Sustainable Fire Detection in Smart Cities Using ResNet101V2 and Optimized Gradient-Boosting Method

Akshat Gaurav[1], Brij B. Gupta[2], Nadia Nedjah[3] and Kwok Tai Chui[4]

[1]Ronin Institute, Montclair, NJ, USA,
[2]Department of Computer Science and Information Engineering, Asia University, Taichung 413, Taiwan,
[3]State University of Rio de Janeiro, Brazil,
[4]Hong Kong Metropolitan University, Hong Kong SAR, China,

[1*]akshat.gaurav@ieee.org, [2]bbgupta@asia.edu.tw, [3]nadia@eng.uerj.br, [4]jktchui@hkmu.edu.hk

Abstract. Fire detection is a critical component in ensuring the safety of smart cities, where timely and accurate identification of fire incidents can prevent widespread damage. In this context, this work provides a sustainable and updated fire detection system for smart cities utilizing ResNet101V2 for feature extraction and Gradient Boosting enhanced using the Sea-Horse Optimization (Sea HO) method. With a precision of 0.96 and a recall of 0.95, it attained a better accuracy of 95\% than often used models such CatBoost, XGBoost, and SVM. Hyperparameter tuning was greatly aided by SeaHO, thereby improving the performance and resilience of the model. With minimum false alarms and great fire detection accuracy, the suggested method is perfect for real-time fire detection and guarantees safety in metropolitan surroundings.

Keywords: FFire Detection, Sustainability, Smart Cities, ResNet101V2, Gradient Boosting, Sea-Horse Optimization.

1   Introduction

In smart cities, fire detection is very important as it immediately affects public safety, infrastructure resilience, and emergency response capacity. The incorporation of sophisticated fire detection systems is crucial to reduce risks related with fire threats as urban environments becoming more complicated and densely inhabited.

Public safety is much enhanced by fire detection systems, especially those using artificial intelligence (AI) and machine learning (ML). Deep learning-based systems such YOLO (You Only Look Once) have showed promise in real-time smoke and fire detection. By use of video feeds from security cameras, these systems may detect possible fire breakouts at early stages, thus enabling quick intervention and so lowering the possibility of catastrophic incidents [1], [2]. The use of such technologies not only enhances the immediate reaction to fire events but also supports a more general strategy of urban safety management, which is very essential in highly populated smart cities[3], [4], [5], [6].

To guarantee public safety, guarantee strong infrastructure, and increase emergency response capacity, smart cities must therefore include advanced fire detection systems into their design. Adoption of these technologies will be essential in protecting communities from fire threats and guaranteeing a sustainable urban future as urban surroundings keep changing.

In this context, the contribution of this paper lies in the sustainable modernization of fire detection systems for smart cities. By utilizing ResNet101V2 for efficient feature extraction and optimizing Gradient Boosting with the Sea-Horse Optimization (SeaHO) algorithm, the proposed system enhances real-time fire detection accuracy while reducing false alarms. This approach contributes to sustainable urban safety infrastructure, offering an advanced solution that balances performance and resource efficiency in diverse smart city environments.

The paper is organized as follows: The Introduction outlines the motivation, problem statement, and contributions. Related Work reviews existing fire detection techniques and identifies gaps. The Proposed Methodology explains the data collection, ResNet101V2-based feature extraction, Gradient Boosting for classification, and Sea-Horse Optimization for hyperparameter tuning. The Experimental Setup and Results section details the dataset, evaluation metrics, and performance comparison with other models. Finally, the Conclusion and Future Work summarizes key findings and suggests future research directions for expanding the system to other smart city applications.

2   Related Work

Talaat et al.[7] propose a smart fire detection system (SFDS) based on the YOLOv8 algorithm, leveraging deep learning to improve real-time fire detection accuracy and reduce false alarms.

Tan et al.[8] reviews medical image segmentation methods based on deep learning, offering a comprehensive analysis of supervised, semi-supervised, and unsupervised techniques. Nedjah et al. [9] utilizes template matching for target detection in images through a co-design system, integrating a hardware coprocessor to accelerate the computationally intensive normalized cross-correlation process. Gao et al.[10] leverages steady-state visual evoked potentials (SSVEP) and the random image structure evolution technique to investigate the neural mechanisms of selective attention, particularly in recognizing serpentine forms. Almomani et al. [11] proposes a hybrid approach for cyberbullying detection on social media, combining deep learning models (e.g., InceptionV3, ResNet50, and VGG16) as feature extractors with machine learning classifiers (e.g., Logistic Regression and SVM).

Sharma et al.[12] propose an early fire detection system integrating wireless sensor networks, UAVs, and cloud computing to monitor environmental parameters and detect fires. Yar et al.[13] propose a modified YOLOv5s model for fire detection, improving on traditional deep learning approaches by addressing issues of high false alarm rates, model size, and time complexity.

Avazov et al.[14] propose an improved YOLOv4-based fire detection system designed to accurately detect even small sparks and raise alarms within 8 seconds of a fire outbreak. El-Hosseini et al.[15] propose a power-aware IoT-based fire detection

model that uses multi-functional sensors in smart cities, with a sleep scheduling approach to save sensor energy and minimize the number of active nodes. Khan et al.[16] propose a deep learning model, FFireNet, which leverages the pre-trained MobileNetV2 model for forest fire detection in smart city applications.

3   Proposed Work

The proposed fire detection system starts with data collecting from security cameras placed in smart cities, where real-time photos or videos are recorded and kept in a database (Figure 1). After processing this gathered visual data using the RexNet 101V2 model—which functions as the feature extraction component—key visual clues suggestive of fire from the input photos are found. Following the extraction of the salient features, they are input into a Gradient Boosting classifier tuned by the Sea-Horse Optimization (SeaHO) technique. Tuning the Gradient Boosting model's hyperparameters to enhance performance and detection accuracy falls to Sea HO. For smart cities, this mix of feature extraction, sophisticated classification, and hyperparameter tuning guarantees a fast, high-accuracy fire detection system.
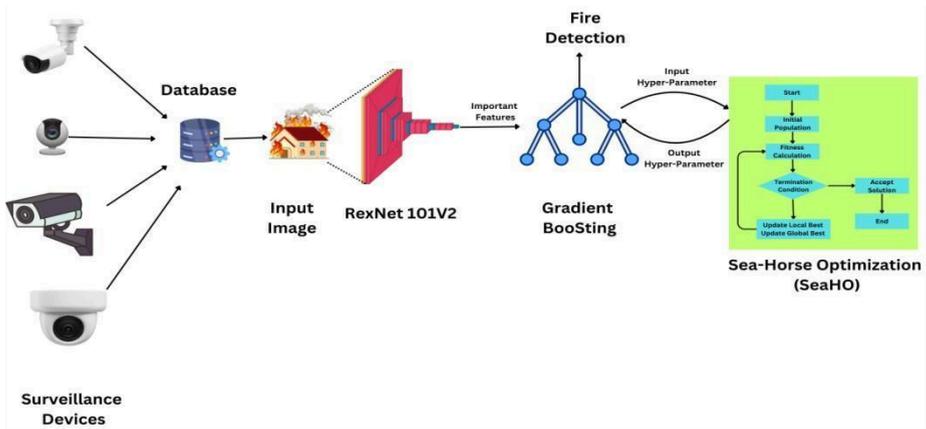

Fig.1. Proposed Model

3.1   Data Collection

Let the set of collected surveillance images be:

$$D = \{I_1, I_2, I_3, ..., I_n\} \tag{1}$$

where each image I_i is represented as a 3D matrix of pixel values:

$$I_i \in R^{h \times w \times c} \tag{2}$$

3.2   **Feature Extraction (RexNet101V2)**

The feature extraction using RexNet101V2 for each image I_i is represented as:

$$F_i = f_{RexNet}(I_i) = \sigma(W * I_i + b)$$

(3)

where:
W are the convolutional weights,
b is the bias term,
$*$ denotes the convolution operation,
σ is the activation function (e.g., ReLU).
For the entire dataset, the feature extraction becomes:

$$F = f_{RexNet}(D) = \{F_1, F_2, ..., F_n\}$$

(4)

3.3   **Hyperparameter Optimization (Sea-Horse Optimization)**

The SeaHO algorithm seeks to minimize the loss function $L(\theta)$ to find the optimal hyperparameters θ^*:

$$\theta^* = arg \min_{\theta} L(\theta)$$

(5)

The loss function $L(\theta)$ (e.g., cross-entropy) for binary classification (fire/no-fire) is:

$$L(\theta) = -\sum_{i=1}^{n}\left[y_i log\hat{y}_i + (1 - y_i)log(1 - \hat{y}_i)\right]$$

(6)

where
y_i is the actual label (fire = 1, no fire = 0),

$\hat{y}_i$ is the predicted probability of the image being a fire.

3.4   **Model Training (Gradient Boosting)**

Gradient Boosting combines decision trees T_t in each iteration t to minimize residual errors. The prediction at iteration t is:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \eta \cdot T_t(F_i)$$

(7)

where η is the learning rate and $T_t(F_i)$ is the prediction from the t-th decision tree.
The final prediction for an image I_i is:

$$\hat{y}_i = \sum_{t=1}^{T} \eta \cdot T_t\left(F_i\right) \tag{8}$$

where T is the total number of trees.

3.5   Prediction

For new test data, the prediction $\hat{y}_i$ for a test image I_i is:

$$\hat{y}_i = f_{GB}\left(f_{RexNet}\left(I_i\right)\right) \tag{9}$$

where $f_{RexNet}\left(I_i\right)$ is the feature extraction process from RexNet101V2 and f_{GB} is the trained Gradient Boosting model.Results and Discussion

3.6   Dataset Represented

Images from Kaggle were gathered for this research from two main classes: "NO_FIRE" and "FIRE." Scenes of indoor or outdoor environments devoid of any evidence of flames or smoke fall under the "NO_FIRE" class of photographs. Conversely, the "FIRE" class includes photos of active fire—that is, both regulated inside flames and more expansive outside fires. This binary classification lets the model distinguish between safe surroundings and those impacted by fire, therefore enabling the real-time fire detection in smart city infrastructure.

Figure 2 shows instances from both classes graphically. Images of a hotel room, airport, and outdoor surroundings free of fire abound in the top row "NO_FIRE" class samples. The bottom row has "FIRE" class examples, in which active flames are readily seen in a variety of settings, including home fires and burning trash.

Fig.2. Class Representation

3.7   Performance of Sea-Horse Optimization (SeaHO)

ResNet101V2 was used for feature extraction in this phase of our research because of its established capacity to gather hierarchical and detailed visual characteristics from

pictures, therefore making it especially fit for fire detection tasks. ResNet101V2 made it possible to extract high-dimensional features from the dataset, which were later trained a Gradient Boosting model from. Especially in classification problems like separating "FIRE" from "NO_FIRE," gradient boosting is a potent ensemble learning strategy that systematically improves weak learners to decrease mistakes, hence providing good prediction performance.   We tuned the hyperparameters of the Gradient Boosting classifier using the Sea-Horse Optimization (SeaHO) method to raise its performance even further. Inspired by the behavior of sea horses, particularly aimed to balance exploration (mostly seeking the solution space) and exploitation (refining the best solutions), SeaHO is a newly created metaheuristic. Finding the best hyperparameters that would improve the generalization capacity of the model while preventing overfitting depends on this balancing.
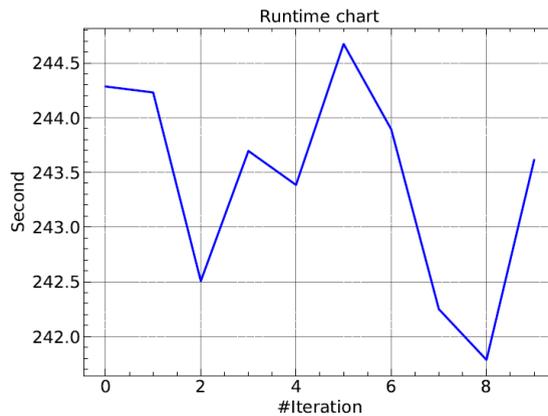

Fig.3. Runtime Measurement for SeaHO

Figure 3 (Runtime Chart) Tracking the time spent on every iteration of SeaHO throughout the optimization process, the runtime chart shows With an average run over iterations of 243 seconds, the very constant runtime shows that the method maintained computational efficiency all along. Practical applications where optimization must be done under acceptable time constraints—especially in real-time systems like fire detection in smart cities—demand this consistency.
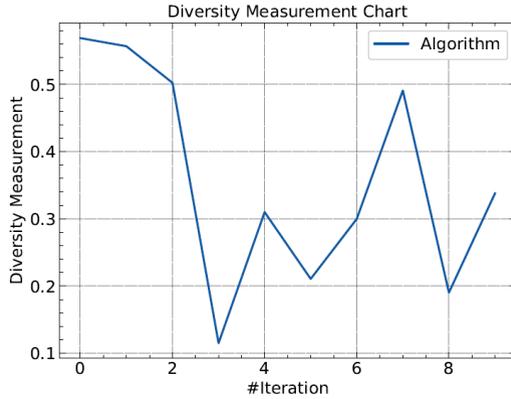
Fig.4. Diversity Measurement for SeaHO

Figure 4, the Diversity Measurement Chart, Diversity measurement captures the exploring behavior of the algorithm. The relative high diversity early in the process suggests that the method investigated a large range of hyperparameter space solutions. Diversity reduces after multiple repetitions, indicating that the method moved toward improving rather good results. Especially, consistent fluctuations in variety show that the method prevented early convergence by restoring exploration, therefore preventing local minima from catching it. Maximizing sophisticated models like Gradient Boosting depends on this adaptive exploration-exploitation strategy.

The chart of exploration against exploitation (Figure 5) offers understanding of how the SeaHO method strikes a compromise between its search approach. Since the method explores a large spectrum of hyperparameter values, initially exploration rules. The method moves toward exploitation as the iterations proceed, focusing on improving the most likely areas of the solution space. This shift emphasizes SeaHO's dynamic adaptability, which helps it to converge toward optimum solutions and prevent over-exploration, hence enhancing model accuracy and resilience.
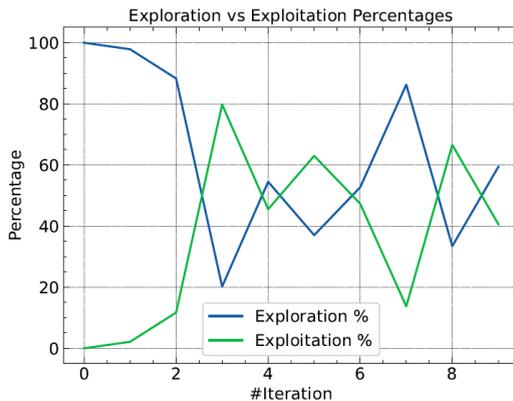

Fig.5. Exploration against Exploitation for SeaHO

3.8   **Performance of Gradient Boosting**

We trained the Gradient Boosting model on the dataset and assessed its performance after Sea-Horse Optimization (SeaHO) method hyperparameter adjustment. We used a confusion matrix and a classification report to evaluate its performance in separating fire from non-fire events as they provide all-encompassing understanding of the classification accuracy of the model.
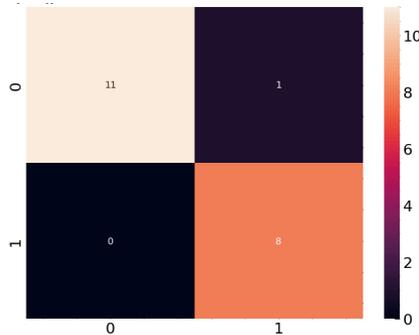

Fig.6. Confusion Matrix

Showed in Figure 6, the confusion matrix shows how well the model distinguishes "FIRE" from "NO_FIRE" events. The matrix shows that the model distinguished 8 cases of "FIRE" as true positives (TP) and 11 cases of "NO_FIRE" as true negatives (TN). One occasion, however, when the model misclassified a "NO_FIRE" picture as "FIRE," produced a false positive (FP). Especially, the model produced no false negative (FN) mistakes, thus all "FIRE" events were accurately classified as fire, a crucial feature of fire detection systems where neglect to find a fire might have grave consequences.

Presenting in Table 1, the classification report offers a more detailed analysis of the performance measures of the model including F1-score for every class, accuracy, and recall. The model attained a perfect accuracy of 1.00 for the "NO_FIRE" class, therefore ensuring that it was always accurate when it projected "NO_FIRE." With a recall for this class of 0.92, the model found 92% of the real "NO_FIRE" occurrences. With an F1-score of 0.96 this mix of accuracy and recall shows that the model is very accurate in identifying non-fire situations.

Table 1. Classification Report.

Heading level	Precision	Recall	F1-score
0	1.00	0.92	0.96
1	0.89	1.00	0.94
Accuracy			0.95
Macro vg	0.94	0.96	0.95
weighted avg	0.96	0.95	0.95

With a somewhat lower accuracy of 0.89, the model showed for the "FIRE" class that 89% of the "FIRE" forecasts were accurate. But the recall for this class was 1.00, hence the model accurately found every fire occurrence without producing any false negatives. In fire detection systems, this great recall is very vital as it guarantees that no real fires get missed. With an F1-score of 0.94, this class's model clearly does quite well overall in identifying fire events.

The confusion matrix and classification report both mirror the model's general accuracy of 95%. With a macro-averaged F1-score of 0.95 the model shows equivalent performance across both classes. with a similar vein, the weighted-average F1-score of 0.95 illustrates the resilience of the model even with somewhat skewed data.

3.9   Comparative Analysis

We evaluated the performance of the suggested method by means of many generally used machine learning models, including SVM, Logistic Regression, Random Forest, AdaBoost, Extra Trees, K-Nearest Neighbors, Naive Bayes, LightGBM, XGBoost, and Catboost. Among the assessment measures are computing time, accuracy, precision, recall, F1-score. Across all measures, the suggested method—which makes use of ResNet101V2 for feature extraction and Sea-Horse Optimization (SeaHO) for tuning Gradient Boosting—perfaced rival models.
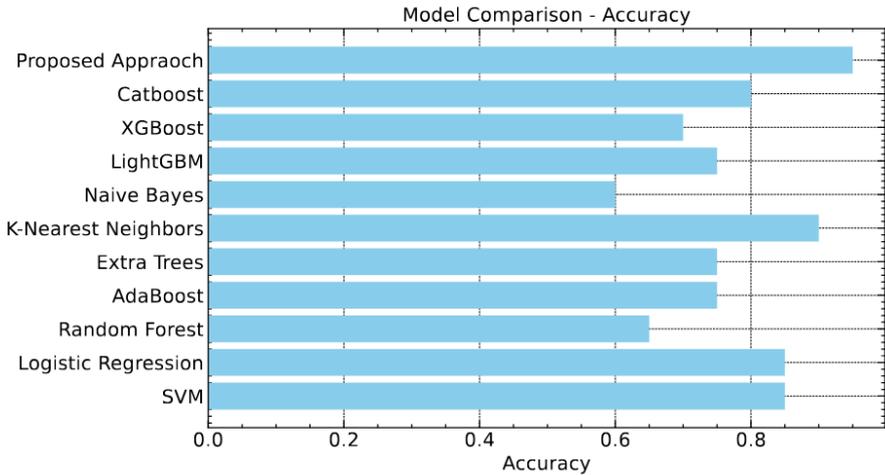

Fig.7. Accuracy Comparison

As shown in Figure 7, the suggested method outperformed all other models, including CatBoost (80%) and K-Nearest Neighbors (90%), attesting to the maximum accuracy at 95%. Out of all the cases, accuracy shows the proportion of properly categorized events, therefore proving the great efficiency of the suggested method in separating "FIRE" from "NO_FIRE."
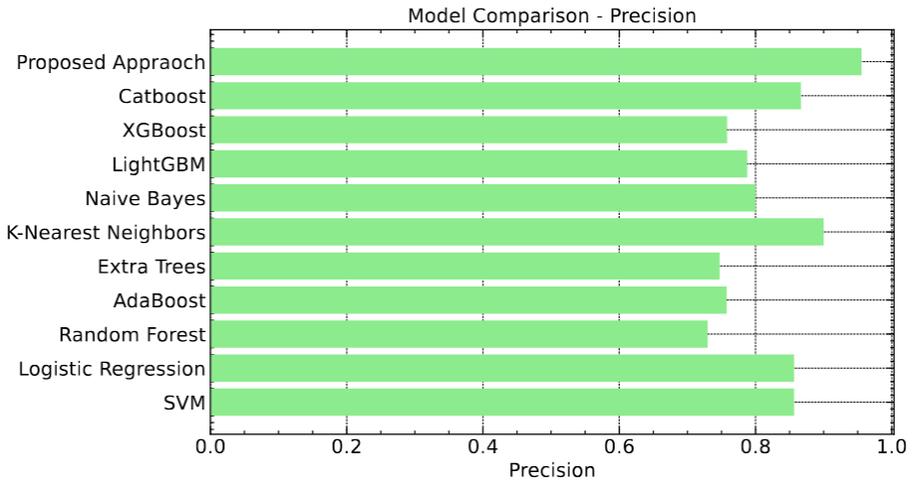
Fig.8. Precision Comparison

Precision gauges the model's capacity to accurately identify positive predictions—that is, fire situations. The suggested method attained an accuracy of 0.96, more than CatBoost (0.87) and SVM (0.86), as Figure 8 shows. This suggests that, in real-world applications, where false alarms must be avoided, the suggested method reduces false positives better than the others, therefore increasing its dependability.
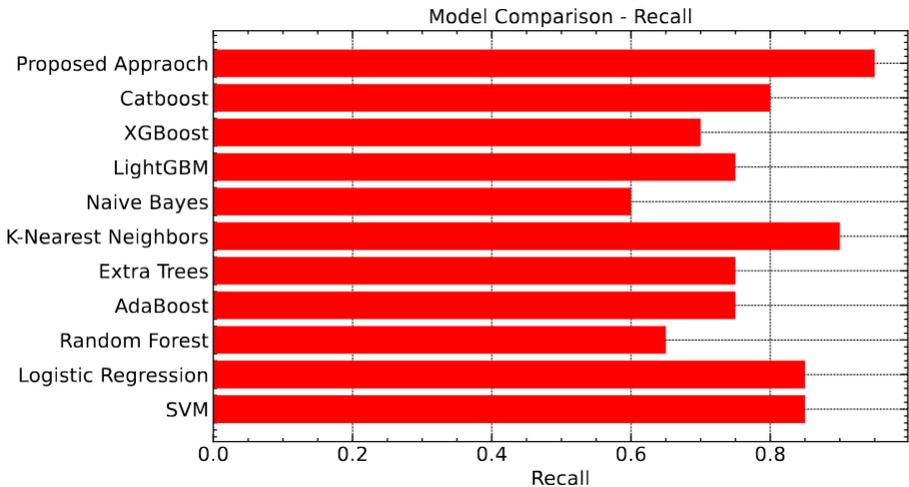

Fig.9. Recall Comparison

Recall: With a recall of 0.95, the recall comparison—shown in Figure 9—showcases the capacity of the proposed strategy to identify real fire cases. Including K-Nearest Neighbors (0.90) and CatBoost (0.80), this exceeded all other models. Fire detection depends on high recall; hence the model must find almost all fire events without excluding any important events.

Figure 10 shows the F1-score, which strikes a mix between recall and accuracy. Once again surpassing all other models, the suggested strategy obtained an F1-score of 0.95. Combining the capacity to prevent false positives with the capacity to properly identify fires, this score shows the general efficacy of the model.
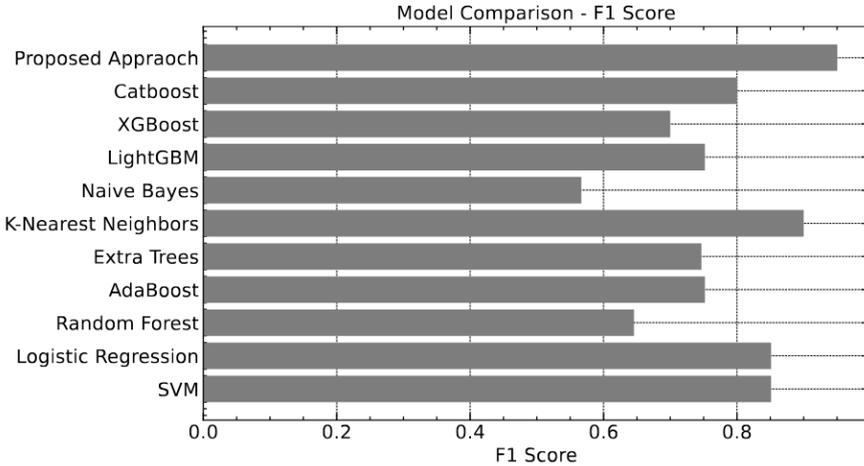

Fig.10.F1-Score Comparison

ResNet101V2 for feature extraction and Gradient Boosting, improved using SeaHO, is a very successful solution for fire detection in smart cities overall, as shown by the suggested technique outperforming competing models across all measures. It offers not only the finest balance between accuracy, recall, and F1-score but also the best purity of accuracy. Furthermore, while the computational time was more than certain models (like Logistic Regression or Random Forest), the better performance in detection justified the trade-off in time for vital applications including fire detection.

4   Conclusion

Using ResNet101V2 for feature extraction and Gradient Boosting improved using the Sea-Horse Optimization (Sea HO) algorithm, this work offered a sustainable and updated fire detection system for smart cities. Outstanding performance, with a 95% accuracy, 0.96 precision, and 0.95 recall, the model exceeded other state-of- the-art models like CatBoost, XGBoost, and SVM. Effective hyperparameter optimization by the SeaHO method improved model accuracy and resilience.These findings imply that the suggested method presents a dependable solution with few false alarms and is quite efficient for real-time fire detection. This approach may greatly increase urban safety and help smart cities to be proactive in fire control by combining cutting-edge machine learning methods with sustainable smart city architecture.

5   **Acknowledgement**

References

1. Almomani, A., Nahar, K., Alauthman, M., Al-Betar, M.A., Yaseen, Q., Gupta, B.B.: Image cyberbullying detection and recognition using transfer deep machine learning. International Journal of Cognitive Computing in Engineering 5, 14–26 (2024)
2. Avazov, K., Mukhiddinov, M., Makhmudov, F., Cho, Y.I.: Fire detection method insmart city environments using a deep-learning-based approach. Electronics 11(1), 73 (2021)
3. Dwivedi, R.K.: Density-based machine learning scheme for outlier detection insmart forest fire monitoring sensor cloud. International Journal of Cloud Applica- tions and Computing (IJCAC) 12(1), 1–16 (2022)
4. El-Hosseini, M., ZainEldin, H., Arafat, H., Badawy, M.: A fire detection modelbased on power-aware scheduling for iot-sensors in smart cities with partial cover- age. Journal of Ambient Intelligence and Humanized Computing 12(2), 2629–2648 (Feb 2021)
5. Gao, S., Cheng, Y., Mao, S., Fan, X., Deng, X.: Ssvep-enhanced threat detectionand its impact on image segmentation. International Journal on Semantic Web and Information Systems (IJSWIS) 20(1), 1–20 (2024)
6. Guo, M., Sheng, T., Shang, S.: The evaluation of building fire emergency responsecapability based on the cmm. International Journal of Environmental Research and Public Health 16, 1962 (2019)
7. Khan, S., Khan, A.: Ffirenet: Deep learning based forest fire classification anddetection in smart cities. Symmetry 14(10), 2155 (2022)
8. Nedjah, N., Cardoso, A.V., Tavares, Y.M., Mourelle, L.d.M., Gupta, B.B., Arya, V.: Co-design dedicated system for efficient object tracking using swarm intelligence-oriented search strategies. Sensors 23(13), 5881 (2023)
9. Park, S.: Smart fire safety management system (sfsms) connected with energymanagement for sustainable service in smart building infrastructures. Buildings 13, 3018 (2023)
10. Sharma, A., Singh, P.K., Kumar, Y.: An integrated fire detection system using iotand image processing technique for smart cities. Sustainable Cities and Society 61, 102332 (2020)
11. Talaat, F., ZainEldin, H.: An improved fire detection approach based on yolo-v8for smart cities. Neural Computing and Applications 35, 20939–20954 (2023)
12. Talaat, F.M., ZainEldin, H.: An improved fire detection approach based on yolo- v8 for smart cities. Neural Computing and Applications 35(28), 20939–20954 (Oct 2023)
13. Tan, J., Zhou, W., Lin, L., Jumahong, H.: A review of semantic medical imagesegmentation based on different paradigms. International Journal on Semantic Web and Information Systems (IJSWIS) 20(1), 1–25 (2024)
14. Vijayakumar, P., Rajkumar, S., Deborah, L.J.: Passive-awake energy consciouspower consumption in smart electric vehicles using cluster type cloud communication. International Journal of Cloud Applications And Computing (IJCAC) 12(1), 1–14 (2022)
15. Xie, Y., Li, P., Nedjah, N., Gupta, B.B., Taniar, D., Zhang, J.: Privacy protectionframework for face recognition in edge-based internet of things. Cluster Computing 26(5), 3017–3035 (2023)
16. Yar, H., Khan, Z.A., Ullah, F.U.M., Ullah, W., Baik, S.W.: A modified yolov5architecture for efficient fire detection in smart cities. Expert Systems with Appli- cations 231, 120465 (2023)