


Real-Time Fire and Smoke Detection and Classification of Video Frames in Smart Cities Based on You Only Look Once Version 10 Model



Mohammed Al-Abbasi^{1*}, Ekhlal Falih Naser², Tamarah Ayad Kareem¹

¹ Electro-Mechanical Engineering College, University of Technology, Baghdad 10066, Iraq

² College of Computer Science, University of Technology, Baghdad 10066, Iraq

Corresponding Author Email: Mohammed.S.Dawood@uotechnology.edu.iq

Copyright: ©2026 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/jesa.590502>

ABSTRACT

Received: 8 January 2026

Revised: 12 April 2026

Accepted: 27 April 2026

Available online: 31 May 2026

Keywords:

computer vision, deep learning, fire detection, IoT, You Only Look Once version 10

To avert significant loss of life and property, early and precise fire detection is essential. There are often delays and restricted coverage in traditional sensor-based fire detection systems. Advances in computer vision and deep learning have made video-based fire detection a competitive alternative. Intelligent city fires can have disastrous results, putting residents' lives in jeopardy and causing property damage. In order to improve accuracy and efficacy, this project aims to develop an intelligent system fire and smoke detection (ISFD) in smart cities. Three steps make up the suggested system for detecting smoke and fire in video frames. In the first stage, frames were extracted from the input video and then resized, normalized, and reduced. The second step involved extracting spots of interest from each frame using a Speeded-Up Robust Features (SURF) detector and descriptor. The You Only Look Once version 10 (YOLOv10) paradigm was used by the system in the third stage to identify the event's class. The technology employs deep learning algorithms to scan video streams in real time and identify events based on visual patterns and attributes. The ISFD strategy may be more cost-effective, generate fewer false alarms, and improve fire detection accuracy when compared to traditional methods. The two primary tiers of the suggested intelligent system for fire and smoke detection are the application layer and the Internet of Things (IoT). The suggested method gathers and interprets data in real time by utilizing the IoT layer. This allows for quicker reaction times and lowers the possibility of harm to people or property. The experimental results show that a high rate of precision, 95.9%, a recall of 98.4%, and an F1-score of 0.971 for every class, ISFD demonstrated cutting-edge results in terms of recall and precision.

1. INTRODUCTION

Our perspectives on Smart cities are radically changing sustainability, safety and urbanization. As smart cities become more prevalent worldwide, protecting citizens and their possessions becomes increasingly important [1]. Consequently, the application of traditional sensor-based fire detection is restricted to indoor environments. Video fire detection provides a plethora of visual fire information and eliminates distance limitations. It is really exciting that visual fire detection is being used in applications. Video flame detection and video smoke detection are two types of fire detection methods that utilize video technology, each based on the specific elements being detected [2]. A method used color information to derive flame outlines and provided flame identification in the spatiotemporal domain. A method for identifying fires using grayscale images taken in tunnels was implemented [3].

It is crucial to have a dependable and effective early fire detection system. Due to the growing trend of urbanization and an increased awareness of the importance of safety, it has become the top priority in smart cities. Early detection of containment and fire can reduce property damage and save

lives. However, this project must address problems such as the unpredictable nature of fire, the requirement for continuous surveillance, and the enormous amounts of data produced by smart cities [4]. To identify fires early, scientists and planners have built vision-based detectors of fire (VFDs) in addition to fire sensors that react to temperature, gas, solid, sound, or flame [5]. The earliest way of detecting fire was manual inspection, which involves assigning individuals to periodically check possible fire spots. However, as civilization has evolved, the drawbacks of manual inspection—such as sluggish reaction times and hazards in the early stages of fires—have grown more obvious, rendering it inappropriate for the demands of the modern world [6]. Closed-circuit television (CCTV) has been widely installed in buildings and public areas as a result of urbanization, making it possible to monitor fire threats at a reasonable cost. Fire detection techniques may potentially be a part of these monitoring systems [7]. Among one-stage algorithms, the YOLO series is notable for its straightforward architecture, high accuracy, and ability to perform real-time detection [8]. Ghost convolutions were utilized to lower the computational load imposed by the second, tiny object identification layer that was added to the YOLOv5 network. The network was able to improve its ability

to detect small things, at a comparatively minimal computing cost, like dim flames [9]. The main structure of the YOLOv3 Darknet-53 is found to have width, resolution, and depth mismatches. They replaced Darknet-53 with an improved Efficient Net and proposed an improved model based on YOLOv3 with faster performance, fewer parameters, and greater detection capabilities [10]. The YOLOv8 network serves as the foundation for the proposed methodology for creating a ship fire detection network. The researchers employed channel and spatial reconstruction convolutions and an extra object-detection layer to decrease duplicate characteristics [11]. These modules could make better the capacity to recognize little items. However, the second detection layer's detection capability was constrained by the harsh hardware conditions aboard the ship [12]. By Utilizing the network of Convolutional Neural Network (CNN), simple linear iterative clustering (SLIC) for segmentation of smoke picture, and density-based spatial clustering of applications with noise (DBSCAN) that may attain faster observation. However, due to its low rate of FPR, which mention worthy model allergy, the offer procedure demanded further more development [13]. Although deep learning methods for smoke detection have shown a lot of potential, using edge equipment for detection has three drawbacks. In the first place, the large amount of parameters and expensive requirements of hardware for huge network models make it strenuous to deploy them for workable activities and meet real-time smoke detection needs. Second, current lightweight samples can discover smoke faster under the similar situations as models with a large network, but their detection accuracy is often significantly worse. When recognizing things with thin characteristics, such as smoke, the fusion of features is usually insufficient, this lowers the detection accuracy. The accuracy and detection speed are therefore performing at disparate levels, which is a problem. Third, during the initial statuses of fire of the forest, a type of smoke called "small smoke" is produced. It is characterized by its thinness and small volume. It cannot effectively extract information because thin and little smoke can only extract a limited number of characteristics. It is additional challenging to discover than regular smoke, which has formerly taken on shape and is vulnerable to disturbances like lens flaws. This boosts the issue of UAVs taking noisy pictures midst missions' discovery, which could lead to missed discovery. An enhanced detection of fire technique for smart metropolises focused on the You Only Look Once version 10 (YOLOv10) model was suggested in order to address the main problems associated with earlier methods involving deep learning. This approach produced the following advantages.

1. Improvement in accuracy: The proposed approach may improve the accuracy of fire detection in intelligent cities when compared to traditional methods. This might be achieved by utilizing the advantages of deep learning algorithms, such as YOLOv10, to recognize fire-specific traits that might be challenging to recognize using traditional image processing methods.

2. Real-time recognition of object: The YOLOv10 algorithm is well known for its speed and ability to identify objects. Therefore, the recommended approach is perfect for smart city applications where prompt fire detection is essential.

3. Reduced false alarms: By using deep learning to comprehend fire-specific characteristics, the proposed method may be able to reduce false alarms, which are common in traditional fire detection systems. It is possible to minimize false alarm expenses and prevent needless emergency

reactions.

4. Effectiveness of Cost: Since the suggested method may be used with inexpensive cameras and technologies, it may be less expensive than traditional fire detection methods since it doesn't require pricey fire detection equipment.

5. Large dataset: This methodology utilized a huge dataset that includes three various kinds of smoke and fire scenarios in addition to usual situations involving more people, unlike other approaches that rely on a small number of datasets. The collection includes real-world images and videos collected from different sources. The collection covers a wide range of fire scenarios, such as tiny and huge fires, inside and outdoor flames, and high and low light levels. A deep neural network extracts crucial information from huge datasets in order to produce accurate predictions and reduce overfitting.

6. Research on deep learning for fire and smoke warning has been ongoing, with target detection algorithms in particular demonstrating encouraging outcomes. Based on the You Only Look Once (YOLOv8) model, a high-precision, lightweight improvement is developed to improve the model's capacity to detect smoke and fire in different environments. It uses partial convolutions to streamline the model and adds an attention block to acquire cross-space learning capabilities. Additionally, a variation of the neck network is used to achieve bidirectional feature fusion. The improved model achieved 77.8% precision and 65.3% recall [14].

The key contribution of this article is YOLOv10, which improves efficiency and accuracy in real-time, end-to-end object identification, especially by removing non-maximum suppression (NMS) and optimizing model design. It accomplishes this contribution through a number of significant improvements, including a rank-guided block architecture, spatial-channel decoupled down sampling, lightweight classification heads with depth-wise separable convolutions, and a dual assignment technique for NMS-free training. Thanks to these developments, YOLOv10 can now attain cutting-edge performance with less computational overhead, which makes it appropriate for a variety of real-time applications. The remaining sections of the search consist of related works, methodology, Experiments and results, and conclusion.

2. RELATED WORKS

Finding and classifying objects with the least amount of delay is the aim of real-time object detection, which is a crucial aspect for real-world applications. Recently, there has been an increase in the usage of deep learning-based techniques for smart city fire detection. Many strategies have been proposed, such as:

1. This article describes the development of a deep learning model for image-based fire and smoke detection. The third version of the Inception Convolution Neural Network, also referred to as the Inception-V3 model, was updated and applied in this situation. The fire image dataset, which includes smoke, has been subjected to the modified Inception-V3 in order to detect fire images. Now, this model has a new function of optimization that efficiently diminishes the cost of computational. The accuracy of this approach was 93.92% [15].

2. The prototype of a Video Surveillance Unit (VSU) that uses two embedded Machine Learning (ML) algorithms and a low-power device to detect and alert users to the

existence of forest fires is shown in this work. The ML models use picture and audio samples as inputs, respectively, to enable timely detection of fire. The primary finding is that, although the two models' results are similar when used separately, using them together yields superior accuracy (93.15%) and recall (92.30%) according to the suggested technique. Ultimately, the Long Range Wide Area Network (LoRaWAN) protocol is used to remotely signal each incident so that the responsible staff may respond quickly [16].

3. This work addresses the current research topic, which is using deep learning-based fire detection systems to achieve high accuracy rates while minimizing processing costs. The paper offers a method that makes use of the YOLOv8 algorithm, which successfully addresses the difficulty of generating models using a particular dataset as well as the ensuing training, validation, and testing procedures. A succinct assessment of the model's efficacy is given by its 79% fire detection precision and 91% recall. This makes it a promising technique for Internet of Things (IoT) monitoring systems to detect fire and smoke [17].

The following succinctly describes the primary justifications for not utilizing transformers in the suggested fire and smoke detection model:

1. While transformer-based systems require more memory and processing power, YOLOv10 is great because it uses low parameters (FLOPs, nano/small versions).
2. While transformer-based is suitable for real-time (NMS-free) but requires robust hardware, YOLOv10 is the best option for real-time (no NMS required).
3. YOLOv10 is more effective and converges more quickly in terms of training time. Transformer-based approaches require more memory and have slower convergence.

Kaggle functions as a data repository, offering hundreds of distinct datasets that are commonly utilized for contests or educational goals.

1. Specificity of Domain and High Diversity: Kaggle provides datasets on a variety of topics, including image classification, natural language processing, healthcare, and finance.

2. Collaboration and Community: Notebooks, discussions, and code examples (Kernels) are connected to each dataset, facilitating faster debugging and better techniques.
3. Applicability for Real-World: Since many datasets of Kaggle are chosen from problems of real-world such as fraudulent transactions' predicting, they are suitable for practical, applied research rather than being restricted to academic benchmarks.
4. Interoperability and Accessibility: Datasets are frequently available for free download.

Because of its extensive, context-aware data, COCO is regarded as a standard benchmark in computer vision. The COCO dataset's primary characteristics are:

1. Non-Iconic Images & Context: Unlike datasets with centered, isolated objects, COCO offers complex, daily situations with several objects interacting, requiring models to comprehend context rather than merely identify shapes.
2. Rich Annotation Density: It offers more than 330,000 photos, of which 200,000 have keypoints, bounding boxes, and segmentation masks (1.5 million object instances).
3. Multi-Task Capability: COCO is not restricted to a single task. It is utilized for:
 - Object Detection: Bounding box identification of objects.
 - Instance Segmentation: Defining specific objects.

3. MATERIALS AND METHODS FOR THE SUGGESTED METHODOLOGY

Three primary steps make up the recommended fire and smoke detection methodology: data collection and preprocessing in the first stage, feature selection using the Speeded-Up Robust Features (SURF) detector in the second stage, and model training and detection using YOLOv10 in the third stage. The specifics of the proposed system were illustrated in Figure 1.

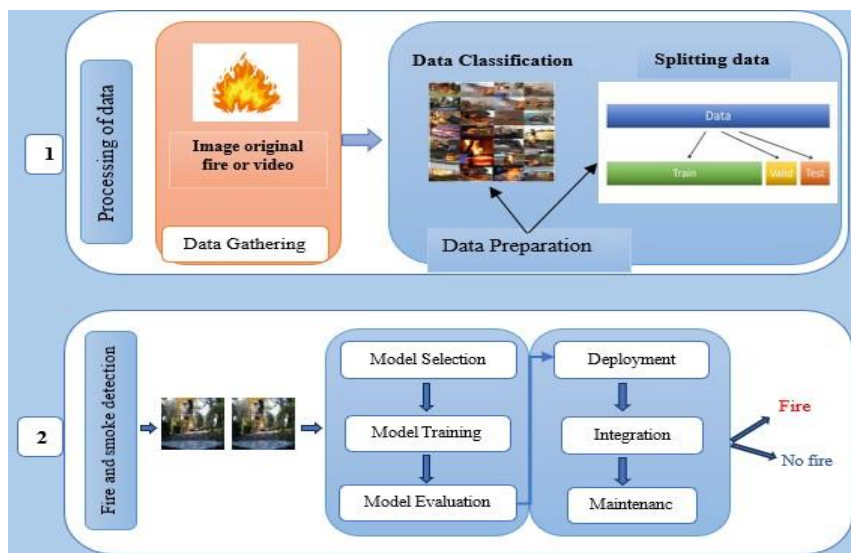


Figure 1. Proposed intelligent smoke and intelligent system fire and smoke detection (ISFD)

3.1 Dataset gathering and resizing

A bespoke dataset containing both fire incidents and non-

fire events that visually resemble fire was developed in order to efficiently train and assess the fire detection system. The method of preparing the dataset included the following crucial

steps:

1. Collection of Data

Video clips of non-fire, smoke, and fire were gathered from a variety of public sources, including as YouTube, open-source repositories, Kaggle datasets of fire detection, and personal fire films. The video clip shows a variety of situations, such as industrial flames, forest fires, domestic fires, and intricate backgrounds enhanced with lighting effects.

2. Augmentation of Data

A variety method of augmentation the data were applied to every frame in order to increase dataset diversity and strengthen the model's capacity for generalization. These methods involve flipping both horizontally and vertically, rotating ($\pm 15-30$ degrees), and adjusting brightness and contrast ($\pm 20\%$). Scaling methods like zooming in and out by up to $\pm 10\%$ were also used. By simulating variations in illumination, camera angles, and environmental factors, these adjustments strengthen the trained model's resilience.

3. Resizing

Before being fed into the model, images or frames are reduced to standard dimensions, such as 224×224 . In order to ensure uniform processing and performance across the neural network, this preserves input size homogeneity.

4. Normalization

Faster convergence during training and increased model accuracy are made possible by normalizing the image pixel values. By scaling the numbers to a range of 0 to 1, the model is able to handle the data more quickly.

5. Reduction of Noise

Noise in raw photos can reduce model performance. To overcome this, a technique of median filtering is employed to smooth the images and remove extraneous noise, resulting in more accurate input for the detection system.

6. Balancing of Data

Equal representation of non-fire and fire photographs in the dataset is necessary for the model to be trained effectively. Data balance approaches are used to guarantee that the training data is not biased toward a single class, hence boosting the model's capacity to generalize.

Impulse noise sometimes referred to as noise of 'salt-and-pepper', can be eliminated effectively via utilized median filter. YOLOv10 utilizes a mix of sophisticated mechanisms of spatial channel, loss functions of shape-aware, and deep learning on extensive datasets with variety situations of illumination differentiate among fire and lighting of background (like headlights, sun glare or streetlights). YOLOv10 examines the texture, motion, and behavior of flickering flames over time.

Through an enhanced architecture of (NMS-free) ("Non-Maximum Suppression") that increases speed of real-time inference and enhances extraction of feature to handle occlusions and complex backdrops, YOLOv10 tackles high-density movements of pedestrian and interference of detection of fire. Consistent dual-label assignment for dense scenes and specialized feature fusion techniques that distinguish pertinent items from background noise are its key superiority in these appointments.

3.2 Feature extraction

In order to elicit the interest points (features) from the input image and capture pertinent data at different scales, the proposed methodology utilized a SURF detector with the YOLOv10 backbone network. A robust detector for local

features, the SURF technique is an extractor used in many computer vision applications, including object identification and 3D reconstruction. SURF is a better approach that works well for real-time implementation. The SURF algorithm's vector descriptor was called to detect interest locations, and this process was founded on the assumption of a scale space [18].

1. Integral Images

An integral image can be utilized to quickly calculate Hessian's determinant blob detector, which is one of the integer approximations used in the SURF technique. Box-kind convolution filters may be calculated quickly thanks to it. Equation 1 provides an illustration of the invasion of an integral image ($I_{\Sigma}(x)$) at $(x = (x,y)T)$.

$$I_{\Sigma} = \sum_{i=0}^{i<x} \sum_{j=0}^{j<y} I(i,j) \tag{1}$$

2. Hessian Matrix-Based Interest Points

Because of its high accuracy performance, a detector can rely on the Hessian matrix. At locations where the determinant was at its maximum, it recognized a blob as a structure. Eq. (2) was used to determine the matrix of Hessian ($H(x, \sigma)$) in (x) at scale (σ) given a point $(x = (x; y))$ in an image (I) .

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \tag{2}$$

The Gaussian second-order derivative ($\partial^2/\partial x^2 g(\sigma)$) in the image (I) at a point (x) is convolved by $L_{xx}(x, \sigma)$, and Gaussians are optimal for scale space analysis. It must be clipped and discretized in the program, as shown in Figure 2, on the left side.

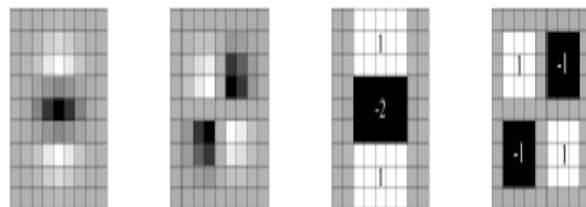


Figure 2. Left to right Gaussian second order partial derivative for L_{xy} (xy -direction), L_{yy} (y -direction) and, respectively [18]

The blob restraint map is calculated at the lowest scale with the (9×9) box filters shown in Figure 2, which approximate a Gaussian distribution with $\sigma = 1.2$ [18].

3. Scale Space Representation

In order to compare the points of interest that must be established at different scales-not least because of the correspondences that are frequently searched for-scale spaces are typically represented as a pyramid of an image. The photos are repeatedly subsampled for greater level pyramid achievement after being smoothed using a Gaussian technique. It is not necessary to apply the same filter repeatedly to the results of a previously filtered layer since integral pictures and box filters are used. Instead, box filters of any size are applied to the main image at precisely the same speed, even in parallel. Consequently, a scale space was examined by upscaling the filter's size rather than repeatedly shrinking an image's size, as

seen in Figure 3. The result of the (9×9) filter is regarded as the first scale layer and can be compared to the scale ($s = 1.2$). The following layers were obtained by filtering the image using ever larger masks while accounting for the discrete type of integral images and the structure of the specific filters [19].

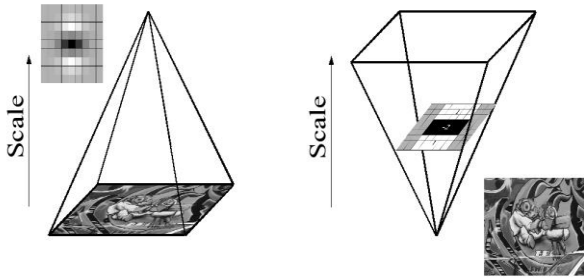


Figure 3. Instead of reducing iteratively the size of an image left, the use of integral-images permits the upscaling of the filter at fixed cost right [18]

4. Interest Point Localization

In order to locate points of interest in a picture and through scales, a non-maximum suppression in the $(3 \times 3 \times 3)$ neighborhood was used. Then, within the scale, the Hessian matrix determinant maxima are interpolated. In this case, the scale difference between the first layers of each octave is rather substantial, making the interpolation of a scale space especially important. Using the Fast-Hessian detector, a paradigm of the identified places of interest is shown in Figure 4 [19].

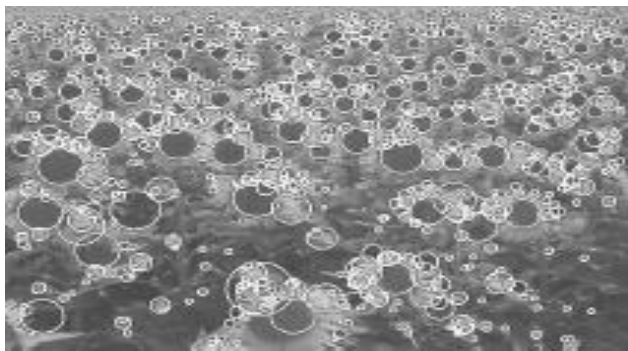


Figure 4. Points of interest detected in a sunflower domain
Note: This type of scene displays the quality of the features that are extracted via employing the Hessian detector [18]

A hybrid model (SURF+YOLOv10) that aims to improve detection quality on small or sparse objects while frequently decreasing the overall inference speed is typically created by adding SURF to the YOLO family or by combining similar handcrafted, robust techniques for feature elicitation with deep learning.

Combining these strategies has the following key effects:

1. Considerably better for detection of small objects: SURF approach enhancement aids in the identification of fine-grained details, which are essential for spotting small, far-off, or hazy targets.
2. Improved understanding of context: Adding strong local characteristics enables better classification in difficult or cluttered backgrounds, even while YOLO excels at capturing high-level semantic information.
3. Greater Precision on Particular Datasets: Combining

YOLO with conventional detection of feature SURF helps maintain high mean Average Precision (mAP) in pictures of rare situations in photography.

3.3 Model training and detection using You Only Look Once version 10

Deep learning approaches outperform conventional methods using object detection for choosing fire detection models. YOLOv10 was used in the suggested system. A model designed for real-time object detection aims to identify and classify fire in images [20]. The effectiveness of YOLOv10 is influenced by the tradeoff between accuracy and speed, as well as the capabilities of the hardware used. Training is carried out utilizing labeled datasets that include both fire and non-fire instances. Cross-validation is used to ensure model reliability.

The following procedures are taken to aggregate the detection outputs from YOLO10 utilizing a decision fusion mechanism:

Procedure 1: The system initiates a fire warning if the model independently detects fire with confidence over a predetermined level.

Procedure 2: Temporal consistency is evaluated over multiple frames to prevent false alarms caused by fleeting false positives.

Procedure 3: Alerts are communicated by a message or notification module in the surveillance system.

A backbone, a Path Aggregation Network (PAN), and prediction heads make up YOLOv10's architecture, which is seen in Figure 5. Features from the input image are extracted by the backbone, and the PAN efficiently gathers and disperses those features across the network. For classification and regression (bounding box) tasks, YOLOv10 presents dual label assignments with two prediction heads: the one-to-many head and the one-to-one head. To enhance prediction accuracy across various scales and scenarios, a Consistent Match Metric is utilized to refine label assignment.

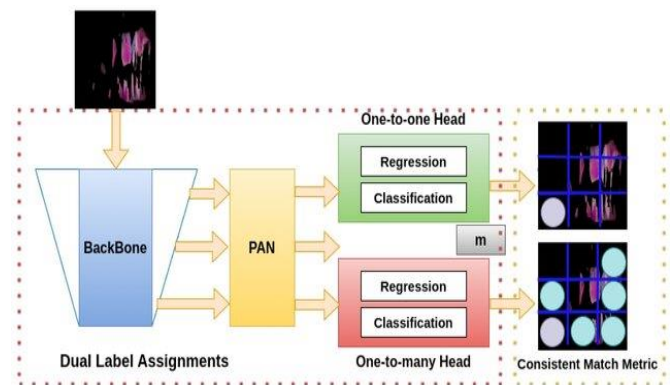


Figure 5. The structure of You Only Look Once version 10 (YOLOv10) [21]

4. SYSTEM ARCHITECTURE AND EXPERIMENTAL SETUP

The intelligent cities project's main objective is to make systems and applications smarter. To achieve this, different requirements and characteristics are combined, such as (a) a secure, scalable, resilient, and open-access infrastructure; (b) an architectural approach that is centered on the citizen or user; (c) the capability to tag, carry, store, wear, share, and retrieve

a large amounts of public and private data, enabling knowledge access at any time and from any location; (d) the ability to perform analytical and integrative application-level tasks; (e) advanced physical and network infrastructure that transfers vast amounts of diverse data and makes complex, remote services and applications possible. A framework must be used to express and arrange the key components, relationships, and data flow of transdisciplinary intelligent cities projects. The IoT layer and the application layer are the two main levels that make up the proposed intelligent system fire detection architecture, as shown in Figure 6.

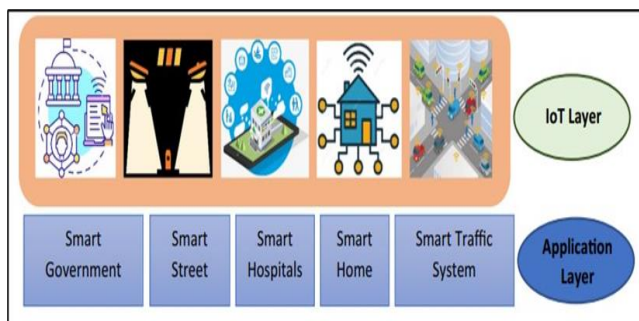


Figure 6. The suggested intelligent system framework

4.1 IoT layer

Physical objects, sensors, and actuators that are linked to the Internet and gather and share data comprise the IoT layer of an intelligent city. Sensors identify variations in the environment, effectors interact with the environment, and actuators move and control systems. To detect patterns, spot anomalies, and predict future events, sensor data is processed, saved, and examined.

4.2 Application layer

One essential component of the application layer is the recommended architecture for an intelligent system that detects fire. The ISFD can discover hazards of fire in a range of places, including public areas, hospitals, government buildings, residential communities, and roads, to safeguard property and people. The application layer is largely responsible for the ISFD's ability to handle a broad range of applications, including intelligent government, intelligent homes, intelligent streets, intelligent hospitals, and intelligent traffic systems.

4.3 Choosing simulation parameters

Table 1. Configuration of the environment

Operating System	CPU	GPU	Programming Language
Windows 10	Intel i5-2330M	T4 GPU	Python 3

To determine the best values for each hyperparameter, we conducted an experimental test. The simulation's input image was configured to be 224×224 pixels in size. After extensive experimentation with different batch sizes and rate of learning values, the parameters chosen to simulate the purposes of training are shown in Table 1. Table 2 illustrates the simulation parameters for the YOLOv10 Model.

Table 2. You Only Look Once version 10 (YOLOv10) parameters of the simulation

Parameters	Value
Input size	224×224
Optimizer	Stochastic gradient descent
Loss	Binary cross-entropy
Rate Learning	0.01
Batch Size	32
Epochs	40
Steps per epochs	100
Validation steps 10	100

5. MODEL EVALUATION

For fire detection quality, the proposed methodology calculates the F1-score recall, and precision. To compute the precision of a proposed method, it was utilized in Eq. (3) [22]. To compute the recall of the proposed method, it was utilized in Eq. (4). Also, to compute the F1-score of the proposed method, it was utilized in Eq. (5) [23].

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$F1 - Score = \frac{Precision \times Recall}{Precision + Recall} \times 2 \quad (5)$$

True Positive (TP): The number of cases that are correctly categorized as positive.

False Positive (False Positive - FP): The number of cases that are incorrectly categorized as positive (when the actual status is negative).

True Negative (True Negative - TN): The number of cases that were correctly categorized as negative.

False Negative (False Negative - FN): The number of cases that are incorrectly categorized as negative (when the actual state is positive).

The classification of things in the pictures—fire, smoke, and others—was done using labeling software. The txt format was used to save labels.

As indicated by the ratio in Table 3, the data set was split at random into training, validation, and testing sets.

Table 3. Details of the dataset

Dataset	Train	Val	Test	Total
Image	4263	660	530	5426

In order to meet real-time necessity, YOLOv10 utilized. The two main measures are mAP@0.5 and mAP@0.5:0.95. In order to ensure that the model correctly locates minor sources of fire, mAP50-95 offers a tighter estimation than mAP50, which analyzes discovery at a 50% threshold of "intersection over union (IoU)". Acceleration of hardware (like "NVIDIA GPUs") to test speed of inference ("FPS") and tiny object recognition performance in the detection of fire via customized estimation scripts.

6. RESULTS AND DISCUSSION

In this section, the experimental results of the proposed methodology are discussed and explained. The proposed method was implemented using Python 3 language. Different of videos and images were used to evaluate the proposed method, taken from the GitHub, Kaggle [24], and websites of news and social media or by filming with certain sensors or cameras. A number of fire, smoke, and human detection scenarios are provided in the collection; they include small- and large-scale fires, indoor and outdoor fires, low- and high-light circumstances, and ordinary fire-free settings. In order to overcome the low count of pixel of "far-off flames", identification of tiny object is specifically addressed by measuring "mAP" on tiny objects, frequently employing upgraded designs like heads of extreme detecting. YOLOv10 has better real-time performance than earlier generations, with inference times frequently less than 15 ms per image on contemporary GPUs, according to recent fire detection tests. To improve feature matching, YOLOv10 is typically combined with conventional methods like SURF, which frequently results in somewhat higher latency but improved detection of small, far-off fire sources. There are 5426 photos in the collection, comprising photographs of people and other things, images of flames and smoke, and images of typical situations devoid of smoke or fire. 4263 and 530 of the dataset are designated for training and testing, respectively,

verification of correctness 660. Considering how big the data set is, the suggested approach has demonstrated its superiority in accurately identifying flames and smoke. The dataset contains color videos and images. The proposed method includes several steps.

Step 1. After uploading the video, all key frames were elicited as illustrated in Figure 7, standardized the size of each frame to 224×224 pixels. Step 2. The points of interest were elicited in second step via applying the SURS detector and descriptor. The empirical result of this stage can be illustrated in Figure 8. Step 3. Furthermore, it has extremely high accuracy when validating and generalizing fresh pictures. We trained the YOLOv10 model on our dataset utilizing learning of transfer, which involves initializing a model using weights already learned on a dataset of COCO and fine-tuning it there. Using a batch size of 16, the model was trained across 40 epochs at an initial learning rate of 0.01. The YOLOv10 setup settings are listed in Table 2.

At Google Colab, the GPU platform was used to develop, train, and evaluate the proposed model. According to test data, our model has an accuracy of 82.1% for fire detection and an accuracy of 97.7% for smoke and detection other and has an accuracy of 98%, with an overall average accuracy of 95.4% (mAP) across all categories. The results of suggested methodology for the detection phase on a sample of frames can be shown in Figure 9.



Figure 7. Samples of images (frames) that contain fire, smoke, and no fire



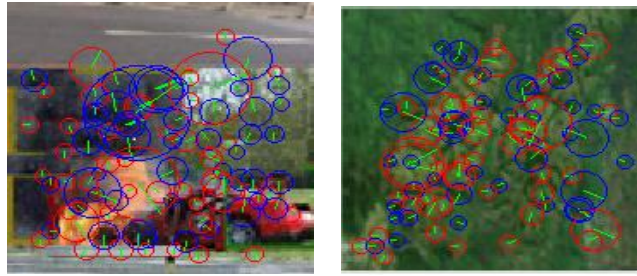


Figure 8. Samples of images (frames) that resulted from the Speeded-Up Robust Features (SURF) detector

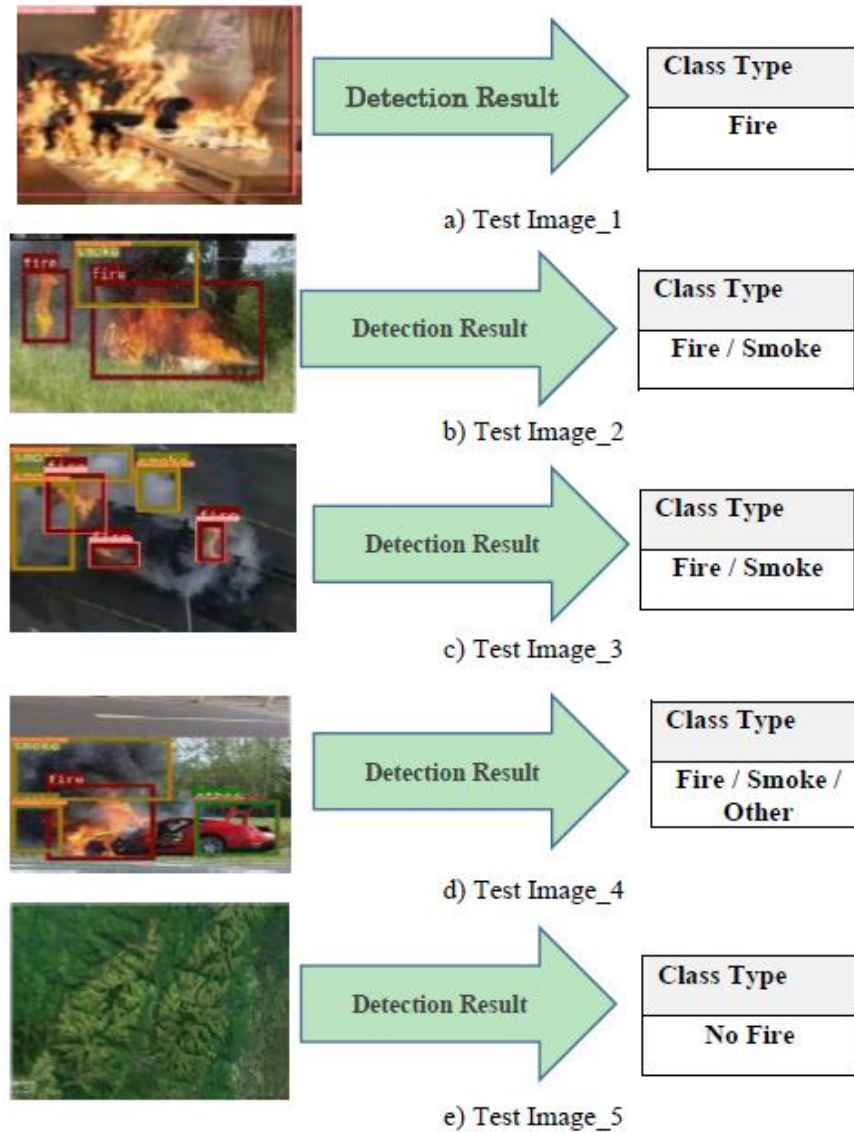


Figure 9. The results of the suggested methodology for the detection phase on a sample of test images

7. PERFORMANCE METRICS

Performance Metrics (PM) score is the weighted average of the recall and accuracy percentages. Consequently, this score considers both false negatives and false positives. Accuracy is sometimes difficult to understand, even though PM is further widespread than precision. When the costs of false positives and false negatives are equal, accuracy performs well.

The accuracy, recall, and F1-score standard metrics were used to assess the proposed system. Furthermore, the dataset's size was taken into account for comparison, while earlier

techniques took either smoke or fire into consideration. We also made a comparison between our system and the most advanced fire detection systems available today, which include a deep learning-based system, an enhanced YOLOv6-based approach [25], a real-time video fire/smoke detection system based on YOLOv2 [26], an enhanced YOLOv5-based smoke detection model [27], a deep learning-based approach for detection for fire detection in smart city environments [28], and a YOLOv3-based approach [29].

It is preferable to consider recall in addition to accuracy if the costs of false positives and false negatives are different.

Precision is defined as the proportion of correctly anticipated observations to all predicted positive outcomes. The complete performance outcomes of the suggested model can be shown in Figure 10. While the smoke and fire and other detection results using the proposed approach including precision, recall and F1-score are also shown in Figure 11.

The systems' performance comparison on the test dataset is displayed in Table 4 and Figure 12. The efficacy of the suggested system in identifying fires in real-world situations was demonstrated by its superior performance over the current methods in terms of precision, recall, and F1-score. The dataset consists of 5426 photos in the collection. Furthermore, on the other hand, our suggested model has an elevated rate accuracy of 95.9% throughout every class with 98.4% of recall and F1-score was 0.971% for detecting fire, smoke, and other things. We credit YOLO v10's enhanced speed and accuracy over earlier iterations of YOLO and additional deep learning-oriented systems of fire detection for our system's better performance. The Comparison of fire detecting systems' performances values can be listed in Table 4 and the histogram of these values can be shown in Figure 12.

An error occurred in calculating the F1-score of the

proposed model due to calculation mistakes. A range of smoke, human outlines and fire are included in this dataset; these include small and large flames, outdoor fire and indoor fire, high and nadir light levels, and typical fire-free environments. The number of images is 5,426. All the images contain fires, but some of the images have smoke with the fire, and some have fire, smoke, and other things such as cars near the fire or smoke. Data augmentation is a two-edged sword in ML since it can add errors for classification and skew sample distributions. Although expansion techniques are intended to improve generalization, their overuse or improper use might undermine the presumption that enlarged data maintains its original classification, leading to distortion. However, because the growth was not enormous, there was no major distortion or addressing error in our suggested system.

Although precise times for "SURF+YOLOv10" are less common than "YOLOv10 only," the outcomes of combining these approaches in a minor increase in computational effort.

The proposed system focused on flame reflections and smoke detection under varying lighting and on nighttime fire detection, and the detection accuracy was good in dealing with highly complex conditions.

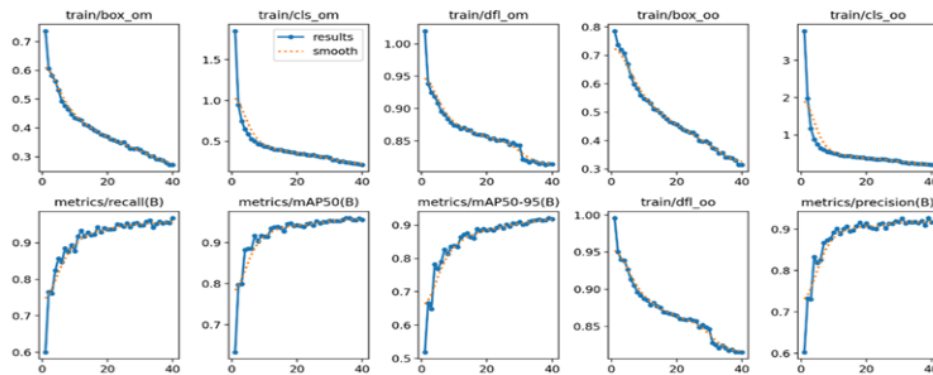


Figure 10. The complete performance measurements outcomes of the suggested model

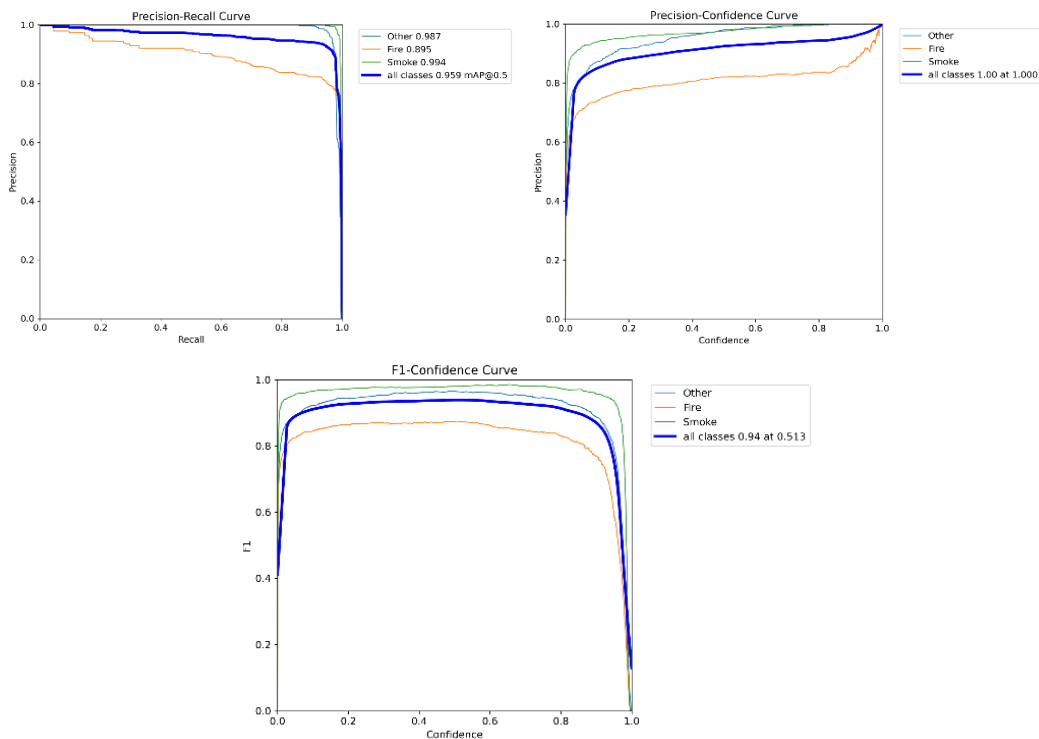


Figure 11. The proposed model's F1-score, precision, and recall model

Table 4. The performance of the systems for the detection of fire is compared

System	Model	Precision %	Recall %	F1-Score	Dataset Size	Fire/Smoke
Norkobil Saydirasulovich [25]	YOLO_v6	93.48	28.29	0.9	4000	Fire/smoke
Mohammed et al. [26]	YOLO_v2	97	97	95.4	4000	Fire/smoke
Wang et al. [27]	YOLO_v5	94.99	78.28	0.858	20.000	Smoke
Talaat and ZainEldin[28]	YOLO_v3	98.1	99.2	0.995	9200	Fire
Avazov [4]	YOLO_v4	98.2	99.7	0.997	9200	Fire
Abdusalomov [29]	YOLO_v8	97.5	95.7	0.962	26.520	Fire/smoke
Proposed Model	YOLO_v10	95.9	98.4	0.971	5426	Fire/smoke/other

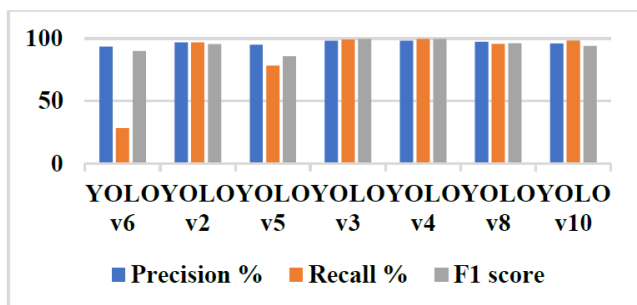


Figure 12. Comparison of fire, smoke and other detecting systems' performances

8. ADVANTAGES AND LIMITATIONS OF THE PROPOSED STUDY

The following are some benefits of utilizing YOLOv10:

1. Superior Performance Balance: Outperforms rival models on all hardware platforms by striking an outstanding balance between speed and accuracy.
2. Unmatched Versatility: The development process for complex applications is made simpler by the fact that a single model family can handle five primary visual AI tasks.
3. A well-managed ecosystem is one that is bolstered by robust growth, a sizable community, regular updates, and extensive resources that guarantee dependability and assistance.
4. Ease of Use: Made to offer a simplified user experience, this feature makes it possible for both beginners and experts to train and implement models with little difficulty.
5. Efficiency of Deployment and Training: It can be utilized with a range of devices, such as cloud servers and edge devices, and is made for lower memory usage and faster training times.

Compared to its predecessors, such as YOLOv8 and YOLOv9, YOLOv10 generally improves computer performance by offering noticeably quicker inference speeds (lower latency) and utilizing less memory (GPU VRAM/RAM). It is very efficient for real-time applications on both GPUs and edge devices thanks to architectural innovations such eliminating the Non-Maximum Suppression (NMS) post-processing step and using efficiency-driven model designs.

The main challenges of YOLOv10 are:

1. When a fire's color, intensity, or texture closely resembles the surrounding environment, the YOLOv10 model may have trouble identifying it (e.g., in low light, foggy conditions, or amid similar colored foliage).
2. Environmental conditions like humidity and, to a lesser extent, light intensity have a major impact on smoke detection. Intense ambient light can affect certain types of sensors, while

high humidity, steam, and condensation are major causes of nuisance alerts.

To overcome these challenges, potential improvements to reduce challenges via employing sophisticated methods (like "CBAM") and bespoke data augmentation (such jittering of color) to enable the model better focus on the fire rather than the background.

The following are the main shortcomings of the YOLOv10:

1. A complicated setup for integrating multiple sources.
2. Limited Versatility: YOLOv10 is just designed for object detection, in contrast to YOLOv11, which includes integrated multitasking capabilities for segmentation, posture estimation, and classification.
3. Support and Ecosystem: It lacks the same level of community support, integrated tools, and ongoing maintenance as it is an academic institution's research-driven paradigm.
4. Usability: Including YOLOv10 in a production process may need more human work.

9. CONCLUSIONS

Fires of intelligent city can have disastrous results, putting residents' lives in jeopardy and causing property damage. The accuracy and speed limitations of conventional fire detection methods make real-time fire detection challenging. This study presents the ISFD system, a highly advanced intelligent fire and smoke detection system built on the YOLOv10 algorithm for real-time fire detection and employing deep learning capabilities. When compared to traditional fire detection systems, the ISFD methodology can reduce false alarm rates; increase the detection accuracy of smoke, fire, and other, and save money. It can help with early fire detection and containment, preserving life and minimizing environmental and property harm. As indicated in Table 4, the ISFD methodology can save money, decrease false alarms, and improve fire detection accuracy when contrast to conventional systems for detection of fire. Other interesting features of intelligent cities, such as leaks of gas or floods, can be discovered by extending it. To identify and locate fires in real time, the suggested method makes use of a deep neural network that has been trained on a sizable collection of fire photos. Our proposed model has an excellent rate of accuracy 95.9% for every class with 98.4% recall and an F1-score of 0.971% for detecting fire, smoke, and other things.

10. FUTURE RESEARCH SUGGESTIONS

The suggestions for future works can be illustrated as follows:

1. I recommend using the SURF or "Oudemansiella

raphanopies phenotype extractor" (s) or CNN methods to elicited points of interests of video frames.

2. Long Short Term Memory (LSTM) and Recurrent Neural Networks (RNNs) can be utilized to analyze information and events within video footage.
3. In real-time environment, YOLO V11 can be utilized to elicit fire and smoke.
4. The suggested model can also be expanded to identify other fascinating features of smart cities, including flooding or leaks of gas.

ACKNOWLEDGMENTS

This article is supported by University of Technology/College of Computer Science, Baghdad/ Iraq.

REFERENCES

- [1] Abdulhadi, H.M., Aldeen, Y.A.A.S., Yousif, M.A., Madni, S.H.H. (2023). Enhancing smart cities with IoT and cloud computing: A study on integrating wireless ad hoc networks for efficient communication. *Baghdad Science Journal*, 20(6): 49. <https://doi.org/10.21123/bsj.2023.9277>
- [2] El-Hosseini, M., ZainEldin, H., Arafat, H., Badawy, M. (2021). A fire detection model based on power-aware scheduling for IoT-sensors in smart cities with partial coverage. *Journal of Ambient Intelligence and Humanized Computing*, 12(2): 2629-2648. <https://doi.org/10.1007/s12652-020-02425-w>
- [3] Zhang, Z., Wang, L., Liu, S., Yin, Y. (2024). Intelligent fire location detection approach for extrawide immersed tunnels. *Expert Systems with Applications*, 239: 122251. <https://doi.org/10.1016/j.eswa.2023.122251>
- [4] Avazov, K., Mukhiddinov, M., Makhmudov, F., Cho, Y.I. (2021). Fire detection method in smart city environments using a deep-learning-based approach. *Electronics*, 11(1): 73. <https://doi.org/10.3390/electronics11010073>
- [5] Zhan, J., Hu, Y., Zhou, G., Wang, Y., Cai, W., Li, L. (2022). A high-precision forest fire smoke detection approach based on ARGNet. *Computers and Electronics in Agriculture*, 196: 106874. <https://doi.org/10.1016/j.compag.2022.106874>
- [6] Peng, R., Cui, C., Wu, Y. (2025). Real-time fire detection algorithm on low-power endpoint device. *Journal of Real-Time Image Processing*, 22(1): 29. <https://doi.org/10.1007/s11554-024-01605-7>
- [7] Sierra, D., Montanaro, W., Kuo, L., Zohuri, B. (2023). Enhancing fire detection through CNN and transfer learning: A comprehensive research study. *Journal of Engineering and Applied Sciences Technology*, 174(5): 2-6. [https://doi.org/10.47363/JEAST/2023\(5\)174](https://doi.org/10.47363/JEAST/2023(5)174)
- [8] Yusro, M.M., Ali, R., Hitam, M.S. (2023). Comparison of faster R-CNN and YOLOv5 for overlapping objects recognition. *Baghdad Science Journal*, 20(3): 15. <https://doi.org/10.21123/bsj.2022.7243>
- [9] Neamah, S.B., Karim, A.A. (2023). Real-time traffic monitoring system based on deep learning and YOLOv8. *Aro-The Scientific Journal of Koya University*, 11(2): 137-150. <https://doi.org/10.14500/aro.11327>
- [10] Zhang, Z., Tan, L., Tiong, R.L.K. (2024). Ship-Fire Net: An improved YOLOv8 algorithm for ship fire detection. *Sensors*, 24(3): 727. <https://doi.org/10.3390/s24030727>
- [11] Talib, M., Al-Noori, A.H., Suad, J. (2024). YOLOv8-CAB: Improved YOLOv8 for Real-time object detection. *Karbala International Journal of Modern Science*, 10(1): 5. <https://doi.org/10.33640/2405-609X.3339>
- [12] Saeid, A.A., Ogla, R., Shaker, S.H. (2025). A novel approach for shape pattern recognition based on boundary features generated by line simplification algorithm. *Baghdad Science Journal*, 22(1): 349-360. <https://doi.org/10.21123/bsj.2024.9517>
- [13] Ilina, O.V., Tereshonok, M.V. (2022). Robustness study of a deep convolutional neural network for vehicle detection in aerial imagery. *Journal of Communications Technology and Electronics*, 67(2): 164-170. <https://doi.org/10.1134/S1064226922020048>
- [14] Gao, P. (2024). A fire and smoke detection model based on YOLOv8 improvement. *International Journal of Advanced Computer Science & Applications*, 15(3): 179-190. <https://doi.org/10.14569/IJACSA.2024.0150318>
- [15] Biswas, A., Ghosh, S.K., Ghosh, A. (2023). Early fire detection and alert system using modified inception-v3 under deep learning framework. *Procedia Computer Science*, 218: 2243-2252. <https://doi.org/10.1016/j.procs.2023.01.200>
- [16] Najeeb, H.D., Ghani, R.F. (2021). Proposed method for scale drawing calculating depending on the line detector and length detector. *Iraqi Journal for Computer Science and Mathematics*, 2(2): 2. <https://doi.org/10.52866/ijcsm.2021.02.02.002>
- [17] Zhang, D. (2024). A yolo-based approach for fire and smoke detection in IoT surveillance systems. *International Journal of Advanced Computer Science & Applications*, 15(1): 87. <https://doi.org/10.14569/ijacsa.2024.0150109>
- [18] Najjar, F.H., AbdulAmeer, A.A., Kadum, S. (2025). Hybrid SVD and SURF-based framework for robust image forgery detection and object localization. *Journal of Robotics and Control (JRC)*, 6(2): 535-542. <https://doi.org/10.18196/jrc.v6i2.25567>
- [19] Ahmed, S., Akbas, A., Naser, E. (2023). Tesseract OpenCV versus CNN: A comparative study on the recognition of unified modern Iraqi license plates. *Revue d'Intelligence Artificielle*, 37(5): 1331-1339. <https://doi.org/10.18280/ria.370526>
- [20] Abdulabass, D.F., Abdulmunim, M.E. (2024). Traffic sign detection using you only look once (YOLOv3) technique. *Iraqi Journal of Science*, 65(10): 5741-5753. <https://doi.org/10.24996/ijcs.2024.65.10.34>
- [21] Liao, L., Song, C., Wu, S., Fu, J. (2025). A novel YOLOv10-based algorithm for accurate steel surface defect detection. *Sensors*, 25(3): 769. <https://doi.org/10.3390/s25030769>
- [22] Naser, E.F., Khudair, E.T., Mahmood, E.S., Malood, A.T. (2024). A comparison between backpropagation neural network and seven moments for more accurate fingerprint video frames recognition. *Baghdad Science Journal*, 21(11): 5. <https://doi.org/10.21123/bsj.2024.8777>
- [23] Hinojosa Lee, M.C., Braet, J., Springael, J. (2024). Performance metrics for multilabel emotion classification: Comparing micro, macro, and weighted F1-scores. *Applied Sciences*, 14(21): 9863.

- <https://doi.org/10.3390/app14219863>
- [24] UCF101 Videos.
<https://www.kaggle.com/datasets/pevogam/ucf101>.
- [25] Norkobil Saydirasulovich, S., Abdusalomov, A., Jamil, M.K., Nasimov, R., Kozhamzharova, D., Cho, Y.I. (2023). A YOLOv6-based improved fire detection approach for smart city environments. *Sensors*, 23(6): 3161. <https://doi.org/10.3390/s23063161>
- [26] Mohammed, M.S., Abbas, A.H., Abdullah, N.A. (2024). Intelligent surveillance systems for fire detection in open areas: A survey. *Iraqi Journal of Science*, 65(5): 2813-2827. <https://doi.org/10.24996/ijs.2024.65.5.36>
- [27] Wang, Z., Wu, L., Li, T., Shi, P. (2022). A smoke detection model based on improved YOLOv5. *Mathematics*, 10(7): 1190. <https://doi.org/10.3390/math10071190>
- [28] Talaat, F.M., ZainEldin, H. (2023). An improved fire detection approach based on YOLO-v8 for smart cities. *Neural Computing and Applications*, 35(28): 20939-20954. <https://doi.org/10.1007/s00521-023-08809-1>
- [29] Abdusalomov, A., Baratov, N., Kutlimuratov, A., Whangbo, T.K. (2021). An improvement of the fire detection and classification method using YOLOv3 for surveillance systems. *Sensors*, 21(19): 6519. <https://doi.org/10.3390/s21196519>