significant improvements over previous versions, particularly YOLOv5, making it a powerful and efficient model for future computer vision applications [4, 14], as illustrated in Figure 3.

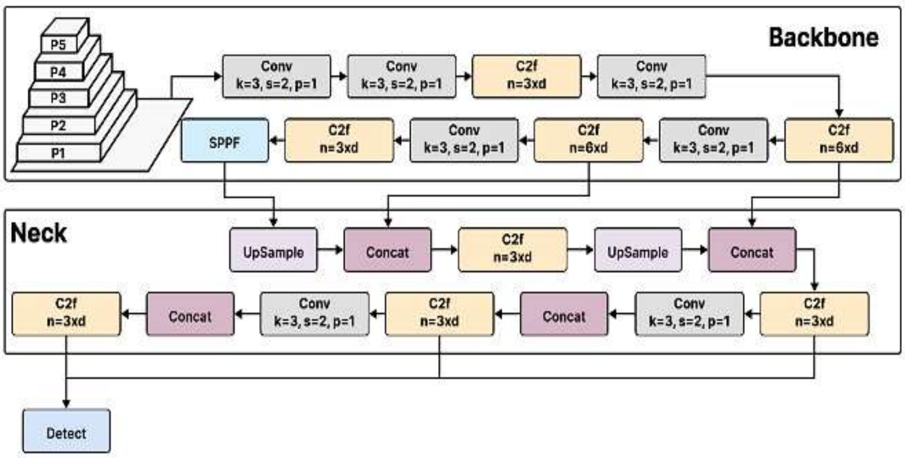

Fig. 3. Architecture of the YOLOv8 model [4].

One of the pivotal improvements in YOLOv8 is the adoption of anchor-free prediction, which eliminates the need for anchor boxes used in previous versions. These changes simplify the object detection process and alleviate challenges associated with anchor boxes, such as limited generalization and difficulties in handling irregularly shaped objects [15].
Regarding architectural design, YOLOv8 features a significant update in the convolutional network. The initial 6x6 convolution used in YOLOv5 has

been replaced with a 3x3 convolution in YOLOv8, enhancing the model's efficiency. Additionally, the core building block has been updated by replacing C3 with C2f, a technique that improves information flow and reduces the number of parameters, thus decreasing the model's size without impacting performance [16]. These architectural enhancements, along with the Fast Spatial Pyramid Pooling (SPPF) feature, augment the model's ability to detect objects across varying scales [16, 17].

YOLOv8 employs new augmentation techniques during training, such as mosaic scaling, allowing the model to learn new patterns under diverse conditions, including partial occlusions and varying backgrounds. These techniques enhance the model’s adaptability to complex scenarios [18].

In terms of performance, YOLOv8 integrates several key components, including the backbone, which relies on the CSP-Darknet53 architecture for feature extraction from the input image. Subsequently, the features pass through the SPPF layer to provide a multi-scale representation that enhances the detection of both small and large objects. These features are integrated into the neck, which utilizes both Feature Pyramid Networks (FPN) and Path Aggregation Networks (PAN) to improve information transfer between different layers and enhance feature integration [19, 20]. Finally, the predictions regarding object locations and classifications are generated using anchor-free detection, which helps reduce complexity and increase the model's speed, particularly in the Non-Maximum Suppression (NMS) process, a crucial step for filtering out inaccurate detections [15, 20].

Similar to YOLOv5, YOLOv8 is available in several versions designed to achieve a balance between detection accuracy and computational efficiency. This study focuses on analyzing three of these versions, as detailed in Table 2. These versions differ in depth, width, number of parameters, and floating-point operations, enabling various applications ranging from resource-constrained environments to those requiring high precision in object detection [21, 22].

Table 2. Different Models of YOLOv5 [13].

Model	Size (pixels)	mAP (50-95)	Parameters (M)	FLOPs (B)
YOLOv8m	640	50.2	25.9	78.9
YOLOv8l	640	52.9	43.7	165.2
YOLOv8x	640	53.9	68.2	257.8

Training and Evaluation

The training of models is a process aimed at enhancing their ability to recognize patterns and extract necessary features from the training data [23]. In this study, training began with the YOLOv8l and YOLOv5l models using manual hyperparameter tuning. An initial learning rate of 0.001 and a batch size of 16 were employed, with the Adam optimizer used to enhance model performance. Initially, these parameters were fixed while experimenting with various numbers of training epochs, starting from 20 and gradually increasing by 10 to determine the optimal count. Once the number of epochs was established at the value that demonstrated the best performance, the learning rate was gradually adjusted to find optimal values for faster training without negatively impacting the model's accuracy. Subsequently, the learning rate and the number of training epochs were fixed, while the batch size was altered to improve processing efficiency. This incremental approach continued until the hyperparameter configuration that yielded the best possible performance was achieved. After attaining optimal performance and selecting the most suitable hyperparameters for both models, the remaining models were trained with these parameters, followed by comparisons among the models using the following performance metrics:

Precision: Precision is a measure that quantifies the ratio of true positive (TP) detections to all positive predictions made by the model, which includes both false positives (FP) and true positives (TP). Precision is represented mathematically by the formula (1) [23, 24].

$$Precision = \frac{TP}{TP + FP} = \frac{TP}{all\ detections} \qquad (1)[24]$$

Recall (Sensitivity): Recall is a metric that measures the ratio of true positive (TP) detections to all ground truth instances, which includes true positives (TP) and false negatives (FN). Recall is represented mathematically by the formula (2) [23, 25].

$$Recall = \frac{TP}{TP + FN} = \frac{TP}{all\ ground\ truths} \qquad (2)[24]$$

Precision and recall are closely related metrics, as improving one often affects the other. For example, when trying to increase precision, the model's sensitivity may decrease, meaning that while the model achieves a higher rate of true positives, it may fail to detect some other positive instances. Conversely, increasing sensitivity or recall may result in decreased precision. To study this relationship more effectively, Precision-Recall Curves are used, where precision is represented on the vertical axis (Y) and recall on the horizontal axis (X). These curves are considered an effective tool for evaluating model performance; a detector (model) is deemed good if it maintains high precision while increasing recall, or if the area under the curve (AUC) increases. This is referred to as mean Average Precision (mAP). In the case of multiple classes, the average precision is calculated for each class separately, and then the mean of these precisions across all classes is found. This is one of the most important quantitative metrics for evaluating model performance [24].

F1-Score: The F1-Score is a metric that combines precision and recall into a single value, providing a balanced measure of model performance, particularly in situations where there is a trade-off between precision and recall. The F1-Score is represented by the following mathematical formula (3) [23, 24].

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \qquad (3)[23]$$

Results and Discussion

After training the models on the Google Colab platform using Tesla T4 processors (16 GB RAM, 1 GPU) under the same conditions, it was concluded that the optimal configuration for both models was achieved using a learning rate of 0.001, a batch size of 16, and the Adam optimizer, with the best performance observed at 40 epochs. The training time ranged from approximately 3 to 5 hours, depending on the model size, from the smallest (m) to the largest (x) in both versions.

Table 3 presents a comparison of the performance of the YOLOv8m and YOLOv5m models based on mean Average Precision (mAP50) across the three stages (training, validation, and testing), along with the F1-score, precision, and recall. The comparison revealed that YOLOv5m's performance was slightly lower than YOLOv8m across all performance metrics. YOLOv5m achieved a mAP50 of 82.4% (85.8% for fire, 79% for smoke) for the training and validation sets, and 82.1% (81.9% for fire, 82.4% for smoke) for the test set. The F1-score was 78% at a confidence threshold of 29.2%, with a precision of 77.7% and a recall of 72.9%.

Table 3. Performance Comparison between YOLOv5m and YOLOv8m Models

Model	bs	lr	epoch	precision	recall	F1	mAP50 train	mAP50 val	mAP50 test
YOLOv5m	16	0.001	40	%77.7	%72.9	%82.1	%82.4	%82.4	%78
YOLOv8m	16	0.001	40	%84.2	%75.8	84%	84.3%	84.3%	%80

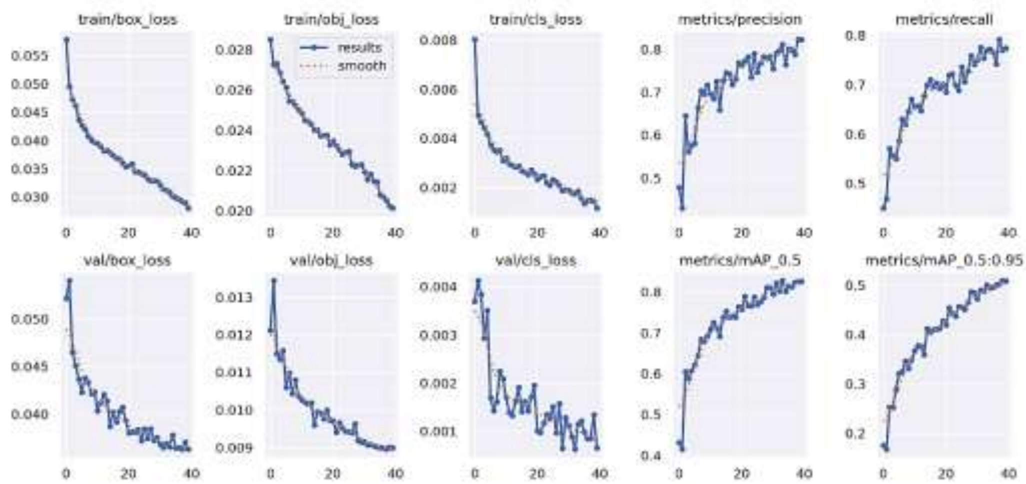

Fig. 4. Training Results for YOLOv5m Model with Parameters Epoch=40, LR=0.001, BS=16.

In contrast, YOLOv8m demonstrated a mAP50 of 84.3% (84% for fire, 84.5% for smoke) during training and validation, and 84% (82.3% for fire, 85.8% for smoke) during testing. The F1-score reached 80% at a confidence threshold of 35.5%, with a precision of 84.2% and a recall of 75.8%. Moreover, YOLOv8m showed more stable performance across various training curves, including box loss, class loss, and distributional focal loss (dfl_loss) for YOLOv8m, and object loss (obj_loss) for YOLOv5m, for both training (train) and validation (val) sets. Precision, recall, and mean Average Precision (mAP50, mAP50-95) are represented on the Y-axis, compared to the increase in the number of training epochs on the X-axis. These results are illustrated in Figure 4 for YOLOv8m, compared to Figure 5 for YOLOv5m.

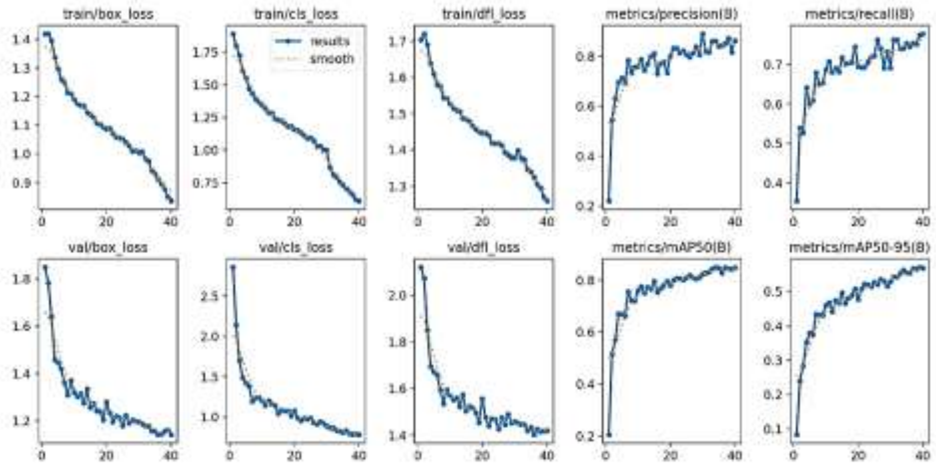

Fig. 5. Training Results for YOLOv8m Model with Parameters Epoch=40, LR=0.001, BS=16.

The YOLOv5l model achieved its best performance with the same hyperparameters used by the YOLOv8l model for optimal performance. However, the performance of YOLOv5l was slightly lower, with a mAP50 score of 82.8% (84.1% for fire, 81.6% for smoke) for both the training and validation sets, and 83.2% (83.5% for fire, 82.9% for smoke) for the test set. The F1-score reached 79% at a confidence threshold of 28.3%, with precision at 83.7% and recall at 70%.

Table 4. Performance Comparison Between YOLOv5l and YOLOv8l Models

Model	bs	lr	epoch	precision	recall	F1	mAP50 train	mAP50 val	mAP50 test
YOLOv5l	16	0.001	40	83.2%	%82.8	82.8%	83.2%	%70	%83.7
YOLOv8l	16	0.001	40	83.6%	85.2%	%85.1	80%	%75.8	%84.2

On the other hand, the YOLOv8l model achieved a mAP50 score of 85.1% (86.5% for fire, 83.6% for smoke) and 85.2% (86.7% for fire, 83.8% for smoke) for the training and validation sets, respectively, and 83.6% (82.2% for fire, 85.1% for smoke) for the test set. The F1-score for YOLOv8l was 80% at a confidence threshold of 35.5%, with precision at 84.2% and recall at 75.8%.

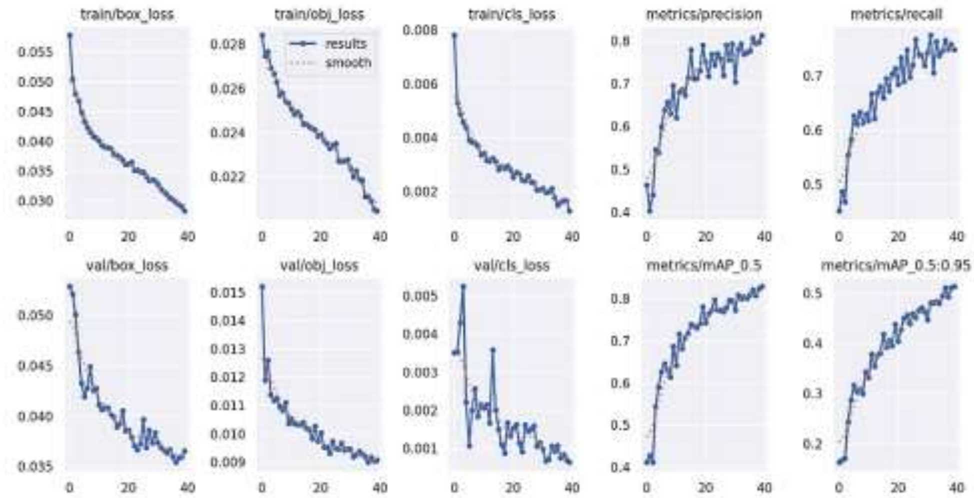

Fig. 6. Training Results for YOLOv5l Model with Parameters Epoch=40, LR=0.001, BS=16.

Furthermore, the YOLOv8l model demonstrated more stable performance across the different training curves, as shown in Figures 6 and 7 for YOLOv5l and YOLOv8l, respectively

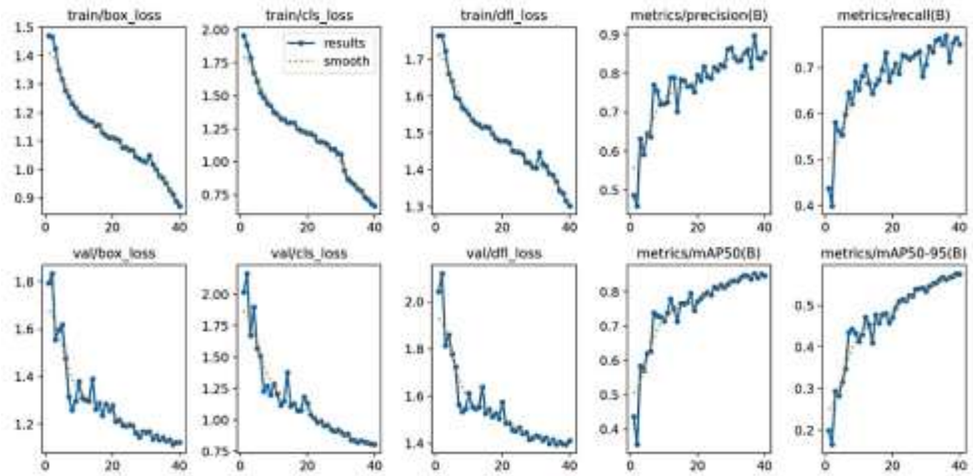

Fig. 7. Training Results for YOLOv8l Model with Parameters Epoch=40, LR=0.001, BS=16.

Table 5 presents a comparison between the performance of the YOLOv8x and YOLOv5x models based on mean Average Precision (mAP50) across three phases: training, validation, and testing, as well as the F1-score for training, precision, and recall. This training was conducted using the hyperparameters that yielded the best performance for both YOLOv8l and YOLOv5l models, specifically (LR=0.001, BS=16, Epoch=40).

Table 5. Performance Comparison Between YOLOv5x and YOLOv8x Models

Model	Bs	lr	epoch	precision	recall	F1	mAP50 train	mAP50 val	mAP50

									test
YOLO v5x	16	0.001	40	78.7%	75.8%	%77	%81.4	%81.4	%80.8
YOLO v8x	16	0.001	40	%84	77.6%	%81	85.8%	85.8%	82.8%

The comparison results indicated that the performance of YOLOv5x was lower than that of YOLOv8x across all performance metrics. The mAP50 score for the YOLOv5x model was 81.4% (83.5% for fire, 79.3% for smoke) for both the training and validation sets, and 78% (80.8% for fire, 75.8% for smoke) for the test set. The F1-score recorded an F1-score of 77% at a confidence threshold of 26.6%.

In contrast, the YOLOv8x model achieved a mAP50 score of 85.8% (86.6% for fire, 85.1% for smoke) for both the training and validation sets, and 82.8% (80.8% for fire, 84.9% for smoke) for the test set. The F1-score for YOLOv8x was 81% at a confidence threshold of 40.6%.

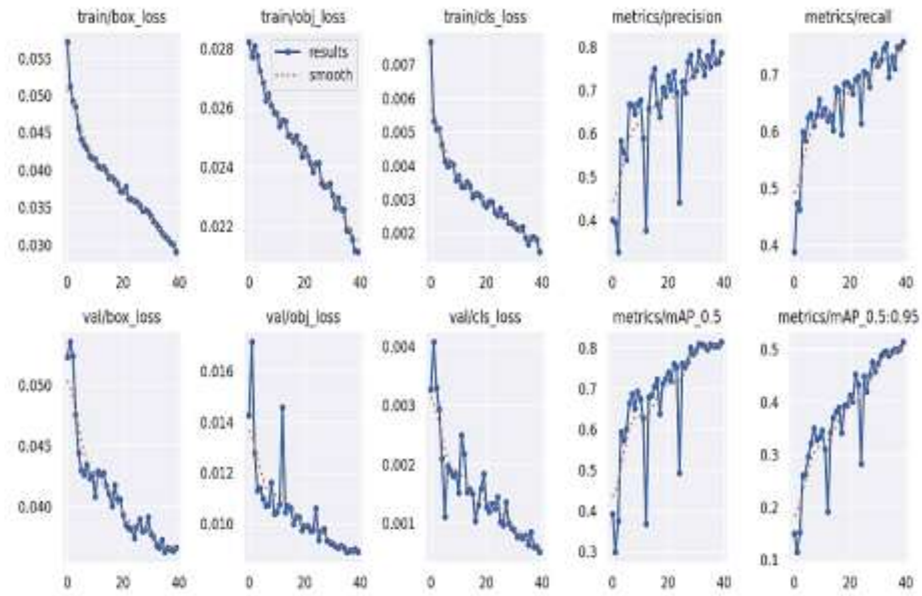

Fig. 8. Training Results for YOLOv5x Model with Parameters Epoch=40, LR=0.001, BS=16.

Moreover, the YOLOv5x model exhibited less stable performance across various training curves, as illustrated in Figure 8, compared to the more stable performance of YOLOv8x, which showed notable fluctuations in most of the mentioned performance metrics, as depicted in Figure 9. Figures 10 and 11 illustrate the predictions of the YOLOv5x and YOLOv8x models on a sample from the test set.

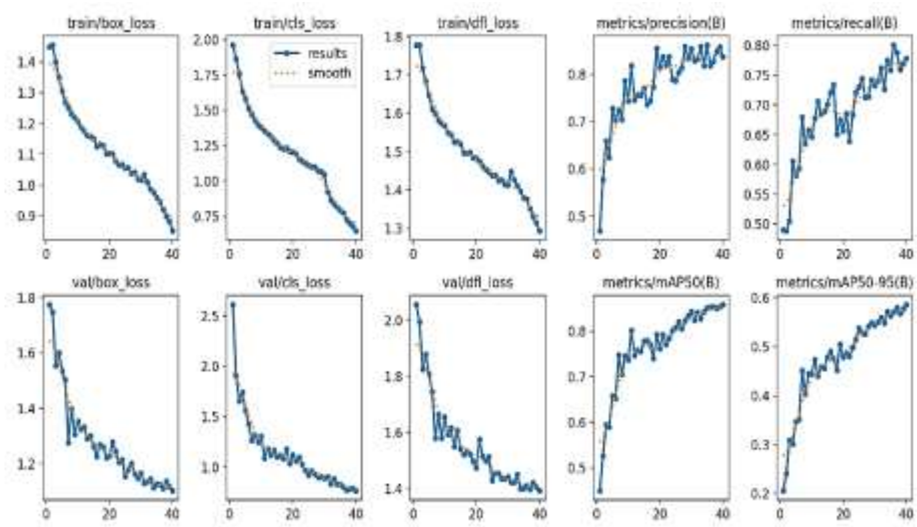

Fig. 9. Training Results for YOLOv8x Model with Parameters Epoch=40, LR=0.001, BS=16.

The trained models demonstrated commendable performance in detecting fires across various scenarios. However, some challenges were observed in complex cases, such as small or unclear fires, where some models, especially YOLOv5, struggled to identify these fires. This is evident from the test results on a sample dataset, which showed the models' challenges in detecting minor fires and transparent smoke, especially under daylight conditions. These challenges highlight areas needing improvement, particularly in the YOLOv5 model.

Fig. 10. Sample predictions from the YOLOv5x model on test images.

When comparing the YOLOv8 and YOLOv5 versions across various variants (x, l, m), it is clear that YOLOv8 outperforms YOLOv5 in most conditions, both in terms of hyperparameter tuning and the overall performance of the different models. This superiority can be attributed to the enhancements made to the YOLOv8 architecture, making it more capable of effectively processing data. This contributes to improved accuracy and flexibility in fire and smoke detection compared to the older version.

Fig. 11. Sample predictions from the YOLOv8x model on test images.

The trained models demonstrated commendable performance in detecting fires across various scenarios. However, some challenges were observed in complex cases, such as small or unclear fires, where certain models, particularly YOLOv5, struggled to identify them. This is evident from the test results on a sample dataset, which showed the models' difficulties in detecting minor fires and transparent white smoke, especially under daylight conditions. These challenges highlight areas needing improvement, particularly in the YOLOv5 model.

When comparing the YOLOv8 and YOLOv5 versions across various variants (x, l, m), it is clear that YOLOv8 outperforms YOLOv5 in most conditions, both in terms of hyperparameter tuning and the overall performance of the different models. This superiority can be attributed to the enhancements made to the YOLOv8 architecture, which makes it more effective at processing data. This contributes to improved accuracy and flexibility in fire and smoke detection compared to the older version.

Regarding hyperparameters, experiments have shown that tuning these parameters plays a crucial role in improving model accuracy. It was found that most models achieve optimal performance with a learning rate (LR) of 0.001, a batch size (BS) of 16, and 40 training epochs. This is due to the similarities in model architecture and operation, where these values helped stabilize performance and achieve higher accuracy.

Moreover, the use of the Adam optimizer positively impacted the models' performance. Models utilizing this optimizer demonstrated stability and better performance, while those using the SGD optimizer showed increased loss indicators. This rise in loss may indicate the phenomenon of overfitting, which stems from the diversity and complexity of the fire and smoke detection problem. The Adam optimizer is distinguished by its ability to adapt the learning rate (LR) for each parameter independently, assisting in managing the complexities of diverse data. In contrast, the fixed learning rate of the SGD optimizer may not be suitable for this variability, leading to challenges in stabilizing the model and a higher likelihood of overfitting.

Overall, YOLOv8 models exhibited more stable performance and significant superiority compared to YOLOv5 models. Although neither model achieved perfect performance across all scenarios, YOLOv8 demonstrated greater potential in detecting fires and smoke, reflecting the advancements introduced in this version. It is anticipated that YOLOv8 will display even higher performance if training data or training configurations are further optimized, aligning with the developments observed in this version over YOLOv5.

This contrasts with a study referenced in [4], which demonstrated the effectiveness of the YOLOv5 model in detecting wildfires using frames from 31 videos specific to these scenarios. In contrast, our study showed that the YOLOv8 model excelled in detecting fires across a variety of scenarios, including forests and indoor and outdoor buildings under different lighting conditions. This performance discrepancy suggests that the YOLOv8 model may be more effective in handling diverse and complex data, while YOLOv5 may be better suited for specific applications, such as wildfires alone. This underscores the fact that model efficiency may vary depending on the type of application and the data used, necessitating further studies to determine the optimal model for each application.

Conclusion and Future Work

In this study, we meticulously examined and compared the performance of the YOLOv5 and YOLOv8 versions by rigorously training the models (XML) on a dataset that simulates various real-world scenarios. This analysis encompassed critical factors, including accuracy, recall, F1 score, and mAP50, as well as training times and the number of epochs required to achieve optimal recall. The study yielded insightful distinctions between the two models, with YOLOv8 consistently outperforming YOLOv5 across standard metrics, underscoring the effectiveness of the enhancements in this version. Despite this superiority, significant challenges remain, particularly in detecting smoke and fires under complex lighting conditions or in natural scenes with similar cases and limited data sizes. This highlights the necessity for larger and better datasets to enhance performance.

For future work, extending the evaluation to diverse datasets encompassing scenarios such as wildfires and industrial fires could provide substantial benefits. This approach may assist in identifying the specific conditions under which each model excels. Such efforts will contribute to improving early detection accuracy for fires and smoke, bolstering the effectiveness of environmental protection strategies and disaster mitigation.

Furthermore, exploring hyperparameters and model architectures in future work will be pivotal for performance enhancement. Focusing on developing models like YOLOv9 or advanced iterations of YOLOv8 could lead to significant advancements in computer vision and environmental monitoring applications. Integrating these models into real-world environments using thermal cameras and video surveillance may provide effective solutions for early detection scenarios, representing a promising direction for enhancing public safety and safeguarding property and lives.

References

1. Chaudhary, Muhammad T., and Awais Piracha. “Natural Disasters—Origins, Impacts, Management.", Encyclopedia, 1(4), October 2021, pp. 1101-1131. https://doi.org/10.3390/encyclopedia1040084.

2. Barmpoutis, Panagiotis, Periklis Papaioannou, Kosmas Dimitropoulos, and Nikos Grammalidis. “A Review on Early Forest Fire Detection Systems

Using Optical Remote Sensing.", Sensors, 20(22), November 2020, 6442. https://doi.org/10.3390/s20226442 .
3. Avazov, Kuldoshbay, Mukhriddin Mukhiddinov, Fazliddin Makhmudov, and Young Im Cho. “Fire Detection Method in Smart City Environments Using a Deep-Learning-Based Approach.”, Electronics, vol. 11, no. 1, January 2022, 73. https://doi.org/10.3390/electronics11010073
4. E. Casas, L. Ramos, E. Bendek, and F. Rivas-Echeverria, “YOLOv5 vs. YOLOv8: Per-formance benchmarking in wildfire and smoke detection scenarios,” Journal of Image and Graphics, vol. 12, no. 2, pp. xx-xx, 2024.
5. Saydirasulov, Norkobil S., Mukhriddin Mukhiddinov, Oybek Djuraev, Akmalbek Ab-dusalomov, and Young-Im Cho. “An Improved Wildfire Smoke Detection Based on YOLOv8 and UAV Images.”, Sensors, vol. 23, no. 20, October 2023, article 8374. https://doi.org/10.3390/s23208374 .
6. BEKKARI Lakhdar, KADI Nadhir, “Détection D'incendie Et De Fumée A L'aide De L'apprentissage Par Transfert” (Fire and Smoke Detection Using Transfer Learning), Université de Kasdi Merbah Ouargla, Ouargla, Algérie, 2022.
7. Y. Al-Smadi, M. Alauthman, A. Al-Qerem, A. Aldweesh, R. Quaddoura, F. Aburub, K. Mansour, and T. Alhmiedat, "Early wildfire smoke detection using different YOLO models," Machines, vol. 11, no. 2, p. 246, Feb. 2023. [Online]. Available: https://doi.org/10.3390/machines11020246
8. Md Tanzil Shahriar and Huyue Li, "A Study of Image Pre-processing for Faster Object Recognition," arXiv preprint, arXiv:2011.06928, 2020.
9. Hussain, Muhammad. “YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complemen-tary Nature toward Digital Manufacturing and Industrial Defect Detection.”, Machines, 11(7), June 2023, 677. https://doi.org/10.3390/machines11070677 .
10. M. L. Mekhalfi, C. Nicolo, Y. Bazi, M. M. A. Rahhal, N. A. Alsharif, and E. A. Ma-ghayreh, "Contrasting YOLOv5, transformer, and EfficientDet detectors for crop circle detection in desert," IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022.
11. J. K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, "PANet: Few-shot image seman-tic segmentation with prototype alignment," in Proc. IEEE/CVF International Confer-ence on Computer Vision (ICCV), Oct. 2019.
12. Mukhriddin Mukhiddinov, Akmalbek Bobomirzaevich Abdusalomov, and Jinsoo Cho, "A Wildfire Smoke Detection System Using Unmanned Aerial Vehicle Images Based on the Optimized YOLOv5," Sensors, vol. 22, no. 23, 2022, 9384. https://doi.org/10.3390/s22239384
13. Edmundo Casas, Leo Ramos, Eduardo Bendek, and Francklin Rivas-Echeverría, "As-sessing the Effectiveness of YOLO Architectures for Smoke and Wildfire Detection," IEEE Access, 2023.
14. Alsamurai, Mustafa Qays Fadhil. "Development and Implementation of YOLOV8-Based Model for Human and Animal Detection During Forest Fires." Unpublished mas-ter's thesis, Altınbaş University, Graduate School of Education, Istanbul, Turkey, 2023.
15. Talaat, Fatma M., and Hanaa ZainEldin. “An Improved Fire Detection Approach Based on YOLO-v8 for Smart Cities.”, Springer, Neural Computing and Applications, vol. 35, July 2023, pp. 20939–20954. https://doi.org/10.1007/s00521-023-08260-0 .

16. Douda, B. E. "Theory of Colored Flame Production." Report AD A 951815, 20 March 1964 .
17. Einhorn, I. N. "Physiological and Toxicological Aspects of Smoke Produced During the Combustion of Polymeric Materials.", Environmental Health Perspectives, vol. 11, June 1975, pp. 163-189. https://doi.org/10.1289/ehp.7511163 .
18. Kelly, Sage, Sherrie-Anne Kaye, Oscar Oviedo-Trespalacios. "What factors contribute to the acceptance of artificial intelligence? A systematic review.", Telematics and In-formatics, vol. 77, February 2023, article 101925. https://doi.org/10.1016/j.tele.2022.101925 .
19. E. G. Johnson, "Yolov8 Architecture vs Yolov5," Medium, Dec. 19, 2023. Available: https://medium.com/@EG_Johnson/yolov8-architecture-vs-yolov5-49d23b462ea6. Ac-cessed: Mar. 3, 2024, 10:00 AM
20. Zili Gui and Jianping Geng, "YOLO-ADS: An Improved YOLOv8 Algorithm for Metal Surface Defect Detection," Electronics, vol. 13, no. 16, pp. 3129, Aug. 2024 .
21. M. Chetoui and M. A. Akhloufi, "Object detection model-based quality inspection us-ing a deep CNN," in Proc. Sixteenth International Conference on Quality Control by Artificial Vision, July 2023. [Online]. Available: https://doi.org/10.1117/12.2689921
22. N. Aishwarya and R. V. Kumar, "Banana ripeness classification with deep CNN on NVIDIA Jetson Xavier AGX," in Proc. 2023 7th International Conference on I-SMAC, Kirtipur, Nepal, 2023, pp. 663–668. doi: 10.1109/I-SMAC58438.2023.10290326
23. Johnson, Justin M., and Taghi M. Khoshgoftaar. "Survey on Deep Learning with Class Imbalance.", Journal of Big Data, vol. 6, no. 1, article no. 27, March 2019, pp. 1-54. Open access, survey paper. https://doi.org/10.1186/s40537-019-0192-5 .
24. Rozada Raneros, Saúl. "Estudio de la arquitectura YOLO para la detección de objetos mediante deep learning." Directed by Ignacio de Miguel Jiménez, Universidad de Val-ladolid, Escuela Técnica Superior de Ingenieros de Telecomunicación, Valladolid, Spain, 2021. http://uvadoc.uva.es/handle/10324/45359 .
25. Mohammad Hossin and Md Nasir Sulaiman, "A Review on Evaluation Metrics for Data Classification Evaluations,"International Journal of Data Mining & Knowledge Man-agement Process (IJDKP), vol. 5, no. 2, March 2015, pp. 1. https://doi.org/10.5121/ijdkp.2015.5201