fire

MDPI

Article

Spatio-Temporal Data Model for Early Wildfire Detection

Damir Krstinić [1,2,*], Jakov Bejo [1,2], Toma Sikora [1] and Marin Bugarić [1,2]

[1] Department Electronics and Computing, Faculty of Electrical Engineering Mechanical Engineering and Naval Architecture (FESB), University of Split, Ruđera Boškovića 32, 21000 Split, Croatia; jakov.bejo.00@fesb.hr (J.B.); toma.sikora@fesb.hr (T.S.); mbugaric@fesb.hr (M.B.)
[2] Code Fire d.o.o., Ruđera Boškovića 32, 21000 Split, Croatia
* Correspondence: damir.krstinic@fesb.hr

Abstract

Early detection is a key tool for mitigating the devastating effects of wildfires. Single-frame detection methods that do not consider inter-frame dependencies often fail to detect smoke plumes at the earliest stage and at greater distances, or produce excessive false alarms. Biological vision is particularly sensitive to motion cues, and this translates well to automated systems. Recent temporal-memory approaches have demonstrated improved performance over purely spatial methods, but typically rely on complex, computationally heavy multi-stage architectures. This study investigates the possibility of encoding temporal and contextual information into additional image channels as a basis for compiling data models with increased information content. Seven distinct data models were proposed, and corresponding datasets were generated to train standard YOLO architectures without modifications to the network structure. The datasets were compiled from real wildfire footage collected from an operational wildfire surveillance system in Croatia, comprising 333 annotated sequences of real fires recorded between 2018 and 2024. Experimental evaluation compared the performance of YOLO models trained on the information-enriched datasets with those trained on standard RGB images. Based on the results, the best data model for early wildfire smoke detection, combining original RGB channels with short-term and long-term temporal memory, was selected. Comparative evaluation demonstrated improved detection accuracy, achieving up to 5 percent higher true-positive detection rate for models trained on spatio-temporal data compared to standard RGB images, while maintaining low inference latency. The proposed approach shifts the focus to the structure and information content of the data while preserving the efficiency of standard convolutional neural network architectures. This approach could be applied to other problems requiring high efficiency and real-time operation, where temporal and contextual information can improve detection performance.

Keywords: data models; smoke detection; wildfire surveillance; YOLO; machine learning

Check for updates

Academic Editor: Tao Chen

Received: 19 February 2026
Revised: 19 March 2026
Accepted: 16 April 2026
Published: 21 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Wildfires are among the most destructive natural phenomena, significantly affecting human lives and entire ecosystems. The consequences of large wildfires are not only short-term and often catastrophic but also cause lasting changes to communities and leave impacts on the ecosystem that can be visible for decades. Early and accurate detection is therefore paramount in mitigating the devastating effects of wildfires, enabling a more rapid response and containment before they escalate into uncontrollable conflagrations [1]. Technological advances and the development of advanced image-processing algorithms

have enabled a shift from systems relying on trained human observers to technological solutions based on surveillance cameras and automatic detection algorithms [2–4]. Although these systems have improved wildfire monitoring, the complex nature of smoke, which is the first visible sign of wildfire during daytime, often results in many false alarms as well as missed detections. The semi-transparent nature of smoke, the lack of clearly defined features, image-quality degradation due to atmospheric and other disturbances, and the need to detect fires at long distances and at the earliest stages render early wildfire smoke detection particularly challenging. This challenge is compounded by the varying shapes, sizes, and colors of smoke plumes, alongside potential sensor degradation, which collectively hinder effective spatial information extraction from a single image. The rapid development of deep learning and convolutional neural networks has greatly enhanced image analysis and computer vision capabilities. These algorithms have found applications across many domains, including early smoke detection for wildfires [5,6]. Specifically, You Only Look Once (YOLO) architectures [7] have emerged as particularly promising due to their real-time performance and high accuracy in object-detection tasks, even in complex environmental conditions [8]. However, the inherent limitations of processing static RGB images for dynamic phenomena like smoke propagation still present significant hurdles for these models, often leading to reduced precision and increased false positive rates in real-world scenarios [9].

The pronounced sensitivity of human vision to motion and change has inspired approaches that combine temporal information from sequences of consecutive frames with the spatial information contained in a single image, aiming to improve detection accuracy. These methods often rely on hybrid multi-stage algorithms that combine different neural network architectures to extract spatial and temporal information [10–12]. While effective, such architectures are computationally intensive, which can be a limiting factor for large-scale deployment where the system must simultaneously process multiple high-resolution video streams. This necessitates alternative strategies that integrate temporal cues with spatial features without significantly increasing model complexity or inference latency [13–15].

To circumvent these limitations while preserving the benefits of temporal information cues, this study investigates the possibility of increasing the information content of the input data by encoding temporal and contextual information into additional image channels, rather than customizing neural network architectures for video stream processing. This work builds upon our prior research that explores the integration of diverse data models for enhanced smoke detection [16]. The proposed strategy reduces computational overhead by directly integrating spatio-temporal data into a unified input format for streamlined processing by existing, highly optimized object detection models.

This study is based on a dataset collected from the archive of intelligent wildfire surveillance system currently operational at 116 locations in the Republic of Croatia, as well as at additional sites in Bosnia and Herzegovina, Montenegro, and Albania [17]. The surveillance locations in Croatia are shown in Figure 1. This archive contains recordings collected under various atmospheric and visibility conditions, with image-quality degradation that occurs during use of surveillance cameras at remote and isolated locations throughout the year. The archive contains footage of real forest fires in the earliest stages and at different distances from the camera.

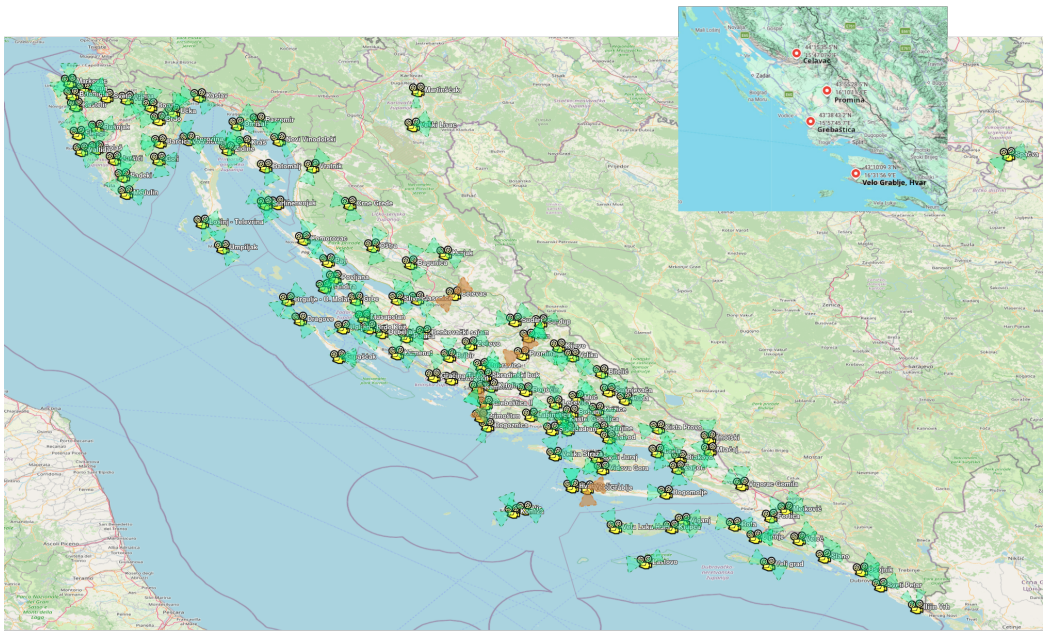
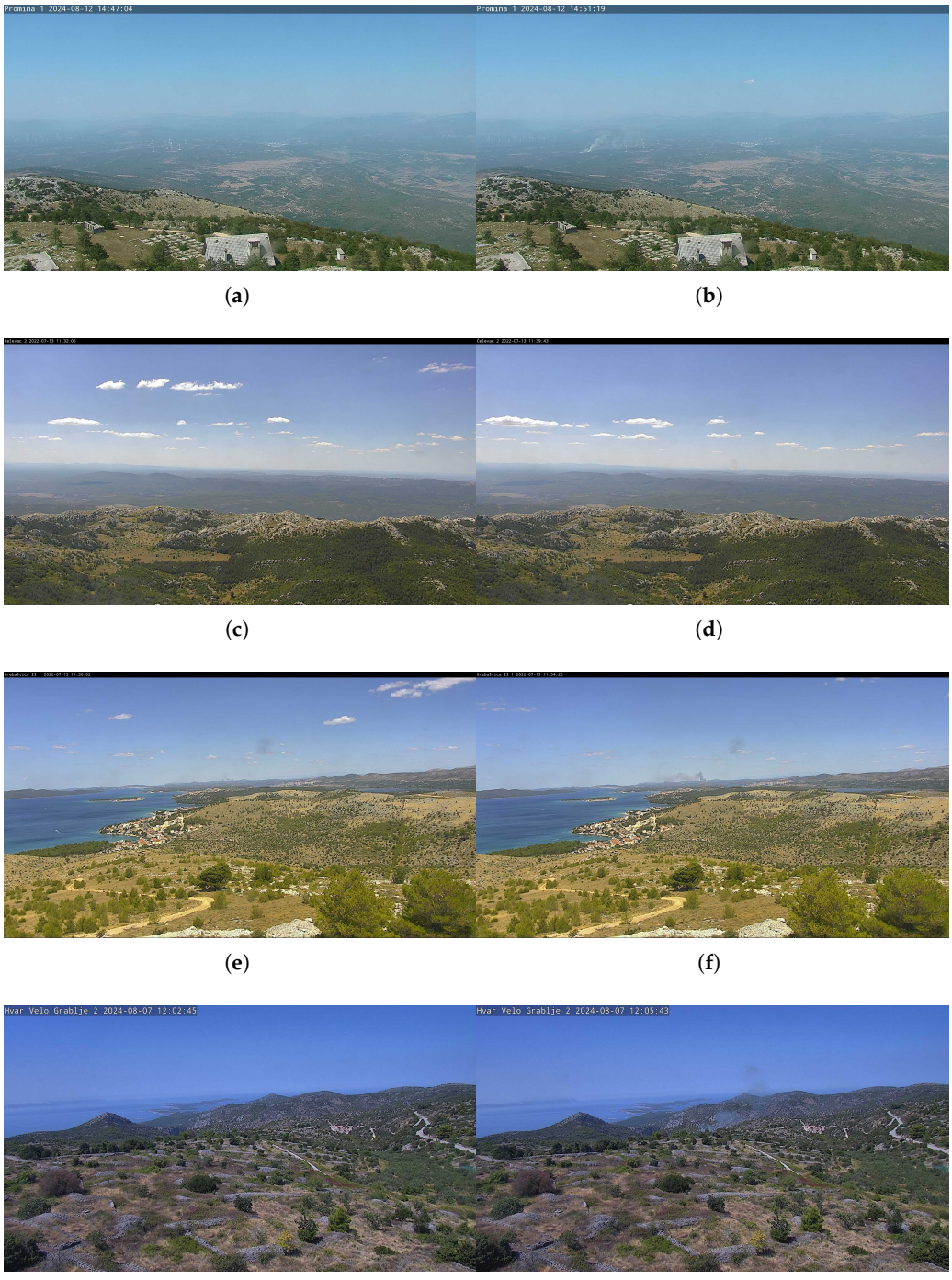


Figure 1. Monitoring locations of the intelligent wildfire surveillance system in Croatia. Red markers indicate the camera locations from which the wildfire example images shown in Figure 2 were collected. The inset shows exact geographic positions of these locations.

Sequences of images with duration of 30 to 45 min were retrieved from the system archive. By processing consecutive frames, additional image channels encoding temporal and contextual information were generated. These supplementary channels were combined with the existing RGB channels to form multidimensional samples that capture spatial, temporal, and contextual information. By combining these additional features extracted from sequences of consecutive frames, seven multidimensional data models with different combinations of spatial, temporal, and contextual information were compiled, including a regular RGB samples as a baseline for comparative evaluation, and a separate dataset was generated for each data model. These multichannel datasets were used to train a standard YOLO architecture, aiming to select the best data model for wildfire detection. The obtained results were compared with those achieved by training the same YOLO architecture in the same set of images using only RGB channels (standard image).

The evaluation methodology follows a three-stage evaluation process: (a) different models of multichannel data were evaluated on a limited-size data in order to select the best data model for wildfire smoke detection; (b) the best-performing data model was retrained and evaluated on large dataset obtained using augmentation techniques; (c) final validation was performed on the original high-resolution image sequences collected from a separate set of monitoring locations that were not used in earlier phases of the study. In the last evaluation step, a production wildfire surveillance environment was simulated to assess both the effectiveness of the proposed solution and its efficiency with respect to potential real-time deployment.

The main contributions of this work are: (a) a method for encoding temporal and contextual information into additional image channels, enabling the use of standard and efficient DNN architectures for processing video sequences; (b) selection of the optimal data model for wildfire smoke detection; (c) a real-time smoke detection algorithm applicable to high-resolution video sequences; (d) a sequence-aware augmentation technique that preserves the temporal relationships between consecutive frames; (e) a wildfire smoke dataset containing original RGB images and additional spatio-temporal channels, made available for future research.



(**a**) (**b**)

(**c**) (**d**)

(**e**) (**f**)

(**g**) (**h**)

Figure 2. Examples of wildfire images from the dataset. Images on the left (**a**,**c**,**e**,**g**) contain first visible signs of smoke that should be automatically detected. In some cases, smoke is barely visible. Right column (**b**,**d**,**f**,**h**): Developed wildfire 3 to 7 min since the outbreak. The geographic locations of the corresponding camera sites are indicated by red markers in Figure 1. (**a**) Location Promina; (**b**) Developed wildfire after 4 min, endangering wind farm; (**c**) Location Ćelavac, simultaneous occurrence of two wildfires; (**d**) Developed wildfires after 7 min; (**e**) Location Grebaštica; (**f**) developed wildfire after 3 min, endangering a densely populated area; (**g**) Location Velo Grablje, Hvar; (**h**) emerging wildfire after 3 min, endangering village.

The remainder of the paper is structured as follows. Section 2 provides an overview of existing research on smoke detection for early wildfire detection and reviews deep learning methods that incorporate temporal information, with an emphasis on the YOLO algorithm.

Section 3 details the methodology of the presented study, including the generation of multi-channel data models and the evaluation procedure of the proposed approach. Section 4 presents the experimental results, followed by the discussion in Section 5. The conclusions are presented in Section 6.

2. Related Work

The evolution of wildfire detection systems, from manual human observers to automated systems, has been primarily driven by computational capability and algorithmic complexity. While early automation approaches provided reasonable results, they required substantial post-processing and fine-tuning. More recently, rapid advances in GPU compute power-enabled deep neural networks (DNNs), particularly convolutional neural networks (CNNs), to become the dominant paradigm in computer vision-based object detection [7].

Current industry standard for CNN-based real-time detection and segmentation tasks is the YOLO family of models, first introduced in 2016 [7]. By replacing the traditional multi-stage localization and detection approach of regular CNNs with single-pass inference (with convolutional layers predicting both bounding boxes and class probabilities) YOLO achieved the balance between accuracy and computational efficiency necessary for embedded hardware deployment. Each new version added architectural improvements and task-specific optimizations [18–20]. Due to their excellent real-time performance, YOLO models were rapidly adopted for wildfire detection [21–24].

Considering the practical constraints of deploying automated systems in remote locations, recent research focuses on lightweight CNN architectures capable of real-time smoke detection on resource-limited hardware. El-Madafri et al. [25] propose a compact convolutional model optimized for lookout towers with CPU limited inference capabilities. Through hierarchical knowledge distillation across multiple models, they reduce complexity while preserving accuracy. Their system processes single RGB frames from fixed cameras and achieves near real-time performance with competitive detection results, demonstrating that carefully designed lightweight CNNs can be deployed on edge devices for operational wildfire monitoring. Similarly, Almeida et al. [26] propose EdgeFireSmoke, a lightweight CNN model capable of fire and smoke detection on images from a video stream in real time. The architecture targets edge computing devices on unmanned aerial vehicles and video surveillance systems.

More recently, attention mechanisms have become popular with detection architectures to improve contextual understanding [27]. The transformer's superior context modeling capabilities improve discrimination between actual wildfire and benign events, yielding better results than CNN-only approaches. Unfortunately, the introduction of the transformer to the backbone comes at the cost of reduced inference speed [28]. Transformer architectures have been most successful in the domain of video action classification [29–31]. The task entails the network consuming a sequence of a significant amount of frames with classification clues spread across the sequence. The projects show large transformer based networks, such as VTN, are capable of gathering the information through the attention mechanism and correctly classify actions. However, as with all attention-based architectures, computational complexity is multiple orders of magnitude higher than conventional CNNs, between 1000 and 8000 GFLOPs for a VTN, compared to approximately 165 GFLOPs for YOLOv8l.

This trade-off between performance and speed also comes to light with the latest YOLOv12 model architecture. While attention mechanics help YOLOv12 achieve better results compared to older models [20], specialized task-specific modifications often prove more effective for smoke detection, as demonstrated in recent studies [32–34]. For example, in F3-YOLO presented in [33], improvements to feature fusion in deeper CNN layers can

surpass attention-based models like YOLOv12 while maintaining computational efficiency of pure CNN architectures.

Despite these architectural advances, with respect to early smoke detection systems, a fundamental limitation remains: they rely on individual frames without incorporating temporal motion data. Models trained on static RGB or thermal images inherently ignore the characteristic motion of smoke plumes. Their shape and speed varies, but at longer distances where spatial resolution degrades, the temporal motion of smoke becomes crucial discriminative information.

In line with this reasoning, recent studies increasingly use temporal information to boost wildfire smoke-detection performance. The SmokeyNet model featured in [12] combines both a CNN backbone to extract the regions of interest and a vision transformer to classify it. In order to provide temporal data for better classification, the model also includes a Long Short-Term Memory (LSTM) [35,36] module to leverage the motion between the current and previous frame. The SmokeyNet model is trained on image sequences that start 40 min before and end 40 min after the start of the wildfire, with each frame 60 s apart. However, its multi-stage architecture involving CNN, LSTM, and a vision transformer component presents computational challenges for deployment in systems with a large number of monitoring locations and cameras that require processing multiple high-resolution images per second [17].

These constraints motivate our novel approach, first proposed in the preceding work [16]. Instead of introducing architectural complexity through sequential processing modules like LSTM, we embed temporal information directly into the input representation through additional image channels. This channel-based encoding of temporal data, combined with contextual channels that capture distance and background information, allows standard YOLO architectures to leverage temporal and spatial contexts without the computational overhead of recurrent and transformer elements. By training the YOLO model on this custom multichannel data model, we expect a large increase in detection accuracy when compared to conventional RGB approaches, while maintaining the inference performance necessary for real-time deployment.

While our preliminary study [16] demonstrated the feasibility of multichannel spatio-temporal encoding for wildfire smoke detection, it relied on a modified YOLOv5 architecture, a limited dataset, and patch-level evaluation. The present work substantially extends this foundation by systematically evaluating seven distinct data models across four YOLOv8 model sizes, significantly expanding the dataset across geographically diverse locations, and introducing a sequence-aware augmentation strategy. Furthermore, the evaluation is conducted on full-resolution sequences under conditions that emulate real operational deployment, including rigorous assessment of detection delay, inference latency, and model complexity—metrics critical for determining suitability for real-time wildfire surveillance.

3. Materials and Methods

This section presents the methodology used to develop different multichannel image data models and evaluate their suitability for early wildfire detection. The presented research can be divided into the following steps:

1. Collecting data from the archive of the operational wildfire surveillance system;
2. Compilation of different multichannel image data models based on various spatial, temporal, and contextual information;
3. Training and evaluation of the standard YOLO architecture on the collected datasets in order to select the best-performing multichannel image data model;

4. Generating an extended dataset using augmentation techniques for the selected multichannel data model, as well as training and evaluation of the standard YOLO architecture on the extended dataset;
5. Evaluation of the trained algorithm on original high-resolution video sequences taken from the cameras of the wildfire surveillance system.

3.1. Dataset Collection

The dataset used in this study was collected from monitoring cameras that are part of an operational wildfire surveillance system implemented in the Republic of Croatia [17]. The dataset was collected with the consent of the company Odašiljači i veze d.o.o., Ulica grada Vukovara 269d, HR-10000 Zagreb, Croatia, which is the operator and owner of the system. This system is currently installed on 116 locations. Most locations are equipped with two Pan–Tilt–Zoom (PTZ) cameras, which are actively used for real-time wildfire detection throughout the year. Frames are retrieved from the camera at a resolution of 1920×1080 and are stored in the archive without any modifications. The dataset was collected from the archive of the system, providing valuable real-world footage of wildfire incidents enhancing the relevance and applicability of this study. The data used in this study were retrieved from the system archive for the period 2018–2024, covering a total of 79 monitoring locations that were active during this period.

In the operational system, video feeds from all surveillance locations are routed to seven regional monitoring centers. Each center is staffed by trained operators who monitor a video wall displaying live feeds from all cameras in the assigned region, with images cycling through different preset positions. When the automated detection algorithm generates an alarm for a potential wildfire, the alarm is visually indicated on the corresponding image on the video wall. The operator is responsible for alarm verification: upon receiving an alert, the operator can stop the automatic camera rotation, switch to manual PTZ control, and zoom in (up to $32\times$ or $40\times$, depending on the camera model) on the suspicious area to perform a detailed visual inspection. The operator either confirms the alarm—triggering the emergency response protocol, which includes notifying the relevant fire suppression services—or dismisses it as a false positive.

This stratification ensures that models are evaluated on entirely unseen environments: the validation and test data are collected from geographically separate locations, providing a realistic assessment of the model's ability to generalize across varying terrains and atmospheric conditions.

Each camera at a given location follows a systematic surveillance protocol involving eight preset positions, which are visited sequentially. At each preset position the camera stops for approximately 15 s. Image capture begins after 5 s, allowing the camera to focus. Over the next 10 s, images are captured at one-second intervals. Due to connection quality variations, the number of frames at a single stop at a preset position may vary from 8 to 11. Captured frames are analyzed to detect smoke, which represents the first visible sign of wildfire. After capturing frames at one preset position, the camera moves to the next. After completing all eight preset positions, the camera returns to the first position, with the entire cycle taking approximately two minutes. This process repeats continuously.

The dataset was collected in the form of sequences, i.e., a set of consecutive image frames taken from one preset position over a period of approximately 30 to 45 min. When extracting sequences, the first visible signs of smoke in the sequence were sought based on information about actual fires or reported agricultural burnings. Each sequence was constructed by taking up to 30 min that precede the appearance of the first visible signs of smoke and approximately 10 min after the smoke appeared, i.e., containing developing smoke. Some sequences containing agricultural burnings exceeded 45 min. During such

activities, smoke often appears simultaneously or within short time intervals at several locations within the camera field of view. Also, smoke can disappear for short or extended periods, before reappearing near its previous location. In these cases, longer sequences were extracted to capture as many independent smoke events as possible, ensuring at least 10 min elapsed before the first smoke appearance in a sequence. Consequently, each sequence is associated with a particular preset position and consists of multiple series of consecutive frames (8 to 11 frames per series) captured at one-second intervals, followed by a temporal gap of approximately two minutes between two consecutive series. Smoke typically appears after approximately 10 series (i.e., about 20 to 30 min from the start of the sequence) and develops over the following several series. In most cases, extracted sequences were limited to the first ten minutes from the initial signs of smoke. This limitation is based on the premise that ten minutes represent the critical early phase of wildfire, where timely detection can significantly influence response and containment efforts. Everything recorded after this ten-minute window no longer qualifies as early signs of wildfire. By focusing on the initial signs of smoke, the dataset emphasizes the importance of developing detection systems capable of recognizing wildfire indicators at its earliest stage. Exceptionally, for some sequences where the fire outbreak point is at a very large distance from the monitoring location (20 to 40 km) and the smoke is hardly visible, a somewhat longer sampling period was considered. Examples of real wildfire events with smoke that should be automatically detected are shown in Figure 2. It can be seen that in some examples, even trained human observer may struggle to detect smoke.

The majority of the sequences were collected during the high-fire season, during the summer months. Part of the sequences were collected throughout the entire year, particularly during agricultural activities and agricultural burnings. Sequences were collected at different times of day and under varying weather, atmospheric and illumination conditions.

To ensure a balanced dataset a sufficient number of sequences without smoke were also included. These non-smoke sequences were carefully selected to represent a wide range of environmental and illumination conditions, as well as different periods throughout the year. Among the selected non-smoke sequences, some were deliberately chosen to contain visual artifacts such as fog, dust, or other environmental conditions that can resemble smoke in appearance. These challenging scenarios were included to improve the model robustness and its ability to distinguish true wildfire smoke from similar-looking visual phenomena. Incorporating such sequences aims to reduce false positives in real-world conditions, where environmental artifacts often mimic early signs of smoke, thereby enhancing the overall reliability and precision of the wildfire detection system.

The total number of locations and sequences, including those containing smoke and those without, allocated to each dataset subset (training, validation, and testing), is summarized in Table 1.

Table 1. Number of locations and collected sequences for training, validation and test data.

Data	Locations	Sequences	Smoke	No Smoke
Train	48	234	147	87
Validation	18	53	29	24
Test	13	46	26	20

Representative 640 × 640 samples from the training and validation datasets are shown in Figure 3. Selected training samples show samples generated from augmented sequences, with augmentation applied at the sequence level as described in Section 3.4. Validation and test samples are extracted exclusively from original, non-augmented frames. Examples from the test dataset are presented in the context of sequence evaluation results in Figure 4.

The smoke regions were manually annotated on all sequences. Annotations were created by drawing polygons around primary smoke emissions in our own custom software tailored for sequence annotation.

3.2. Data Models

Sequences collected from the wildfire surveillance system archive were used to generate temporal and contextual information to be embedded into the additional image channels. Two different approaches for encoding temporal information were examined. The first approach leverages the image acquisition protocol—a short sequence of consecutive frames followed by a longer pause between series—to estimate short-term and long-term temporal memory. The latter is based on short-time foreground estimation. Additionally, one approach was used to generate contextual information channel.



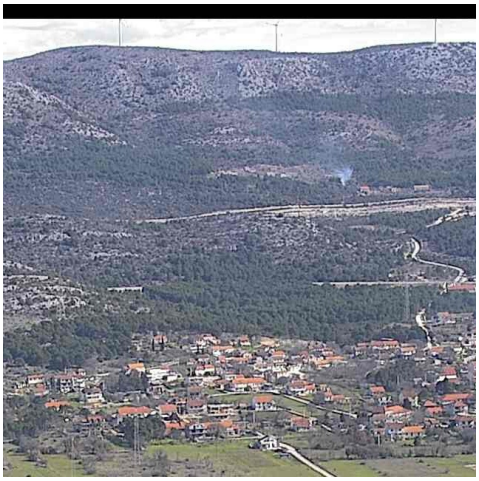
(**a**) Training sample 1 (augmented)



(**b**) Training sample 2 (augmented)

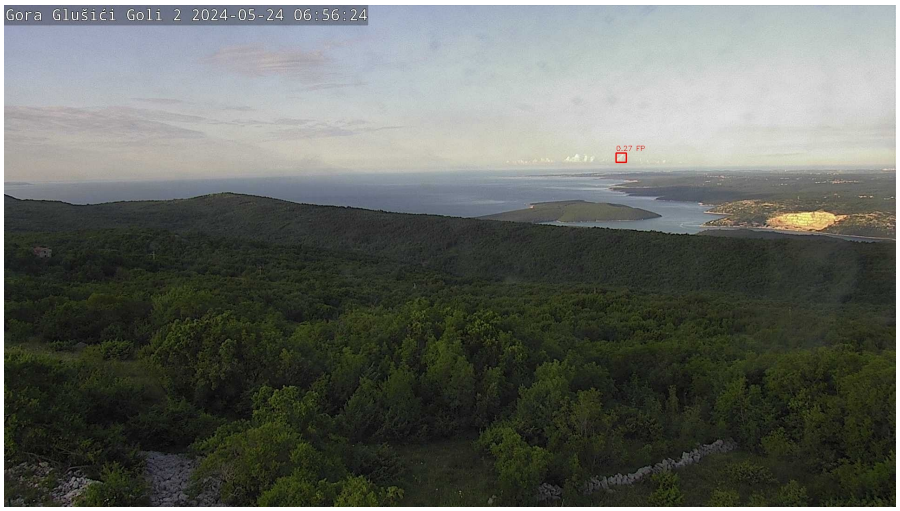


(**c**) Validation sample 1

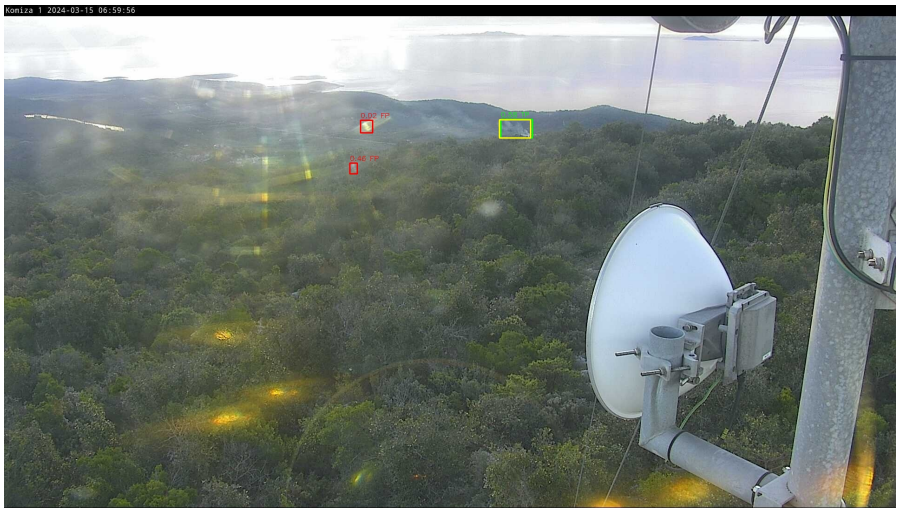


(**d**) Validation sample 2

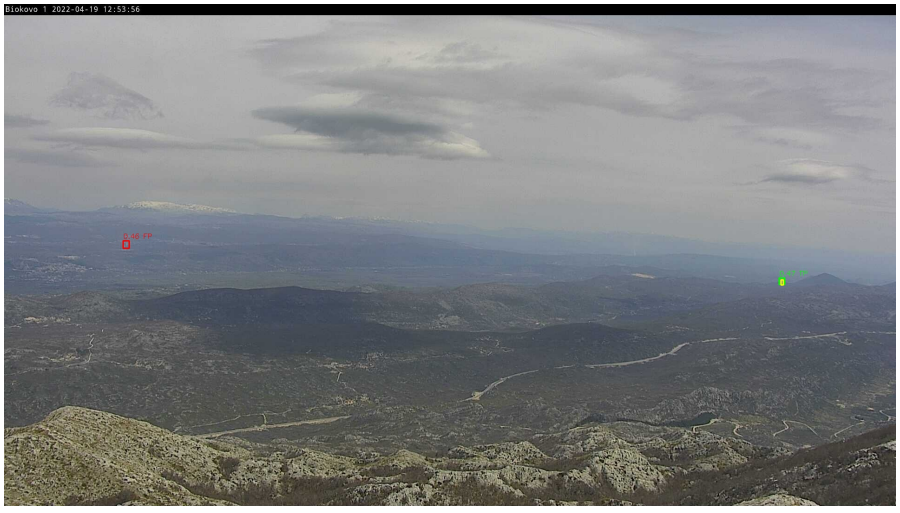
Figure 3. Representative samples from training (**a**,**b**) and validation (**c**,**d**) datasets. Training samples illustrate the effect of the sequence-aware augmentation strategy. Validation and test samples are original, non-augmented frames.



(**a**) Location Gora Golušići Goli, FP detection



(**b**) Location Komiža, TP and FP detection



(**c**) Location Biokovo, TP and FP detection—subsequently identified as real smoke

Figure 4. Examples of false-positive (FP, red bounding boxes) and true-positive (TP, green bounding boxes) detections from the test sequences: (**a**) FP detection of clouds on the horizon; (**b**) simultaneous TP detection and FP detection caused by lens flare and sunlight reflections; (**c**) simultaneous TP and FP detections under overcast conditions with atmospheric haze, where the FP was subsequently identified as actual smoke missed during manual annotation.

3.2.1. Long- and Short-Term Memory Encoding

Only the blue channel was used for calculating of the long- and short-term memory channels, as the scattering of sunlight on smoke molecules in the atmosphere is inversely proportional to the wavelength [37]. This means that the scattering is greatest in the wavelengths of blue light, making the blue component of RGB image the most effective for calculating the temporal image features [16].

Short-time memory is defined as running average of the blue channel, defined as:

$$S_i = (1 - \alpha)S_{i-1} + \alpha I_i^B, \quad (1)$$

where short-time memory S_i is the temporal image at time step i, I_i^B is blue channel of the current frame I_i, and α is set to 0.1. This value was selected based on the image acquisition protocol: with frames captured at one-second intervals, $\alpha = 0.1$ corresponds to an effective temporal horizon of $1/\alpha = 10$ time steps, which matches the duration of a single camera stop at a preset position. This ensures that short-term memory S_i is predominantly influenced by frames captured during the current stop, while older frames contribute negligible weight. Long-time memory L_i at time step i is simply defined as temporal image (1), but without considering frames in the current series, i.e., frames captured during the last stop of the camera at the observed preset position. The algorithm for computing long- and short-time temporal channels is illustrated in Figure 5. Long-time memory takes into the account only frames that are at least 2 min old and it is not affected by most recent series of frames.

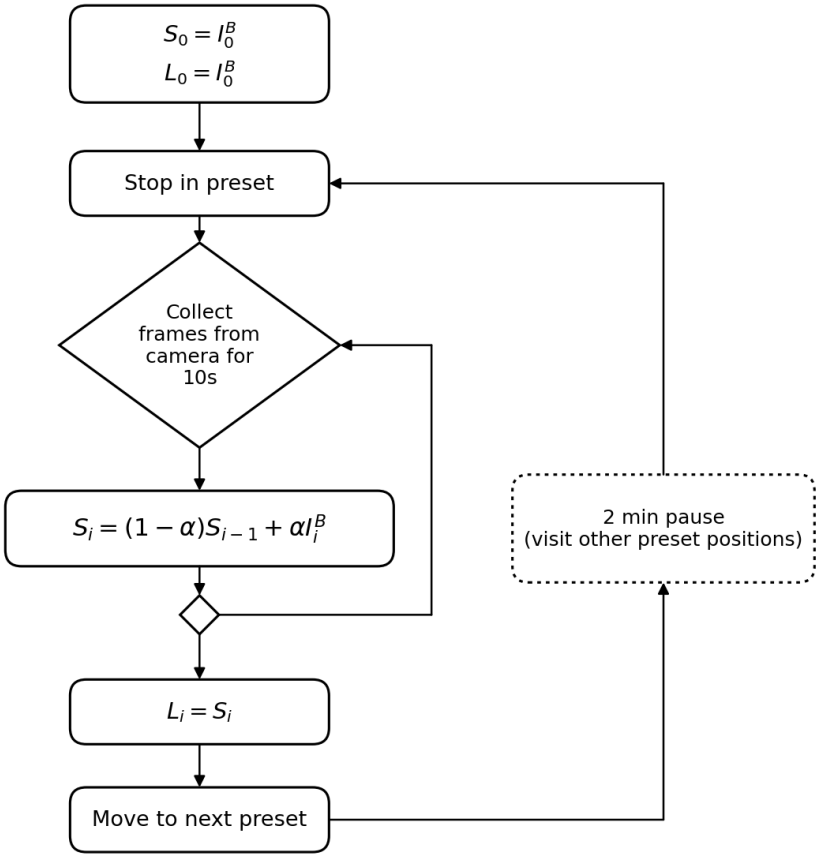


Figure 5. Short- and long-term memory channels.

The short-term and long-term temporal information were encoded as a dual-channel image with the same dimensions as the original frame captured from the camera.

3.2.2. Short-Time Foreground Estimation

Short-time foreground image is based on the running average background subtraction with the adaptive threshold [38]. Limited temporal window was used for the computation of the short-time foreground, where only frames from the current stop in the observed preset position were considered. This short temporal window allows the detection of rapid subtle changes that characterize early smoke.

Each time the camera stops at the observed preset position, the background image is reset to the current frame $B_i = I_i$ and foreground is set to an empty image. Every subsequent frame updates the background and foreground images. First, difference between current frame and background image is computed:

$$D_i = |I_i - B_i| \tag{2}$$

A foreground F_i is a blob of pixels at time step i, mathematically defined by:

$$F_i = \{x : D_i(x) > T_i(x)\}, \tag{3}$$

where $T_i(x)$ is the threshold value and $D_i(x)$ is difference (2) at pixel x. Background is updated only in the region of the image not included in the foreground:

$$B_{i+1} = \begin{cases} (1-\alpha)B_i(x) + \alpha I_i(x), & \text{for } x \notin F_i \\ B_i(x), & \text{for } x \in F_i \end{cases} \tag{4}$$

with α set to 0.1.

Adaptive threshold is based on the premise that threshold should be higher in the parts of the scene where there is inherent flickering and change, i.e., in parts that depicts a landscape whose characteristics are such that even when there are no significant events in the image, there is a difference in pixel values in successive frames. Threshold is updated for each pixel using:

$$\hat{T}_{i+1} = (1-\alpha)T_i + \alpha D_i \tag{5}$$

$$T_{i+1} = \max(\hat{T}_{i+1}, 0.7t), \tag{6}$$

where $\hat{T}_{i+1}$ denotes the intermediate threshold value obtained from the exponential smoothing update, and T_{i+1} is the final threshold after applying the lower-bound constraint. At the very beginning of the sequence, threshold T_0 is set to the initial value $t = 0.05$ for all pixels. Adaptive threshold should represent long-term dynamic characteristics of the observed region in the landscape, not only last few seconds. Thus, threshold T_i is computed using whole history and it is not reset to T_0 every time camera stops at the observed preset position. Equation (6) ensures that the foreground detection threshold cannot become too low, i.e., it can never fall below 70% of the initial threshold value. The difference D_i and the adaptive threshold T_i were encoded as a dual-channel image with the same dimensions as the original frame.

3.2.3. Distance Channel

Based on the precise location and height at which each camera is mounted, a digital twin of the camera environment has been developed, featuring an accurate terrain model for all cameras included in the system. This model enables the calculation of the distance of each pixel in the image, that is, the distance of the terrain represented by that pixel from the camera itself. Distance, given in meters, is normalized by a fixed value of 25,000. Pixels representing a distance greater than 25 km are assigned a value of 1. This saturation

threshold reflects the maximum operationally relevant detection range of the surveillance system, beyond which smoke is rarely discernible under typical atmospheric conditions. This distance information is encoded as a single-channel image with the same resolution as the original frame taken from the camera, representing a relative distance map.

By combining the original RGB frames and the above-described approaches for generating temporal and contextual information, seven different data models were constructed:

1. RGB—Original RGB image taken from the camera;
2. Temporal image—Three-channel image consisting of the blue channel of the current frame and two channels representing short-term and long-term memory, as defined in Section 3.2.1.
3. RGB + Temporal—Five-channel samples consisting of the original RGB image and two channels representing short-term and long-term memory.
4. RGB + Distance—Four-channel samples containing RGB image and relative distance from the camera for each pixel.
5. RGB + Temporal + Distance—Six-channel samples consisting of the original RGB image, short- and long-term memory, and relative distance for each pixel.
6. RGB + Foreground—Five-channel image, consisting of RGB channels and two additional channels defined in Section 3.2.2: D_i (2) representing difference between the current frame and short-term background, and T_i (5) representing long-term dynamic characteristics of the image.
7. All channels (RGB + Temp. + Dist. + Fgr.)—Eight-channel data samples consisting of the original RGB image and all additional channels defined above.

For the evaluation of the proposed data models, separate dataset for each model was generated from the sequences collected from the archive of the wildfire surveillance system. The baseline dataset containing RGB samples was also generated and used for comparative analysis of other data models.

3.3. Data Model Evaluation

Fixed-size data samples of 640 × 640 pixels were extracted from the original sequences. For sequences containing annotated smoke regions, the samples were generated by positioning a 640 × 640 window around the labeled smoke region with a random horizontal and vertical shift, ensuring that the target phenomenon is present within the extracted sample. When generating positive samples, i.e., samples containing smoke, care was taken to ensure that at least 4 s had elapsed between two frames used to extract smoke samples to avoid adding nearly identical smoke representations.

Negative samples were created from both sequences with smoke and sequences without smoke, by randomly positioning 640 × 640 window over a region without smoke. When selecting negative samples, areas of image at the top of the frame, typically representing sky above the horizon, and the bottom region of the image, representing terrain near the camera, were skipped. Frames spaced by at least two minutes were used to generate negative samples. Additionally, for sequences containing smoke negative samples were not taken after the first occurrence of smoke in the sequence.

For each unique point in the spatio-temporal coordinates, that is, for each individual 640 × 640 window in a specific frame, a total of 7 data samples were generated according to the previously defined seven data models. In this way, it is ensured that all datasets contain an equal number of samples, with each sample generated from the same region in a particular frame, having a counterpart in each of the datasets. Separate datasets were generated for training, validation and testing. Division of locations used to collect sequences into the non-overlapping subsets ensures that samples in training, validation and test datasets are completely distinct, i.e., they represent geographically separated

landscapes. The test dataset was set aside and not used until the final evaluation of the models on the original high-resolution sequences. Number of positive and negative samples in training, validation, and test datasets are given in Table 2. The training set was generated in two ways: (a) without data augmentation, shown in the first row of Table 2, and (b) with data augmentation applied, shown in the second row of the same table. A detailed description of the augmentation procedure is given in Section 3.4.

Table 2. Number of extracted 640 × 640 patches.

Data	Samples	Smoke	No Smoke
Train (no augmentation)	9077	5238	3839
Train (augmented)	74,502	37,251	37,251
Validation	1694	847	847
Test	1854	927	927

Compiled datasets were used to train standard YOLOv8 architecture. To accommodate multichannel input images with C channels, the only architectural modification required was adjusting the input layer to accept tensors $\mathbf{X} \in \mathbb{R}^{H \times W \times C}$ instead of the standard $\mathbf{X} \in \mathbb{R}^{H \times W \times 3}$. Four YOLOv8 models (nano, small, medium and large) were trained for each data model. In this experiment, YOLOv8 architecture was used because of its good performance in small object detection. The aim of this evaluation step was to select the data model most suitable for early smoke detection. For this experiment, the training dataset generated without data augmentation was used in order to train and evaluate a large number of YOLO models (28 models in total) within an acceptable time frame. The average training time for a single model was approximately 6 h, resulting in a total training time of roughly 168 h for all 28 models in this first evaluation stage. All experiments were conducted on two NVIDIA Tesla T4 graphics accelerators, each equipped with 16 GB of VRAM. Evaluation of the trained models was conducted on the validation dataset with the aim of selecting the best data model for early smoke detection.

3.4. Multichannel YOLO Training on Augmented Dataset

Based on the evaluation conducted in the previous step, the best data model was selected. In the subsequent phase, the best-performing model was retrained on a larger dataset. To increase the training dataset size, augmentation techniques were used. The extended training dataset was generated only for the data model selected in the previous step—i.e., for the data model that yielded the best results when YOLOv8 models were trained on the smaller, non-augmented dataset. As part of generating the extended dataset, RGB samples were also generated. This dataset was used to compare performance of standard YOLO architecture trained on multichannel data versus the same architecture trained on the RGB images. Augmentation was not used for generating the validation dataset and the testing dataset. These datasets consistently contained only original samples generated directly from the sequences obtained from the archive throughout the entire study.

When applying augmentation techniques, particular attention was given to the specific approach for generating samples from the sequence of images. Specifically, augmenting individual frames would introduce inconsistencies in the temporal channels, since those channels encode information derived from the entire sequence history. Changes resulting from frame-level augmentation—including illumination shifts, geometric transformations, and camera shake simulation—affect temporal channels in ways that cannot be easily modeled or compensated analytically. Augmentation must therefore be applied consistently at the sequence level. Therefore, a specific augmentation strategy was developed to increase the training dataset, based on two fundamental premises:

1. No changes can be applied directly to channels based on the dynamic properties of the image, i.e., calculated from a sequence of successive images. All modifications can only be applied to the original RGB images retrieved from the camera.
2. All augmentation procedures applied to a sequence of images must be consistent with possible changes that could realistically occur between two consecutive frames. For example, any augmentation based on changes in image geometry must be applied to all images in the sequence in the same way and with the same parameters.

Based on the above premises, for each sequence in the training dataset, an augmentation pipeline was implemented using 3 separate sets of transformations:

1. Sequence transform—set of transformations applied to the entire sequence. Transformations were applied to each frame in the sequence with exactly the same parameters for all transformations in the set.
2. Series transform—set of transformation applied to all frames in a series, i.e., to a set of consecutive frames captured at one-second intervals during a single stop at a preset position. This set of transformations reflects changes that may occur during a period of approximately two minutes, which is the time it takes for the camera to return to the same preset position. Transformations were applied to each frame in the series with exactly the same set of parameters.
3. Frame transform—set of transformations applied to each frame in the sequence with the randomly generated parameters. These transformation correspond to the changes that can occur in one second.

The augmentation pipeline was started a total of 7 times on all sequences in the training dataset. Sequence transform settings for each augmentation pipeline are provided as columns in Table 3. The first row of the table defines a fixed scaling factor, set to 2 in the second pipeline run, 0.6 in the third run and fixed to 1 (original size) for other runs. The remaining rows represent the probability of a particular transformation being used in Sequence transform set in the observed augmentation pipeline run. In each augmentation pipeline run, the probability of applying one of the transformations was fixed to 1, while the probability of applying other transformations is given in the corresponding column of the table. For example, in the first run, a horizontal flip was applied to all images, the frame size was fixed to the original resolution, while the probability of applying other transformations was set to 0.2. In the second and third runs, a fixed scaling factor was established, and the probability of other transformations was set to 0.2. The same approach was used in the remaining four runs of the augmentation pipeline.

Table 3. Augmentation strategies: sequence transform parameters for different transformation in each augmentation pipeline run.

Transformation	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Scale	1.0	2.0	0.6	1.0	1.0	1.0	1.0
Random scale	0.0	0.0	0.0	0.2	0.2	0.2	0.2
Horizontal flip	1.0	0.2	0.2	0.2	0.2	0.2	0.2
Rotate	0.2	0.2	0.2	1.0	0.2	0.2	0.2
Perspective	0.2	0.2	0.2	0.2	0.2	1.0	0.2
Optical distortion	0.2	0.2	0.2	0.2	1.0	0.2	0.2
HSV	0.2	0.2	0.2	0.2	0.2	0.2	0.6
Random brightness	0.2	0.2	0.2	0.2	0.2	0.2	1.0
Random gamma	0.2	0.2	0.2	0.2	0.2	0.2	0.6

The set of transformation included in the Sequence transform set, along with the parameters for each of the included transformation, were randomly selected at the beginning

of each augmentation pipeline run, i.e., at the first frame in the sequence, and were fixed for all frames in the sequence. Data augmentation was implemented using Albumentations library [39].

Series transform set was applied to a series of consecutive frames captured during a single stop at the observed preset position. Series transform augmentation was based on RandomFog transformation, applied with the probability of 0.5 and the fog density coefficient with upper limit set to 0.15. This low fog density simulates realistic atmospheric changes that may occur within the approximately two-minute interval required for the camera to return to the same preset position. In addition, a Frame transform based on the ISONoise augmentation was applied independently to each frame in the sequence to simulate sensor noise. Each augmentation run employed independently randomized transformation parameters, ensuring diversity among the augmented sequences.

Samples for the extended training dataset were extracted from the augmented sequences on-the-fly, without the generated sequences themselves being stored. The same procedure was followed for sampling as it was used for obtaining samples from the original sequences. When extracting positive (smoke) samples, a minimum temporal gap of 4 s between two frames from which the sample were generated was enforced. For the negative sample, a temporal gap of 2 min was enforced and no sample was extracted after the first occurrence of the smoke in sequence. The total number of positive samples in the augmented dataset is shown in the second row in Table 2. The total number of samples in the extended training set is approximately 8 times greater than the number of samples generated without augmentation, corresponding to the original sequence and seven augmented variants. Samples were generated only for the data model that demonstrated the best results in the previous step (training on the limited dataset). In addition to this, standard RGB image samples were also generated. Each sample in the extended training dataset represents a unique point in the spatio-temporal coordinates and has a counterpart in both datasets (multichannel and RGB).

The multichannel YOLOv8 model was trained on the extended dataset for the selected data model. RGB data samples were used to train YOLOv8, YOLOv11, and YOLOv12 architectures. Training on the augmented dataset was conducted for four model sizes (nano, small, medium, large) for each combination of data model and YOLO architecture. In total, sixteen models were trained on the extended dataset. A comparative analysis of the performance of all trained models was conducted on the validation dataset. It should be noted that YOLOv11 and YOLOv12 were released during the later stages of this research and were included as additional baselines trained on standard RGB data. Their inclusion serves to demonstrate that the performance gains of the proposed 5-channel spatio-temporal data model hold even when compared against more recent and advanced architectures, thereby strengthening the central hypothesis that encoding spatio-temporal information into additional image channels improves detection performance independently of the underlying architecture.

3.5. Evaluation on High-Resolution Sequences

The final evaluation was conducted by simulating a real production environment of the wildfire surveillance system. In this experiment, sequences from the test dataset were used, collected from set of geographically distinct surveillance locations that were not used for collecting data samples for the training or validation datasets. The test dataset was not used in any of the previous steps. Data collection from geographically distinct locations ensures that the experimental evaluation was conducted on landscapes and environments that none of the trained models had previously encountered.

The experiment was conducted by retrieving frames from the sequence in the order they were stored, with appropriate time intervals between consecutive images, which corresponds to the actual operating conditions where images are retrieved from the camera in the same manner. Acquisition of images from the camera is described in detail in Section 3.1. All trained YOLO models were applied to the original frames from the test sequences without any pre-processing and at full resolution (1920 × 1080) to fully simulate the production environment of an early wildfire detection system.

3.6. Evaluation Metrics

For early wildfire detection, both limiting false alarm rate and lowering the number of missed detections are critical. Therefore, the models developed in this work must simultaneously achieve high precision and high recall. Precision measures the proportion of true-positive (TP) wildfire detections relative to the total number of predicted detections and reflects the model's ability to suppress false positives (FP), thus increasing operational trust in automated alerts. Recall quantifies the proportion of correctly detected wildfire smoke instances relative to the total number of instances, i.e., manually labeled instances of smoke by a human expert, and indicates the model's capability to minimize missed detections. Precision–Recall (PR) curves are used to illustrate the trade-off between these two metrics across different confidence thresholds. Lower confidence thresholds generally increase recall while allowing for more false positives, whereas higher thresholds reduce false detections but risk missing true events. The $F1$-score, defined as the harmonic mean of precision and recall, is used in this work as a primary indicator of overall detection performance. It is expressed as

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

In the first two evaluation stages (Sections 3.3 and 3.4), the F_1-score, Precision, and Recall were obtained using the Ultralytics YOLO validation framework on the validation dataset. In the final evaluation step, experiment was conducted on 46 high-resolution sequences in the testing dataset that had not been presented to the algorithm until that point. For detection models that utilize temporal channels, temporal information was computed on-the-fly, during the analysis of each individual sequence and compiled into the multichannel data samples, which were then fed to the detection algorithm. Evaluation metrics were computed using a custom evaluation pipeline that aggregates true positives, false positives, and false negatives for each sequence. In this step, the simulation environment also tracked the model responsiveness, defined as the delay between the first ground-truth appearance of smoke in the sequence (hand-labeled by a human expert) and the first true-positive detection.

In addition, inference speed was evaluated, defined as the time required to process a single frame on the target hardware accelerator. Given the intended integration of the proposed models into an operational wildfire surveillance system, inference latency is a key metric determining system scalability. Inference performance is evaluated only in the final stage, where high-resolution input images are used to reflect deployment conditions. Although absolute latency in real deployments may vary, relative performance trends are expected to scale consistently.

4. Results

This section presents the results of the experimental evaluation. The evaluation process was conducted in three steps, as described in Section 3. First, the effectiveness of the YOLOv8 model trained on different multichannel data models was compared. In the

second stage of evaluation, the YOLOv8 model was trained on the multichannel dataset that yielded the best results in the previous step. In this step augmentation was used to enlarge the training dataset. The obtained results were compared with those achieved by training the YOLOv8, YOLOv11, and YOLOv12 models on RGB data. In the first two steps, the effectiveness of the all detection algorithms was evaluated using the validation dataset. In the final step, evaluation was conducted on sequences from test dataset at full resolution, designed to simulate real deployment conditions and assess operational responsiveness using the proposed delay metric.

4.1. Data Model Selection

In order to select optimal multidimensional data model for early wildfire detection, all generated multidimensional datasets, described in Section 3.2, were evaluated by testing detection performance of four YOLOv8 models (nano, small, medium and large) for each data configuration. All models were evaluated using validation dataset. Evaluation results in terms of F_1-score for all YOLO models and datasets are given in Table 4, with each row corresponding to one multidimensional dataset. Top two results for all dataset–YOLO model combinations are highlighted in bold. The F_1-score for different combinations of multichannel data and YOLO models clearly shows that multidimensional datasets containing channels based on dynamic changes in scene enable more accurate predictions compared to datasets that contain only spatial-domain information (original RGB image and original image with per-pixel distance). Moreover, datasets that include dynamic information based on short-term and long-term memory, as defined by Equation (1), outperform the data with temporal information based on short-term foreground extraction. The highest F_1-scores were achieved by training YOLO models on a 5D dataset that combines RGB image with short-term and long-term temporal information (1) based on blue channel of the image, and on the 8D dataset that contains all channels in both spatial and temporal domains.

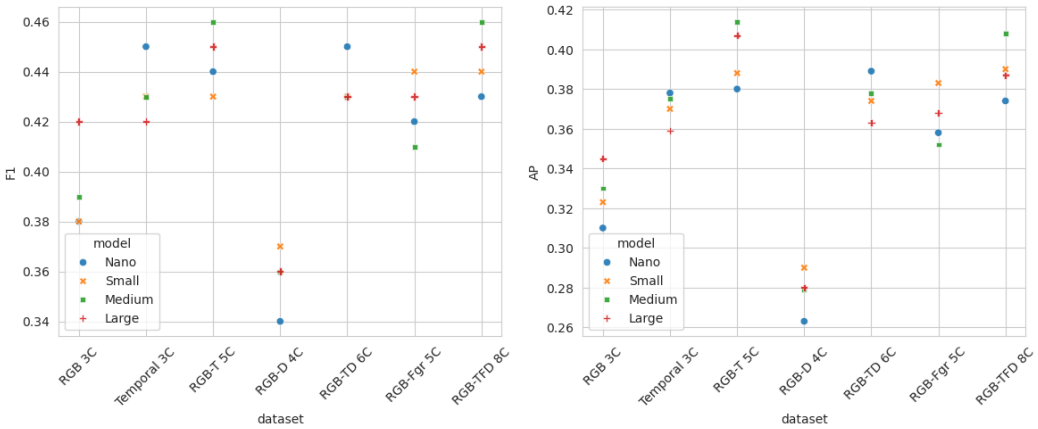
Table 5 shows the average precision for all combinations of datasets and YOLO models, with the top two results highlighted in bold. The best results were achieved by training on a five-dimensional dataset comprising RGB data and temporal information derived from the blue channel. The results from Tables 4 and 5 are graphically presented in Figure 6.

Table 4. F_1 measure for different multichannel datasets, trained on raw data (no augmentation), evaluated on validation dataset.

	Num.Ch.	Nano	Small	Medium	Large
RGB	3	0.38	0.38	0.39	0.42
Temp.	3	**0.45**	0.43	0.43	0.42
RGB + T.	5	0.44	0.43	**0.46**	**0.45**
RGB + Dist.	4	0.34	0.37	0.36	0.36
RGB + TD	6	**0.45**	0.43	0.43	0.43
RGB + Fgr.	5	0.42	0.44	0.41	0.43
RGB + TFD	8	0.43	0.44	**0.46**	**0.45**

Table 5. Average Precision (*AP*@0.5) for different multichannel datasets, trained on raw data (no augmentation), evaluated on validation dataset.

	Num.Ch.	Nano	Small	Medium	Large
RGB	3	0.310	0.323	0.330	0.345
Temp.	3	0.378	0.370	0.375	0.359
RGB + T	5	0.380	0.388	**0.414**	0.407
RGB + Dist.	4	0.263	0.290	0.279	0.280
RGB + TD	6	0.389	0.374	0.378	0.363
RGB + Fgr.	5	0.358	0.383	0.352	0.368
RGB + TFD	8	0.374	0.390	**0.408**	0.387



(**a**) F_1 measure (**b**) Average Precision

Figure 6. F_1 measure and Average Precision for different multichannel datasets, trained on raw data (no augmentation), evaluated on validation dataset. From left to right: RGB images (3 channels); Temporal image (3 channels); RGB + Temporal image (5 channels); RGB+distance (4 channels); RGB + Temporal + distance (6 channels); RGB+Foreground (5 channels); RGB + Temporal + Foreground + distance (8 channels).

It can be observed that the model that includes pixel-to-camera relative distance alongside RGB channels performed worse than the RGB-only model. This may result from the normalization step, which hinders clear interpretation of distance values relative to RGB color channels, as well as from errors and noise in distance maps that are often difficult to perfectly align with the actual camera field of view. The observed degradation suggests that naive inclusion of distance information is insufficient without precise geometric alignment and calibration, which remains a challenging problem in real-world deployments.

4.2. Training on Augmented Data

Evaluation of the models trained on non-augmented datasets showed that the best-performing multidimensional data configuration was obtained by combining the original RGB images with short-term and long-term memory based on the blue image channel. In this step an extended training dataset was generated using augmentation procedure described in Section 3.4. Training data samples were generated for the selected 5-channel data model as well as for the standard RGB data. Four standard YOLOv8 models (nano, small, medium and large) were trained on 5-channel data. YOLOv8, YOLOv11 and YOLOv12 model were trained on the RGB data. The effectiveness of all trained models was compared on the validation dataset.

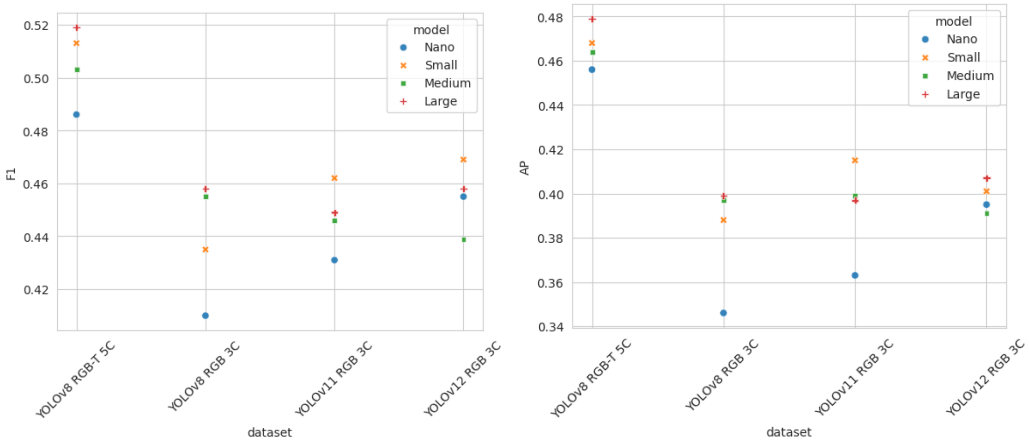
Table 6 presents the F_1 score results for all models, with two best results highlighted in bold. Average precision for all models is shown in Table 7. Graphical representation of the results from Tables 6 and 7 is shown in Figure 7.

Table 6. F_1 measure for YOLOv8 trained on 5D spatio-temporal data vs. YOLOv11 and YOLOv12 trained on 3D RGB data, evaluated on the validation dataset.

	Dataset	Nano	Small	Medium	Large
YOLOv8	Spatio-temporal 5D	0.49	**0.51**	0.50	**0.52**
YOLOv8	RGB 3D	0.41	0.44	0.46	0.46
YOLOv11	RGB 3D	0.43	0.46	0.45	0.45
YOLOv12	RGB 3D	0.46	0.47	0.44	0.46

Table 7. Average Precision (*AP*@0.5) for different multichannel datasets.

	Dataset	Nano	Small	Medium	Large
YOLOv8	Spatio-temporal 5D	0.46	**0.47**	**0.47**	**0.49**
YOLOv8	RGB 3D	0.35	0.39	0.40	0.40
YOLOv11	RGB 3D	0.36	0.42	0.40	0.40
YOLOv12	RGB 3D	0.40	0.40	0.39	0.40



(**a**) F_1 measure (**b**) Average Precision

Figure 7. F_1 measure and Average Precision for YOLOv8 spatio-temporal 5-channel, YOLOv8 RGB, YOLOv11 RGB and YOLOv12 RGB trained on an augmented dataset and evaluated on a validation dataset.

The experiment results clearly indicate the benefits of compiling temporal information into additional image channels, as YOLOv8 models trained on multichannel data clearly outperform all models trained on RGB data.

4.3. Evaluation on Sequences

The final evaluation stage assesses model performance under realistic deployment conditions, using full-resolution sequences from geographically distinct surveillance locations not used in any prior training or validation step. YOLO models evaluated in this final experiment are the same models that were trained on the augmented training dataset in the previous step.

The evaluation was performed separately on each of the sequences from the testing dataset. Frames from a sequence were taken in the order they were retrieved from the camera and used to compute long-term and short-term memory. This temporal information was compiled on-the-fly with the original RGB channels and fed into the YOLOv8 models trained on 5-channel data. Inference was performed on full-resolution images. Detection was also performed on the original RGB image frames using YOLOv8, YOLOv11 and YOLOv12 models trained on augmented dataset in the previous evaluation step.

Figure 8 shows F_1-score over confidence thresholds for all models. A wider F_1 curve—i.e., a broader range of confidence values over which the model achieves high performance—indicates greater stability and lower sensitivity to the choice of detection threshold. The best result in F_1-score was achieved by YOLOv8 medium model trained on 5-channel spatio-temporal data. This model has an F_1-score exceeding 0.8 for a threshold interval of approximately 0.2 to 0.6, indicating that it provides relatively high precision and recall for this range of confidence values. Relatively good results were also achieved by other YOLOv8 models trained on 5-channel data. YOLO models trained on RGB images generally achieved lower results, with the YOLOv12 model performing the best among

3-channel models, but with a narrower F_1 curve, indicating a higher sensitivity to the choice of detection threshold.

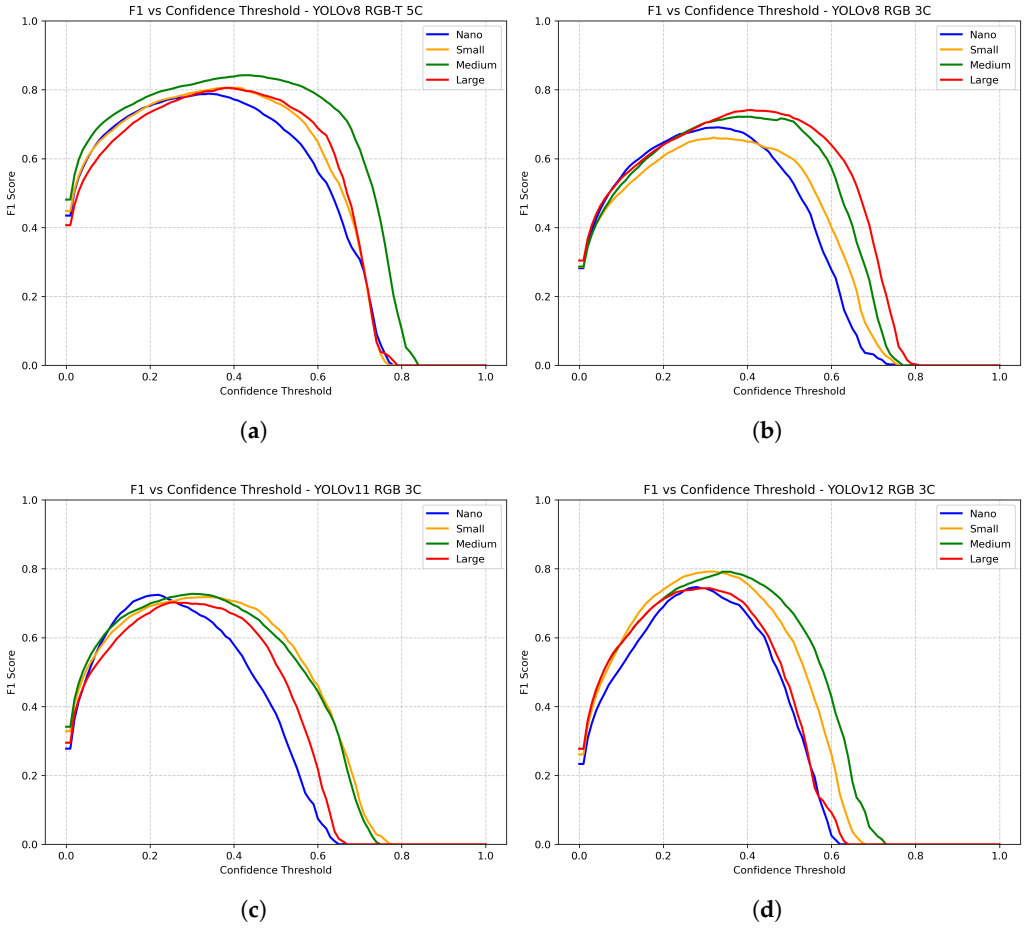


Figure 8. F_1-score for different confidence thresholds for YOLOv8 models trained on 5-channel spatio-temporal data, as well as YOLOv8, YOLOv11 and YOLOv12 models trained on RGB images. The evaluation results on full-resolution sequences collected from a separate set of surveillance locations (not used for training or validation) are presented. (**a**) YOLOv8 trained on 5-channel data (RGB + Temporal); (**b**) YOLOv8 trained on RGB data; (**c**) YOLOv11 trained on RGB data; (**d**) YOLOv12 trained on RGB data.

The higher F_1-scores on test sequences compared to validation results are attributable to the sample-extraction protocol. Validation patches (640 × 640 px) excluded the top of the frame (sky) and the bottom region (near-camera terrain). Full-resolution sequence evaluation includes these regions, which trained models seldom misclassify, thereby inflating the aggregate F_1-score. This discrepancy is an inherent limitation of patch-level evaluation: by excluding easily classified regions such as sky and near-camera terrain, patch-level metrics do not fully reflect real-world performance. Consequently, patch-level and sequence-level results should not be directly compared; patch-level evaluation serves primarily as a model selection criterion, while sequence-level evaluation on geographically distinct test locations provides the most reliable assessment of operational performance.

To assess the suitability of all models for deployment in real-world wildfire detection system, detection delay was measured on the test set of sequences. We define detection delay as the time elapsed from the first appearance of smoke in a sequence to the corresponding first true-positive (TP) detection, measured in seconds. The ground truth for the first appearance of smoke in a sequence is defined as the frame in which a human expert first annotated smoke. It should be emphasized that this first sign of smoke may be very

difficult to spot. When annotating sequences, human experts were allowed to navigate backward through the sequence after first marking of smoke on a frame, and trace the same smoke to earlier frames. It should be noted that, since human experts were allowed to trace smoke backward through the sequence after the initial annotation, the first frame with GT smoke tag may represent an earlier and more challenging detection point in which it would be very difficult to detect smoke in real time. Consequently, "no delay" detections should be interpreted as cases where the algorithm matched the expert's retrospective annotation, which represents a particularly challenging detection point for an automated system operating in real time without the ability to review past frames retrospectively.

When measuring detection delay, smoke occurrences that could be continuously tracked across multiple frames in a sequence were treated as independent instances. Multiple independent smoke events could occur within a single sequence, especially during agricultural burnings. There were 86 such independent instances across all test sequences. Further, for a detection to be considered as a successful true positive (TP), a 10 min limit was imposed, based on the assumption that the first 10 min represent the critical early phase of wildfire in which timely detection can significantly affect suppression success. Detections delayed by more than ten minutes were accounted for as false-negative (FN) detections.

The detection delay measurement results for different combinations of YOLO models and data models are presented in Table 8. The first column shows the number of TP detections, followed by the number of FN detections in the first 10 min. The "No delay" column represents the number of detections where smoke was detected on the same frame annotated by an expert, i.e., the number of detections with no delay. The last two columns represent the mean delay and the maximum detection delay for the successful detections. When computing detection delay, the interval between two consecutive frame series captured during consecutive stops at the observed preset position was fixed at 2 min. For example, if the detector did not detect smoke in a series of frames captured at one-second intervals but detected it in the next series, the elapsed time between these two series was counted as 2 min. In practice, this interval may vary slightly between sequences, but such variation has a negligible effect on the experimental results.

In terms of the ratio of true-positive detections, models trained on 5-channel data yield, on average, 2 to 5 percent better results compared to other models trained on RGB images. Even small improvements in early wildfire detection can have disproportionately large real-world impact, as each additional early detection increases the likelihood of successful containment. Although the YOLOv12 model shows slightly lower true-positive detection rates, it is interesting to note that it achieves shorter detection delays for successful detections. This difference may be attributed to the attention mechanism integrated into YOLOv12, which may enable faster localization of subtle smoke regions once a detection is triggered.

Representative examples of true positive detections and misclassifications observed during the sequence evaluation are shown in Figure 4. The most common sources of false alarms include clouds and sunlight effects. FP detection shown in Figure 4a represents clouds near the horizon at distances exceeding 25 km, which could potentially be suppressed using the available per-pixel distance map. FP detections in Figure 4b include lens flare and sunlight reflections. These types of false alarms tend to be temporally clustered in morning hours at east-facing camera positions and evening hours at west-facing camera positions, suggesting that temporal post-processing could reduce their occurrence. In Figure 4c, simultaneous TP and FP detections under overcast conditions are shown. Subsequent careful inspection revealed that this FP detection corresponds to actual smoke that was missed during the manual annotation process, suggesting it in fact represents a true positive.

Table 8. Successful detections and detection delay. Columns from left to right: TP detections, FN detections, the percentage of TP detections, the number of detections with no delay, mean delay in seconds (s), maximum detection delay (s) for the successful detections.

RGB-T 5C YOLOv8	Detected (TP)	Missed (FN)	Rate (%)	No Delay	Mean Delay	Max Delay
Nano	54	32	62.8	44	74.4	210
Small	59	27	68.6	48	67.3	210
Medium	57	29	66.3	47	74.1	315
Large	57	29	66.3	50	46.3	210
RGB 3C YOLOv8						
Nano	56	30	65.1	41	77.5	315
Small	55	31	64.0	44	96.6	316
Medium	59	27	68.6	50	70.6	315
Large	51	35	59.3	42	105.6	316
RGB 3C YOLOv11						
Nano	56	30	65.1	44	105.4	316
Small	55	31	64.0	41	98.1	525
Medium	51	35	59.3	41	84.6	525
Large	53	33	61.6	46	45.6	105
RGB 3C YOLOv12						
Nano	53	33	61.6	44	36.2	110
Small	57	29	66.3	52	42.6	105
Medium	52	34	60.5	44	40.4	105
Large	54	32	62.8	46	27.0	105

The F_1-score vs. inference latency on full-resolution images, as shown in Figure 9, provides the assessment of the detection algorithm usability for real-time smoke detection. Each curve corresponds to one of the trained YOLO models (YOLOv8 using 5-channel spatio-temporal data samples, and YOLOv8, YOLOv11 and YOLOv12 trained on RGB images), with F_1-score vs. latency indicated for four models sizes (Nano, Small, Medium and Large). Latency is consistent and as expected across all models: the simplest model detects fastest, followed by more complex models with increasing parameter counts. YOLOv11 is generally the fastest, as expected due to its optimizations for edge devices [19]. Attention mechanics used by YOLOv12 [20] give it a slight edge in qualitative results compared to older YOLOv11 and YOLOv8. However, these attention mechanics also heavily slow it down, making even its nano version slower than the YOLOv11 small model. Comparing YOLOv8 models based on RGB images and the same models processing 5-channel spatio-temporal data samples, it can be seen that the 5-channel models have slightly higher inference time. However, this difference is not significant. On the other hand, the F_1-score for the 5-channel models is significantly better compared to all other models based on 3-channel RGB images.

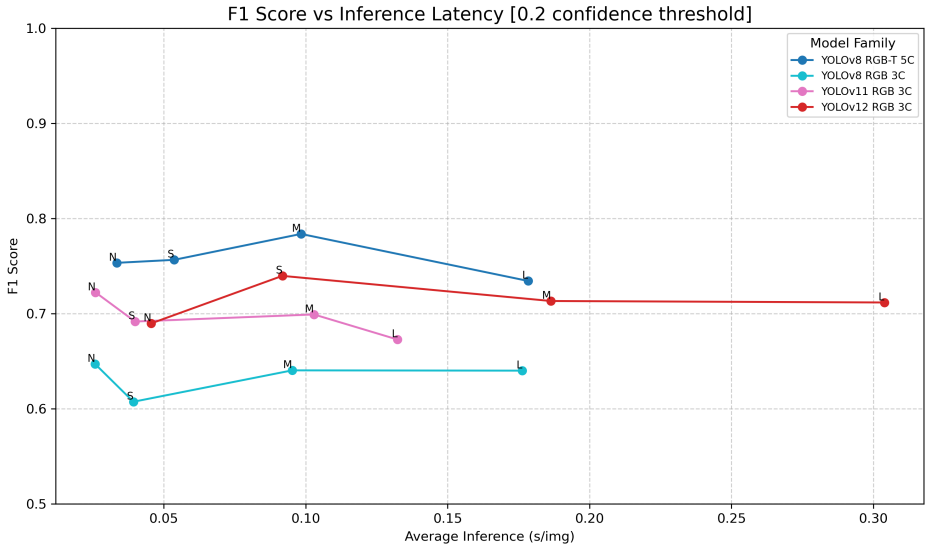


Figure 9. F_1-score vs. inference latency for different data models and YOLO versions. Time in seconds on the x-axis refers to the processing time of a single frame from the full-resolution sequences (1920 × 1080).

5. Discussion

The experimental results demonstrate that encoding spatio-temporal information directly into additional image channels consistently improves early wildfire smoke detection compared to models trained on standard RGB images. The following discussion analyses the observed performance differences between data model configurations, situates the proposed approach within the broader context of related work, and considers its practical implications for operational wildfire surveillance systems.

Among the evaluated data models, the 5-channel spatio-temporal configuration combining RGB images with short-term and long-term temporal memory based on blue channel consistently achieved the best results, outperforming other compiled data models, including the temporal data model based on short-time foreground estimation. The memory-based temporal encoding appears better suited for this problem than foreground estimation, likely because it captures changes at two complementary temporal scales that align naturally with the image acquisition protocol. Long-term memory preserves the scene state from the previous camera visit to the observed preset position, approximately two minutes earlier, enabling the detection of changes that accumulated over this interval—including more pronounced smoke expansion. Short-term memory captures faster changes occurring between consecutive frames at one-second intervals, while its slow update rate prevents rapid scene fluctuations from dominating the signal. This dual-scale temporal encoding is tailored to the acquisition protocol of the present surveillance system; systems with different capture procedures—such as continuously streaming cameras without inter-series pauses—would require a corresponding re-parameterization of the temporal memory.

Contrary to expectations, the inclusion of the per-pixel distance map did not improve detection performance and in fact slightly degraded results compared to the RGB baseline. Although preliminary results from our prior work [16] suggested a potential benefit of contextual distance information, evaluation on the larger dataset with a more rigorous experimental design did not confirm this finding. Several factors may contribute to this outcome: camera calibration in remote outdoor environments is inherently imprecise, preventing accurate alignment between the terrain model and the camera field of view. Additionally, camera shake and gradual shifts in camera orientation caused by prolonged outdoor exposure introduce further misalignment over time. Nevertheless, encoding

contextual information in alternative forms—such as terrain contour or land cover type derived from GIS data—may offer a more robust approach and is identified as a promising direction for future research.

Direct numerical comparison with results reported in other wildfire smoke-detection studies is not straightforward, as the available benchmarks differ substantially in dataset characteristics, detection distances, and evaluation protocols. The present study specifically targets very early smoke detection at large distances under real operational conditions, arguably a more challenging setting than some other published datasets [1,6]. Nevertheless, a qualitative comparison with related approaches is informative. Methods that explicitly model temporal dynamics, such as CNN-LSTM architectures [35,36] or the SmokeyNet [12] model combining LSTM and vision transformer components, have demonstrated improved detection performance over purely spatial approaches. However, their computational complexity makes them unsuitable for processing multiple high-resolution streams simultaneously or for deployment on resource-constrained edge devices [28]. For this reason, the proposed approach encodes temporal information into additional input channels rather than performing sequential intra-frame inference, preserving the simplicity and computational efficiency of standard single-stage detectors while achieving improved detection performance—making it well suited for large-scale operational deployment.

The low inference latency demonstrated on full-resolution sequences confirms the suitability of the proposed approach for real-time processing of multiple high-resolution video streams in large-scale surveillance systems. Since the proposed method operates by enriching the input data rather than modifying the network architecture, it is compatible with any CNN-based object detection framework—including potentially more efficient architectures than those evaluated in this study, such as lightweight single-stage detectors optimized for embedded hardware. This opens the possibility of deployment on resource-constrained edge devices, such as embedded hardware installed directly at remote monitoring locations, which would reduce network bandwidth requirements and enable faster local response. The analysis of false-positive detections revealed that the most common sources of false alarms—clouds near the horizon, sunlight reflections, and atmospheric haze—exhibit characteristic spatial and temporal patterns that could be exploited by post-processing strategies to further reduce false alarm rates in operational systems.

The present study focuses on a specific surveillance system with a well-defined image acquisition protocol, and the proposed temporal memory encoding is tailored to this protocol. For systems with different acquisition procedures—such as continuously streaming cameras, different frame rates, or varying inter-series intervals—the data model, particularly the definition of long-term memory, would need to be adapted accordingly. Similarly, application to other detection problems would require adjustment of the temporal and spatial scale of the encoded information to match the characteristics of the target phenomenon. Future research will explore alternative forms of contextual information encoding, such as terrain contour and land cover type derived from GIS data, which may provide more robust contextual cues than the distance map evaluated in this study. Additionally, the evaluation of other deep learning architectures beyond the YOLO family, adaptation of the approach for nighttime fire detection, and potential architectural modifications aimed at better exploiting the specific information contained in the additional channels are identified as promising directions for further development.

6. Conclusions

This work addresses the challenge of detecting smoke—the first visible sign of wildfire—at the earliest stage, at extended distances, and under potentially poor visibility conditions. The fundamental idea behind the proposed study is that the data used

for training is at least as important as the machine learning algorithm itself, if not more so. The study examined various approaches for encoding spatial, temporal, and contextual information into a multidimensional dataset, expanding the information contained in the original image captured by the camera at a unique point in spatio-temporal coordinates.

Experiments were conducted by training standard YOLOv8 architectures on different multidimensional data models and comparing the results with those obtained using the same YOLOv8 architecture on standard RGB images captured by the camera. The scientific soundness of the experiments was ensured by a rigorous approach to dataset creation, in which the training, testing, and evaluation datasets were obtained from completely separate monitoring locations. Experimental evaluation demonstrated that the 5-channel spatio-temporal data model— combining original RGB images with short-term and long-term temporal memory—achieves the best results and outperforms newer YOLOv11 and YOLOv12 architectures trained on standard RGB images. Furthermore, by enriching the training data rather than modifying the network architecture, the proposed approach avoids significant increases in model complexity and training time while preserving the inference speed required for near-real-time deployment in operational wildfire monitoring systems.

The main contribution of this study is the innovative model for encoding temporal information into additional image channels, which enables the use of standard machine learning models without significant increases in training complexity and inference time. Furthermore, a technique for augmenting the temporal sequence of images has been proposed that takes into account the interdependencies and changes within the sequence. Although this study focuses on the early detection of wildfires, the proposed data modeling approach could also be useful in other problems where temporal or contextual information can enhance the effectiveness of detection algorithms without significant interventions in the algorithm's architecture and without considerable increases in complexity. This is particularly relevant for domains where the target phenomenon lacks clearly defined spatial features, appears at large distances, or is observed under degraded imaging conditions—such as maritime surveillance, small object detection in aerial imagery, traffic monitoring in adverse weather, and other applications involving rapidly changing environments. By integrating temporal information directly into the data, the need for multi-step image analysis can be avoided, which often includes background subtraction or other area of interest extraction procedures.

Future research will involve the analysis and examination of additional data models, evaluation of other deep learning architectures with multichannel image data, and the adaptation of architectures aimed at better leveraging the specific information contained in the additional image channels, while maintaining the simplicity and efficiency of architectures oriented toward static data. In the field of wildfire detection, the same approach will be attempted for detecting fires at night.

Author Contributions: Conceptualization, D.K.; Methodology, D.K.; Software, J.B. and T.S.; Validation, D.K., J.B., T.S. and M.B.; Formal analysis, D.K. and M.B.; Investigation, D.K.; Resources, D.K. and M.B.; Data curation, D.K. and J.B.; Writing—original draft, D.K.; Writing – review & editing, D.K., J.B., T.S. and M.B.; Visualization, J.B.; Supervision, D.K. and M.B.; Project administration, D.K. and M.B.; Funding acquisition, D.K. and M.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was co-financed by the Research, Development and Innovation (IRI) Program of Split-Dalmatia County, Croatia.

Data Availability Statement: The code used in this study is publicly available on GitHub, including the modified YOLOv8 implementation adapted to accept multichannel input (https://github.com/dkrst/yolov8 branch multichannel, accessed on 18 March 2026) and the scripts for generating multi-

channel data samples from image sequences, including the sequence-aware augmentation pipeline (https://github.com/dkrst/sequence_tools, accessed on 18 March 2026). The dataset, exceeding 400 GB in total size, is available upon reasonable request, subject to the data sharing agreement with the system operator, Odašiljači i veze d.o.o.

Acknowledgments: All data used in this research were obtained from the Fire Detect AI wildfire surveillance system [17], owned and operated by Odašiljači i veze d.o.o., Ulica grada Vukovara 269d, HR-10000 Zagreb, Croatia.

Conflicts of Interest: Authors D.K. and M.B. are co-founders and co-owners of Code Fire d.o.o. Author J.B. is employed by Code Fire d.o.o. and is a PhD student at FESB, University of Split. Author T.S. declares no conflict of interest. Code Fire d.o.o. had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results. However, under an agreement with FESB, University of Split, Code Fire d.o.o. holds the rights to commercialize the results of this research.

References

1. Casas, E.; Ramos, L.; Bendek, E.; Rivas, F. Assessing the Effectiveness of YOLO Architectures for Smoke and Wildfire Detection. *IEEE Access* **2023**, *11*, 96554–96583. [CrossRef]
2. Batista, M.; Oliveira, B.; Chaves, P.; Ferreira, J.C.; Brandao, T. Improved real-time wildfire detection using a surveillance system. In Proceedings of the World Congress on Engineering 2019, London, UK, 3–5 July 2019.
3. Bondarenko, V.; Vasyukov, V. Hardware and software complex configuration for automated wildfire detection. In Proceedings of the 2012 IEEE 11th International Conference on Actual Problems of Electronics Instrument Engineering (APEIE), Novosibirsk, Russia, 2–4 October 2012; pp. 101–104. [CrossRef]
4. Štula, M.; Krstinić, D.; Šerić, L. Intelligent forest fire monitoring system. *Inf. Syst. Front.* **2012**, *14*, 725–739. [CrossRef]
5. Saleh, A.; Zulkifley, M.A.; Harun, H.H.; Gaudreault, F.; Davison, I.; Spraggon, M. Forest fire surveillance systems: A review of deep learning methods. *Heliyon* **2023**, *10*, e23127. [CrossRef] [PubMed]
6. Sharma, J.; Granmo, O.; Goodwin, M.; Fidje, J.T. Deep Convolutional Neural Networks for Fire Detection in Images. In *Engineering Applications of Neural Networks*; Springer International Publishing: Cham, Switzerland, 2017; pp. 183–193.
7. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
8. Raita-Hakola, A.; Rahkonen, S.; Suomalainen, J.; Markelin, L.; de Oliveira, R.A.; Hakala, T.; Koivumäki, N.; Honkavaara, E.; Pölönen, I. Combining YOLO V5 and transfer learning for smoke-based wildfire detection in boreal forests. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, 1771–1778. [CrossRef]
9. Chetoui, M.; Akhloufi, M.A. Fire and Smoke Detection Using Fine-Tuned YOLOv8 and YOLOv7 Deep Models. *Fire* **2024**, *7*, 135. [CrossRef]
10. Jeong, M.; Park, M.; Nam, J.Y.; Ko, B.C. Light-Weight Student LSTM for Real-Time Wildfire Smoke Detection. *Sensors* **2020**, *20*, 5508. [CrossRef]
11. de Venâncio, P.V.A.B.; Campos, R.J.; Rezende, T.M.; Lisboa, A.C.; Barbosa, A.V. A hybrid method for fire detection based on spatial and temporal patterns. *Neural Comput. Appl.* **2023**, *35*, 9349–9361. [CrossRef]
12. Dewangan, A.; Pande, Y.; Braun, H.W.; Vernon, F.; Perez, I.; Altintas, I.; Cottrell, G.W.; Nguyen, M.H. FIgLib & SmokeyNet: Dataset and Deep Learning Model for Real-Time Wildland Fire Smoke Detection. *Remote Sens.* **2022**, *14*, 1007. [CrossRef]
13. Vdoviak, G.; Sledević, T. Temporal Encoding Strategies for YOLO-Based Detection of Honeybee Trophallaxis Behavior in Precision Livestock Systems. *Agriculture* **2025**, *15*, 2338. [CrossRef]
14. Alzahrani, N.; Bchir, O.; Ismail, M.M.B. YOLO-Act: Unified Spatiotemporal Detection of Human Actions Across Multi-Frame Sequences. *Sensors* **2025**, *25*, 3013. [CrossRef]
15. van Leeuwen, M.C.; Fokkinga, E.P.; Huizinga, W.; Baan, J.; Heslinga, F.G. Toward Versatile Small Object Detection with Temporal-YOLOv8. *Sensors* **2024**, *24*, 7387. [CrossRef]
16. Krstinić, D.; Šerić, L.; Ivanda, A.; Bugarić, M. Multichannel data from temporal and contextual information for early wildfire detection. In Proceedings of the 2023 8th International Conference on Smart and Sustainable Technologies (SpliTech), Split/Bol, Croatia, 20–23 June 2023; pp. 1–6. [CrossRef]
17. OIV Digital Signal and Networks, Odašiljači i veze d.o.o., Ulica Grada Vukovara 269d, HR-10000 Zagreb. OIV Fire Detect AI. 2025. Available online: https://oiv.hr/en/services-and-platforms/oiv-fire-detect-ai/ (accessed on 29 November 2025).

18. Jocher, G.; Stoken, A.; Borovec, J.; NanoCode012; ChristopherSTAN; Liu, C.; Laughing; Hogan, A.; Lorenzomammana; Tkianai; et al. ultralytics/yolov5: v3.0. *Zenodo* **2020**. [CrossRef]
19. Khanam, R.; Hussain, M. YOLOv11: An Overview of the Key Architectural Enhancements. *arXiv* **2024**. [CrossRef]
20. Tian, Y.; Ye, Q.; Doermann, D. YOLOv12: Attention-Centric Real-Time Object Detectors. *arXiv* **2025**. [CrossRef]
21. Tao, C.; Zhang, J.; Wang, P. Smoke Detection Based on Deep Convolutional Neural Networks. In Proceedings of the 2016 International Conference on Industrial Informatics—Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII), Wuhan, China, 3–4 December 2016; pp. 150–153. [CrossRef]
22. Zhang, A.; Zhang, A.S. Real-Time Wildfire Detection and Alerting with a Novel Machine Learning Approach. *Int. J. Adv. Comput. Sci. Appl.* **2022**, *13*. [CrossRef]
23. Gonzalez, A.; Zuniga, M.D.; Nikulin, C.; Carvajal, G.; Cardenas, D.G.; Pedraza, M.A.; Fernandez, C.A.; Munoz, R.I.; Castro, N.A.; Rosales, B.F.; et al. Accurate fire detection through fully convolutional network. In *7th Latin American Conference on Networked and Electronic Media (LACNEM 2017)*; IET: Hertfordshire, UK, 2017; pp. 1–6. [CrossRef]
24. Zhang, Y.; Rui, X.; Song, W. A UAV-Based Multi-Scenario RGB-Thermal Dataset and Fusion Model for Enhanced Forest Fire Detection. *Remote Sens.* **2025**, *17*, 2593. [CrossRef]
25. El-Madafri, I.; Peña, M.; Olmedo-Torre, N. Real-Time Forest Fire Detection with Lightweight CNN Using Hierarchical Multi-Task Knowledge Distillation. *Fire* **2024**, *7*, 392. [CrossRef]
26. Almeida, J.S.; Huang, C.; Nogueira, F.G.; Bhatia, S.; de Albuquerque, V.H.C. EdgeFireSmoke: A Novel Lightweight CNN Model for Real-Time Video Fire–Smoke Detection. *IEEE Trans. Ind. Inform.* **2022**, *18*, 7889. [CrossRef]
27. Cao, J.; Peng, B.; Gao, M.; Hao, H.; Li, X.; Mou, H. Object Detection Based on CNN and Vision-Transformer: A Survey. *IET Comput. Vis.* **2025**, *19*, e70028. [CrossRef]
28. Lee, S.I.; Koo, K.; Lee, J.H.; Lee, G.; Jeong, S.; O, S.; Kim, H. Vision transformer models for mobile/edge devices: A survey. *Multimed. Syst.* **2024**, *30*, 109. [CrossRef]
29. Neimark, D.; Bar, O.; Zohar, M.; Asselmann, D. Video transformer network. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 3163–3172.
30. Feichtenhofer, C.; Fan, H.; Malik, J.; He, K. Slowfast networks for video recognition. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6202–6211.
31. Xing, Z.; Dai, Q.; Hu, H.; Chen, J.; Wu, Z.; Jiang, Y.G. Svformer: Semi-supervised video transformer for action recognition. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 18816–18826.
32. Polenakis, I.; Sarantidis, C.; Karydis, I.; Avlonitis, M. Smoke Detection on the Edge: A Comparative Study of YOLO Algorithm Variants. *Signals* **2025**, *6*, 60. [CrossRef]
33. Zhang, P.; Zhao, X.; Yang, X.; Zhang, Z.; Bi, C.; Zhang, L. F3-YOLO: A Robust and Fast Forest Fire Detection Model. *Forests* **2025**, *16*, 1368. [CrossRef]
34. Zhu, W.; Niu, S.; Yue, J.; Zhou, Y. Multiscale wildfire and smoke detection in complex drone forest environments based on YOLOv8. *Sci. Rep.* **2025**, *15*, 2399. [CrossRef]
35. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
36. Krichen, M.; Mihoub, A. Long Short-Term Memory Networks: A Comprehensive Survey. *AI* **2025**, *6*, 215. [CrossRef]
37. Jakovčević, T.; Stipaničev, D.; Krstinić, D. Visual spatial-context based wildfire smoke sensor. *Mach. Vis. Appl.* **2013**, *24*, 707–719. [CrossRef]
38. Collins, R.; Lipton, A.; Kanade, T.; Fujiyoshi, H.; Duggins, D.; Tsin, Y.; Tolliver, D.; Enomoto, N.; Hasegawa, O.; Burt, P.; et al. *A System for Video Surveillance and Monitoring*; Technical Report CMU-RI-TR-00-12; Robotics Institute, Carnegie Mellon University: Pittsburgh, PA, USA, **2000**.
39. Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and Flexible Image Augmentations. *Information* **2020**, *11*, 125. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.