SpectroFire: Zero-Shot Fire and Smoke Detection via Real-Time Edge Inference and IoT Data Networking

1st HyeYoung Lee
Department of Artificial Intelligence
Korea University
Seoul, South Korea
uohesha@korea.ac.kr

2nd Muhammad Nadeem
Department of Artificial Intelligence
SPILab Corporation
Ulsan, South Korea
mnadeem@spilab.kr

***Abstract*—Wildfires pose a significant threat to human life and result in substantial economic and environmental losses. Early detection and classification of wildfires is a critical task for safeguarding life and infrastructure across diverse environments, including forests, industrial complexes, and urban areas. Existing supervised approaches rely on large labeled datasets and task-specific retraining, which limit scalability in real-world deployments. In this paper, we present SpectroFire, a lightweight CLIP-style dual encoder model for zero-shot fire and smoke detection. By combining a custom CNN-based visual encoder with contrastive alignment to textual prompts, SpectroFire eliminates the need for task-specific fine-tuning while retaining strong performance under diverse conditions. Experiments on the Kaggle fire and smoke dataset demonstrate that SpectroFire achieves 90% accuracy without fire-specific retraining and sustains real-time throughput of 2239 FPS (0.45 ms) on Raspberry Pi 5 devices. Beyond standalone inference, we evaluate edge–IoT deployment scenarios where only compact event-level outputs are transmitted instead of raw video streams. This design maintains low-latency processing and ensures rapid response for UAV-based wildfire monitoring and disaster management systems in IoT data networking.**

***Index Terms*—Zero-Shot Learning, Fire and Smoke Classification, Prompt-Based Inference, Lightweight CNN, Embedded AI, Cross-Modal Representation, IoT data networking**

I. Introduction

Wildfires are intensifying globally, driven by rising temperatures, prolonged droughts, and unsustainable land use practices. Beyond immediate destruction, wildfires cause long-term ecological degradation—including soil erosion, water contamination, and the loss of carbon-sequestering forests—thereby accelerating climate change. In addition, wildfire smoke introduces harmful particulates into the atmosphere, leading to poor air quality and increased public health risks.

Recent incidents underscore the scale and urgency of this challenge. In 2023, South Korea reported 596 forest fires, burning nearly 5,000 hectares of land [1]. In early 2025, catastrophic wildfires in Los Angeles County resulted in at least 29 fatalities and over $250 billion in damages—the most costly natural disaster in U.S. history [2]. As wildfires become more severe and unpredictable, timely and accurate detection is essential to minimize damage and support rapid response efforts.

However, wildfire detection remains difficult due to dynamic fire behavior, variable environmental conditions, and the limitations of conventional monitoring systems. Manual observation methods are prone to delays and errors, while fixed sensors are susceptible to environmental noise and are constrained in spatial coverage. Although satellite imaging offers broad monitoring, it is hindered by low temporal resolution and occlusion from cloud or smoke cover, often delaying detection when time is critical.

Conventional wildfire detection systems, such as lookout towers and manual reporting, form the basis of early warning frameworks but suffer from key limitations. Human-dependent methods can lead to delays in detection, subjective errors, and inefficiencies, particularly in large or hard-to-reach forested areas [3]. Automated sensor networks—using smoke, temperature or gas sensors—provide continuous monitoring but are highly sensitive to environmental noise (e.g., fog, dust, or controlled burns), often resulting in false alarms. Their fixed spatial deployment also limits coverage, reducing effectiveness in detecting distant or rapidly spreading fires [4].

Satellite imagery enables large-scale wildfire monitoring by detecting thermal anomalies and smoke plumes over wide areas. However, its effectiveness is constrained by low temporal resolution, limited image quality, and frequent occlusion from clouds or dense smoke, often resulting in delayed detection [5], [6].

Unmanned aerial vehicles (UAVs) equipped with RGB or thermal sensors offer greater flexibility and high-resolution data acquisition. They can rapidly scan affected areas and support localized monitoring. However, manual UAV operation is labor-intensive, and real-time onboard fire classification remains computationally demanding. These limitations underscore the need for AI-driven approaches that offer scalable, efficient, and accurate fire detection under diverse environmental conditions.

In recent years, computer vision (CV) techniques based on deep learning have significantly advanced wildfire monitoring and response. Convolutional Neural Networks (CNNs) [7] have been widely adopted for fire and smoke detection, classification, and segmentation tasks. Among these, segmentation-based approaches enable the precise delineation of fire regions, enhancing monitoring accuracy and facilitating rapid intervention [8].

Object detection models such as YOLO and SSD have been successfully applied to fire detection under complex environmental conditions. YOLO-based variants provide real-time inference capabilities, making them particularly suitable for UAV-based applications [9], while region-based CNNs offer robust performance across diverse datasets [10]. These deep learning models have demonstrated high accuracy but still rely heavily on large annotated datasets and lack generalization in unseen or dynamically evolving fire scenarios.

Despite recent progress, fire detection models based on supervised learning remain limited by their reliance on large labeled datasets and their poor generalization to unseen environments. Curating annotated fire data is costly, time-consuming, and often impractical in dynamic real-world settings. Furthermore, models trained on specific fire types or locations frequently fail under novel conditions, reducing their reliability in critical deployments [11].

To address these limitations, we propose a zero-shot, prompt-aligned fire and smoke classification framework optimized for edge devices and IoT data networking. Inspired by the CLIP architecture [11], the method learns cross-modal representations by aligning lightweight CNN-based visual embeddings with semantic textual prompts through contrastive learning. This design enables inference on unseen fire scenarios without task-specific fine-tuning, offering high scalability and efficiency for embedded wildfire monitoring applications.

The proposed framework aligns visual embeddings with natural language prompts using a contrastive loss objective, enabling cross-modal retrieval and zero-shot classification. During inference, the model requires only textual descriptions such as “a fire scene in picture” or “a smoke scene in picture” to identify fire-related scenarios without any task-specific fine-tuning.

The key contributions of this work are as follows:

- We design and implement a lightweight, prompt-aligned dual-encoder framework for zero-shot fire and smoke classification, optimized for embedded edge devices.
- Beyond vision accuracy, SpectroFire is optimized for real-time operation in IoT data networking. By transmitting only event-level outputs instead of raw video streams, our design significantly reduces communication overhead while preserving low-latency response in distributed UAV–IoT systems.
- We apply LoRA-based parameter-efficient fine-tuning, reducing training overhead while preserving model of size 0.46 MB, and demonstrate real-time feasibility by achieving 2339 FPS inference on a RPi 5.
- We validate SpectroFire using the Kaggle fire and smoke dataset and further evaluate its end-to-end performance in network-aware settings, confirming its scalability and robustness for collaborative UAV–IoT wildfire monitoring in distributed environments.

The remainder of the paper is structured as follows. In Section II, we review prior work on sensor-based and vision-based wildfire detection methods. Section III describes the proposed zero-shot classification framework in detail. Section IV presents experimental results validating the model’s effectiveness. Section V discusses current limitations and potential directions for future research. Finally, the conclusion is presented in Section VI.

II. Related Works

Sensor-based and vision-based fire and smoke detection and classification methods are used to classify fire and smoke in a real-time environment.

Sensor Based Methods: Ground-based early detection systems utilize either standalone sensors, such as fixed, Pan-Tilt-Zoom (PTZ), or 360° cameras, or interconnected networks of terrestrial sensors. The strategic placement of these devices is essential to guarantee sufficient coverage and visibility. Therefore, these sensors are commonly installed on watchtowers, elevated structures positioned at strategically high points to monitor areas prone to fire risk. They serve not only for early detection but also for verifying and pinpointing the locations of reported fire incidents. Early fire detection typically relies on two types of cameras: optical and IR cameras; both are capable of capturing imagery from low to ultra-high resolution, depending on the specific detection scenario [12]. While optical cameras capture the color details of a scene, IR sensors detect and measure the thermal radiation emitted by objects, offering complementary information for fire detection [13]. In recent developments, early detection systems have been introduced that integrate both optical and IR cameras to enhance detection accuracy and reliability. Computer-based approaches are capable of handling large volumes of data with the goal of maintaining consistent accuracy while minimizing false alarms. In the sections that follow, we first explore traditional techniques relying on handcrafted features and then discuss more recent DL methods that enable automated feature extraction. Detection techniques utilizing optical sensors or RGB cameras extract features linked to the physical characteristics of flames and smoke, including color, motion, spatial and temporal patterns, and texture attributes. Various color spaces have been employed for early fire detection, including RGB, HSV, CIELAB, YCbCr, CIELAB, and YUV. However, a major limitation of color-based fire detection methods is their likelihood of producing high false alarm rates, as relying solely on color information is often inadequate for achieving early and reliable fire detection [14]. Zhang et al. [15] proposed a probabilistic model based on color features to identify potential fire regions and further incorporated motion characteristics to determine the final presence of fire. Avgerinakis et al. [16] first localized potential smoke regions by identifying candidate blocks, then extracted Histogram of Oriented Gradients (HOG) and Histogram of Optical Flow (HOF) features to jointly capture appearance and motion cues. Similarly, Mueller et al. [17] adopted a dual-strategy approach incorporating both optimal mass transport formulations and data-driven optical flow models for motion dynamics.

Vision-based DL methods: Over the past decade, AI and DL techniques have significantly advanced a wide range of CV tasks, including object detection and segmentation, satellite image analysis, medical diagnosis, autonomous driving and road monitoring. This progress is largely attributed to the rich feature representations learned by convolutional layers, which enable DL models to perform pixel-wise classification and accurately capture object appearance in segmentation tasks. These algorithms have demonstrated greater reliability and performance compared to traditional ML models [18]. In recent years, extensive research has focused on employing DL methods for effective fire detection. Muhammad et al. [19] introduced a novel, energy-efficient, and computationally lightweight CNN architecture for fire detection, localization, and semantic understanding of fire appearance, inspired by the SqueezeNet framework and tailored for deployment in CCTV surveillance systems. Bochkov et al. [20] proposed UUNet, a novel concatenative DL architecture that integrates binary and multiclass U-Net models. This design enables color-based multiclass segmentation of signals derived from the binary segmentation of single-nature objects, such as fire regions. Additionally, they introduced a custom fire-image dataset consisting of 6,250 samples of 224 × 224 resolution. Experimental results demonstrated that UUNet outperformed the original U-Net by 3% in multiclass segmentation and 2% in binary segmentation tasks. Akhloufi et al. [21] integrated an encoder–decoder architecture with their Deep-Fire model, achieving an F-measure score of 97.09% on the training set and 91% on the test set. The model was trained on a limited dataset of 419 images from the Corsican wildfire dataset, using Dice loss as the optimization objective. Huang et al. [22] proposed the Wavelet-CNN framework, which integrates CNNs with wavelet-based analysis. In this approach, multiple features are incorporated into multiple CNN layers, enhancing representation learning and detection performance. The method demonstrated improved results when applied to backbone architectures such as MobileNetV2 and ResNet50. Hassan et al. [23] applied transfer learning on SqueezeNet for fire detection and classification. The model achieved an accuracy of 95% on the benchmark fire dataset. Barmpoutis et al. [24] employed the Faster R-CNN framework to localize potential fire regions, incorporating multidimensional texture analysis to improve detection accuracy. Despite its effectiveness, the approach results in increased computational demands due to the incorporation of high-dimensional texture feature analysis. Talaat et al. [25] employed YOLOv8 for fire detection in smart cities. They achieved a precision of 97.1% for forest fire detection. DL models like CNNs, ResNets, and Yolo variants have significantly improved detection accuracy but require large-scale annotated datasets, failing under unseen environmental conditions. In contrast, our proposed CLIP-like zero-shot learning method offers a robust, annotation-free alternative specifically adapted for RGB fire and smoke classification.

III. Method

This section first defines the fire and smoke classification task, followed by a description of the proposed CLIP-style zero-shot classification framework. The overall architecture of SpectroFire is presented in Fig. 1.

A. Model Framework

The proposed model, SpectroFire is inspired by CLIP but tailored for lightweight, edge-oriented deployment. It consists of two key components: a visual encoder and a text encoder, both projecting input into a shared embedding space. The overall framework is designed to be lightweight and efficient for real-time deployment. Unlike large vision-language models that have hundreds of millions of parameters, SpectroFire emphasizes a compact architecture to reduce computational cost and memory footprint, making it feasible for use on edge devices such as surveillance drones or IoT cameras. Both the image and text representations are D- dimensional vectors in a common embedding space, so that their compatibility can be measured directly via similarity. During training, the two encoders learn jointly such that matching image-text pairs provide closely aligned embeddings while non-matching pairs are pushed far apart. We detail each component of the framework below, followed by the training objective used for cross-modal alignment.

1) Visual Encoder The visual encoder $f(\cdot)$ is a lightweight four-block CNN (Conv + ReLU per block) that uses strided convolutions, not max pooling, to downsample efficiently. After standard resizing and normalization, features are aggregated with adaptive average pooling and projected to the shared embedding space $\mathbb{R}^D$. The projection is a LoRA-injected linear layer, which enables low-rank, parameter-efficient fine-tuning with minimal overhead and is suitable for edge devices. The output $f(I)$ is L_2-normalized, so dot products become cosine similarities. This design captures global semantics and fine fire or smoke cues while maintaining high throughput for zero-shot classification via a contrastive similarity head.

2) Text Encoder The text encoder $g(\cdot)$ maps a prompt T to a D-dimensional embedding shared with the image space. We use a reduced-size Transformer (fewer layers/heads): tokens are embedded, passed through Transformer blocks with LayerNorm, and the mean representation is linearly projected to $\mathbb{R}^D$. The output is L_2-normalized, enabling cosine-similarity comparison with image embeddings. This compact design keeps computation and memory low while preserving semantic discrimination between concepts such as ”fire” and ”smoke”.

B. Cross-Modal Alignment and Similarity

We compute cosine similarity between L_2-normalized image and text embeddings to measure semantic correspondence. A learnable temperature τ controls the sharpness of the scores. For a batch of pairs $\{(I_i, T_i)\}_{i=1}^N$, let

$$s_{ij} = \frac{f(I_i)^\top g(T_j)}{\tau}.$$

Training uses a symmetric contrastive objective (InfoNCE):

$$\mathcal{L} = \tfrac{1}{2}\Big[\frac{1}{N}\sum_{i} - \log\frac{e^{s_{ii}}}{\sum_j e^{s_{ij}}} + \frac{1}{N}\sum_{j} - \log\frac{e^{s_{jj}}}{\sum_i e^{s_{ij}}}\Big],$$

which pulls matched image–text pairs together and pushes mismatches apart. This alignment yields a shared embedding space where prompts such as “fire” or “smoke” act as prototypes, enabling zero-shot recognition by selecting the class whose prompt has the highest cosine similarity.

C. Prompt Engineering

Since the text encoder's output can be sensitive to the phrasing of the input text, we design natural language prompts, such as ”a fire scene in picture” for fire scenarios and ”a smoke scene in picture” for smoke to encode prior knowledge about fire and smoke scenarios. These prompts serve as semantic anchors in the cross-modal space, allowing the model to associate learned patterns with high-level contextual meanings. During inference, the model computes cosine similarity between encoded features and textual prompts, facilitating zero-shot fire scene classification without explicit retraining. By covering fundamental fire and smoke conditions, prompt-based supervision effectively generalizes the model's decision boundary across diverse unseen scenarios, enabling robust wildfire detection even in variable environmental settings.

IV. Experiments

In this section, experiments are conducted to demonstrate the effectiveness of the proposed SpectroFire model. We compare our model with the CLIP baseline and MobileNetV2, a lightweight supervised classification method. The performance of all the models is evaluated on Kaggle fire and smoke dataset.

A. Dataset

The dataset consists of 23,730 images of fire and smoke. The dataset includes different fire and smoke scenarios such as garbage burning, paper and plastic burning, agricultural crop burning, and home cooking, covering a comprehensive range of fire scenarios. The dataset contains a variety of fire and smoke scenarios of different sizes. According to the fire stages, the early, middle, and late stages of fire are included. According to the fire environment, different backgrounds are included, such as fog and nighttime. According to the shooting angle, fire images taken from different angles are included, such as distant, close and above. Additionally, the dataset also covers fire images with other interference factors, such as light, clouds, and steam. The dataset samples are shown in Fig. 2 and dataset statistics are presented in Table. I

TABLE I
Statistics of the Kaggle fire and smoke dataset

Split	RGB Fire	RGB Smoke
Train	9661	8082
Valid	2415	1020

B. Experimental Setup

Experiments were conducted on both high-performance and edge environments. For training and large-scale evaluation, we used a GPU server equipped with a single NVIDIA H100 GPU (80 GB of memory). For embedded deployment benchmarking, we employed a Raspberry Pi (RPi) 5 with a quad-core ARM Cortex-A76 CPU and 4 GB LPDDR4 RAM. The software environment included Python 3.12, PyTorch 1.12, and TensorFlow 2.8.0. Model training utilized the Adam optimizer with an initial learning rate of 0.0001 and parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$. All models were trained for 100 epochs with binary cross-entropy (BCE) loss. Input images were resized to 224×224 pixels, normalized using ImageNet statistics, and trained with a batch size of 64. For edge deployment, static quantization to 8-bit precision was applied using TensorFlow Lite. In addition to standalone inference benchmarking, we evaluated network-aware deployment conditions to emulate real-world IoT scenarios. RPi 5 nodes transmitted detection results to a central server over Wi-Fi and 4G links. Instead of streaming raw video, only low-dimensional classification embeddings and event-level alerts were transmitted, which reduced bandwidth. End-to-end latency, including both on-device inference and IoT data networking, was measured. This experimental setup highlights that SpectroFire is optimized not only for high-throughput local inference but also for efficient operation within distributed edge–cloud networks, an essential requirement for UAV-based wildfire monitoring and IoT-driven disaster management systems.

C. Evaluation Metrics

Evaluating a model is crucial for improving its efficiency. Various metrics can be used to assess model performance. In this study, we used the F1 score [26] as the primary evaluation metric to measure and enhance the model's effectiveness.

D. Experimental Analysis

Table II presents a comparative evaluation of three fire detection and classification models on the Kaggle dataset. We compare our proposed method, SpectroFire, against a baseline CLIP model and a fully supervised MobileNetV2 classifier. The CLIP baseline, while offering zero-shot flexibility, shows limited performance with only 67% accuracy and an inference speed of 174 FPS. This result highlights the limitations of unmodified vision-language models in fine-grained anomaly detection tasks such as fire classification, without additional adaptation. The MobileNetV2 model, trained in a fully supervised manner, achieves a high accuracy of 94%, establishing a strong upper-bound for lightweight supervised architectures. However, its inference speed reaches only 226 FPS, making it less suitable for real-time, high-throughput embedded systems. Our proposed model, SpectroFire, delivers a compelling trade-off by achieving 90% accuracy which is only 4% lower than MobileNetv2 while providing a tenfold improvement in inference speed, reaching 2239 FPS with an average inference time of just 0.45 ms per frame. This substantial throughput advantage is achieved without

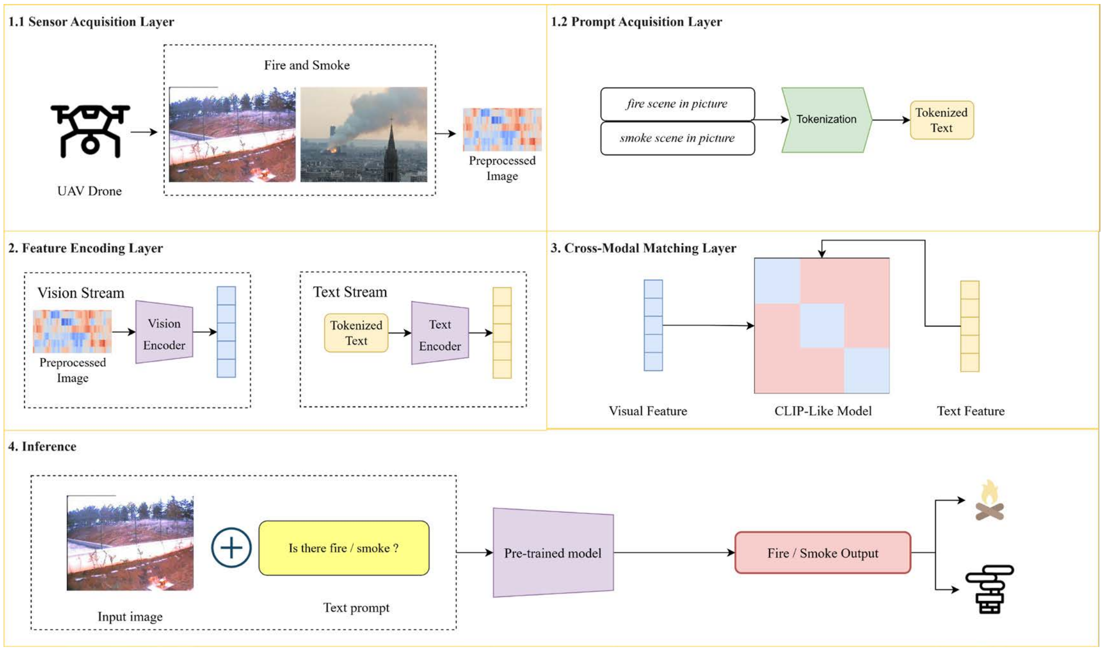

Fig. 1. SpectroFire: Overview of the proposed fire and smoke classification framework

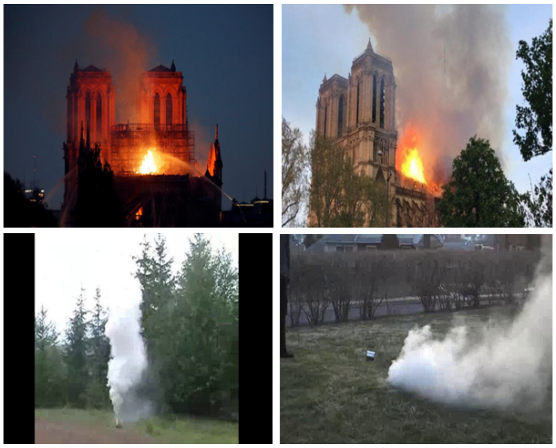

Fig. 2. Sample Images from Kaggle fire and smoke dataset. In the first row we include the RGB fire images and in second row we presented RGB smoke images

requiring supervised fine-tuning, making the system highly suitable for low-latency deployment on resource-constrained devices, such as the RPi 5. Additionally, SpectroFire benefits from LoRA-based parameter-efficient tuning and quantization-aware optimization, both of which contribute to maintaining high performance while reducing computational and memory overhead. The model's prompt-driven generalization capability further enables adaptation to unseen environments and camera domains, eliminating the need for retraining.

TABLE II
PERFORMANCE COMPARISON OF CLASSIFICATION MODELS ON KAGGLE FIRE AND SMOKE DATASET

Methods	Accuracy	FPS	Inference Time (ms)
CLIP (Baseline)	67	174	5.75
MobileNetv2 (Supervised)	94	226	4.41
SpectroFire (Ours)	**90**	**2239**	**0.45**

E. Evaluation of Quantized Model

We evaluate the performance of an 8-bit quantized version of our model on a RPi 5 platform, targeting the deployment of efficient fire detection models in resource-constrained edge environments. The original model size was 0.42 MB, which was reduced to 0.36 MB after applying 8-bit static quantization, achieving a 10.6% reduction in memory footprint. This quantization process was carefully selected to minimize storage and memory usage while maintaining high classification performance, critical for real-time embedded applications operating under strict resource budgets.

To assess real-world feasibility, we conducted single-image inference tests on the RPi 5, equipped with a quad-core ARM Cortex-A76 CPU and 4 GB of LPDDR4 RAM. The quantized SpectroFire model achieved an average inference time of 0.45 milliseconds per image, corresponding to approximately 2239 FPS. These results confirm that the quantized model meets the real-time processing threshold necessary for UAV-based wildfire monitoring, where rapid scene analysis is crucial for early fire detection and timely intervention.

V. LIMITATIONS AND FUTURE WORK

While SpectroFire demonstrates strong performance in zero-shot fire classification, it also has several limitations that open opportunities for future research. First, the current framework supports only binary classification (fire vs. smoke) and does not differentiate between fire stages or surrounding context (e.g., smoke, vegetation type). Extending the model to handle multi-class or multi-label recognition scenarios remains an important direction. Second, although the system generalizes well to unseen scenes, its performance under extreme occlusion (e.g., heavy smoke, dense foliage) and low-resolution thermal imagery has not been fully assessed. Third, SpectroFire relies on handcrafted prompt templates during inference. Prompt engineering introduces sensitivity to phrase choice, which could be mitigated by using recent prompt-tuning or vision-language pretraining strategies. Additionally, the current model operates on single-frame image input. Temporal modeling using video streams or frame history could improve detection stability and enable early prediction of fire spread. The model may face difficulty for fire and smoke detection when UAV is flying over hight altitudes. To address this limitation, we plan to collect data from varying heights. From a deployment perspective, UAV-based operation may face real-world constraints such as limited bandwidth, frame drops, and environmental noise. Future work will therefore focus on integrating fault-tolerant communication protocols and video-based aggregation mechanisms to improve robustness in distributed, large-scale wildfire surveillance networks. Furthermore, we plan to conduct real-world UAV deployment experiments, embedding SpectroFire directly on drones to validate system performance under realistic conditions. This will provide critical insights into operational feasibility, scalability, and resilience for practical wildfire monitoring scenarios.

VI. Conclusion

This paper presents SpectroFire, a lightweight CLIP-style dual encoder for zero-shot fire and smoke detection. By combining a custom CNN-based visual encoder with contrastive alignment to textual prompts, the model eliminates the need for task-specific supervised retraining while retaining strong performance. Experiments showed that SpectroFire achieves 90% accuracy on the Kaggle fire and smoke dataset and a real-time throughput of over 2200 FPS on RPi 5 devices. Beyond standalone performance, the model was also evaluated under network-aware deployment scenarios, where transmitting only event-level outputs instead of raw video reduced bandwidth consumption. This confirms the model's feasibility for collaborative UAV and IoT-based wildfire monitoring in IoT data networking.

References

[1] Statista, "Number of forest fires in south korea," 2023, accessed: February 25, 2025. [Online]. Available: https://www.statista.com/statistics/1296399/south-korea-forest-fire-outbreaks/

[2] Associated Press, "California governor asks congress for nearly $40 billion for los angeles wildfire relief," 2025, accessed: February 25, 2025. [Online]. Available: https://apnews.com/article/71ec591a60c05d45432382095dbfd147

[3] L. Ramos, E. Casas, E. Bendek, C. Romero, and F. Rivas-Echeverría, "Computer vision for wildfire detection: a critical brief review," *Multimedia Tools and Applications*, vol. 83, no. 35, pp. 83 427–83 470, 2024.

[4] X. Wang, H. Zhou, W. P. Arnott, M. E. Meyer, S. Taylor, H. Firouzkouhi, H. Moosmüller, J. C. Chow, and J. G. Watson, "Evaluation of gas and particle sensors for detecting spacecraft-relevant fire emissions," *Fire Safety Journal*, vol. 113, p. 102977, 2020.

[5] S. G. Xu, S. Kong, and Z. Asgharzadeh, "Wildfire detection using streaming satellite imagery," *Proceedings of the 2021 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 2899–2902, 2021.

[6] C. Ding, X. Zhang, J. Chen, S. Ma, Y. Lu, and W. Han, "Wildfire detection through deep learning based on himawari-8 satellites platform," *International Journal of Remote Sensing*, vol. 43, no. 13, pp. 5040–5058, 2022.

[7] K. O'shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[8] V. Khryashchev and R. Larionov, "Wildfire segmentation on satellite images using deep learning," *2020 Moscow Workshop on Electronic and Networking Technologies (MWENT)*, pp. 1–5, 2020.

[9] A. Abdusalomov, N. Baratov, A. Kutlimuratov, and T. K. Whangbo, "An improvement of the fire detection and classification method using yolov3 for surveillance systems," *Sensors*, vol. 21, no. 19, p. 6519, 2021.

[10] A. Nguyen, H. Nguyen, V. Tran, H. X. Pham, and J. Pestana, "A visual real-time fire detection using single shot multibox detector for uav-based fire surveillance," *2020 IEEE Eighth International Conference on Communications and Electronics (ICCE)*, pp. 338–343, 2021.

[11] A. Radford, J. W. Kim *et al.*, "Learning transferable visual models from natural language supervision," *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

[12] A. E. Çetin, K. Dimitropoulos, B. Gouverneur, N. Grammalidis, O. Günay, Y. H. Habiboğlu, B. U. Töreyin, and S. Verstockt, "Video fire detection–review," *Digital Signal Processing*, vol. 23, no. 6, pp. 1827–1843, 2013.

[13] B. U. Töreyin, R. G. Cinbiş, Y. Dedeoğlu, and A. E. Çetin, "Fire detection in infrared video using wavelet analysis," *Optical Engineering*, vol. 46, no. 6, pp. 067 204–067 204, 2007.

[14] P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, and N. Grammalidis, "A review on early forest fire detection systems using optical remote sensing," *Sensors*, vol. 20, no. 22, p. 6442, 2020.

[15] Z. Zhang, T. Shen, and J. Zou, "An improved probabilistic approach for fire detection in videos," *Fire Technology*, vol. 50, pp. 745–752, 2014.

[16] K. Avgerinakis, A. Briassouli, and I. Kompatsiaris, "Smoke detection using temporal hoghof descriptors and energy colour statistics from video," in *International workshop on multi-sensor systems and networks for fire detection and management*, 2012.

[17] M. Mueller, P. Karasev, I. Kolesov, and A. Tannenbaum, "Optical flow estimation for flame detection in videos," *IEEE Transactions on image processing*, vol. 22, no. 7, pp. 2786–2797, 2013.

[18] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 7, pp. 3523–3542, 2021.

[19] K. Muhammad, J. Ahmad, Z. Lv, P. Bellavista, P. Yang, and S. W. Baik, "Efficient deep cnn-based fire detection and localization in video surveillance applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 7, pp. 1419–1434, 2018.

[20] V. S. Bochkov and L. Y. Kataeva, "Wuunet: Advanced fully convolutional neural network for multiclass fire segmentation," *Symmetry*, vol. 13, no. 1, p. 98, 2021.

[21] M. A. Akhloufi, R. B. Tokime, and H. Elassady, "Wildland fires detection and segmentation using deep learning," in *Pattern recognition and tracking xxix*, vol. 10649. SPIE, 2018, pp. 86–97.

[22] L. Huang, G. Liu, Y. Wang, H. Yuan, and T. Chen, "Fire detection in video surveillances using convolutional neural networks and wavelet transform," *Engineering Applications of Artificial Intelligence*, vol. 110, p. 104737, 2022.

[23] A. Hassan and A. I. Audu, "A lightweight cnn model for vision based fire detection on embedded systems," *FUOYE Journal of Engineering and Technology*, vol. 9, no. 4, pp. 624–628, 2024.

[24] P. Barmpoutis, K. Dimitropoulos, K. Kaza, and N. Grammalidis, "Fire detection from images using faster r-cnn and multidimensional texture analysis," *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8301–8305, 2019.

[25] F. M. Talaat and H. ZainEldin, "An improved fire detection approach based on yolo-v8 for smart cities," *Neural Computing and Applications*, vol. 35, no. 28, pp. 20 939–20 954, 2023.

[26] S. A. Hicks, I. Strümke, V. Thambawita, M. Hammou, M. A. Riegler, P. Halvorsen, and S. Parasa, "On evaluation metrics for medical applications of artificial intelligence," *Scientific reports*, vol. 12, no. 1, p. 5979, 2022.