An Effective Forest Fire Detection Framework Using Heterogeneous Wireless Multimedia Sensor Networks

BURAK KIZILKAYA, Computer Engineering, Middle East Technical University Northern Cyprus Campus, 99738, Mersin 10, Turkey and School of Engineering, University of Glasgow, Glasgow, UK
ENVER EVER, Computer Engineering, Middle East Technical University Northern Cyprus Campus, 99738, Mersin 10, Turkey
HAKAN YEKTA YATBAZ, Computer Engineering, Middle East Technical University Northern Cyprus Campus, 99738, Mersin 10, Turkey
ADNAN YAZICI, Dept. of Computer Science, SEDS, Nazarbayev University, Kazakhstan, and Dept of Computer Engineering, Middle East Technical University, Ankara, Turkey

With improvements in the area of Internet of Things (IoT), surveillance systems have recently become more accessible. At the same time, optimizing the energy requirements of smart sensors, especially for data transmission, has always been very important and the energy efficiency of IoT systems has been the subject of numerous studies. For environmental monitoring scenarios, it is possible to extract more accurate information using smart multimedia sensors. However, multimedia data transmission is an expensive operation. In this study, a novel hierarchical approach is presented for the detection of forest fires. The proposed framework introduces a new approach in which multimedia and scalar sensors are used hierarchically to minimize the transmission of visual data. A lightweight deep learning model is also developed for devices at the edge of the network in order to improve detection accuracy and reduce the traffic between the edge devices and the sink. The framework is evaluated using a real testbed, network simulations and 10-fold cross-validation in terms of energy efficiency and detection accuracy. Based on the results of our experiments, the validation accuracy of the proposed system is 98.28%, and the energy saving is 29.94%. The proposed deep learning model's validation accuracy is very close to the accuracy of the best performing architectures when the existing studies and lightweight architectures are considered. In terms of suitability for edge computing, the proposed approach is superior to the existing ones with reduced computational requirements and model size.

Additional Key Words and Phrases: IoT, WMSNs, deep learning, edge computing, energy efficiency, heterogeneous WMSN architecture

1 INTRODUCTION

IoT infrastructures, smart sensors, and wireless sensor networks (WSNs) are some of the most promising technologies of our time, especially for fully autonomous systems. Traditional WSNs are generally used to transmit scalar data such as temperature, humidity, and light related information. The size of scalar data is usually much smaller than that of multimedia data. However, the accuracy of systems based solely on scalar data may not be sufficient in some critical applications which require real-time monitoring and control. Environmental monitoring for disaster management, such as the detection of forest fire and floods, and medical applications, such as wireless patient

Authors' addresses: Burak Kızılkaya, b.kizilkaya.1@research.gla.ac.uk, Computer Engineering, Middle East Technical University Northern Cyprus Campus, 99738, Mersin 10, Turkey and School of Engineering, University of Glasgow, Glasgow, UK ; Enver Ever, eever@metu.edu.tr, Computer Engineering, Middle East Technical University Northern Cyprus Campus, Guzelyurt, 99738, Mersin 10, Turkey; Hakan Yekta Yatbaz, hakan.yatbaz@metu.edu.tr, Computer Engineering, Middle East Technical University Northern Cyprus Campus, Guzelyurt, 99738, Mersin 10, Turkey; Adnan Yazıcı, adnan.yazici@nu.edu.kz,yazici@ceng.metu.edu.tr, Dept. of Computer Science, SEDS, Nazarbayev University, Nur-Sultan, Kazakhstan, and Dept of Computer Engineering, Middle East Technical University, Ankara, Turkey.

monitoring, are two popular examples of applications in which multimedia sensors are commonly used with various scalar sensors.

Wireless Multimedia Sensor Networks (WMSNs) have access to multimedia sensors. They are used effectively with new image processing and machine learning (ML) techniques in various types of applications, including surveillance systems, traffic and transport, smart healthcare delivery, surveillance of environment and localization systems [19], [37], [45], [63]. One of the main constraints of WMSNs is energy consumption since the transmission of multimedia data is expensive. WMSN nodes are limited not only in terms of battery, but also in terms of memory and processing power. It is therefore crucial to use the available resources as efficiently as possible.

Recent studies on environmental disasters, including forest fires, show that the autonomous systems have great potential for improvement [19]. The forest fires in Athens, where at least 90 people died, 164 adults and 23 children were injured, the California wildfires in which 185,800 hectares of forest were burned in 2018 [59], and the very recent fires in Australia, where around 12 million acres were burned [50], are events that early and accurate forest fire detection systems could have a significant impact on these fires.

The main objective of this study is to present an accurate and energy-efficient framework for emergency applications combining energy-efficient scalar sensors, multimedia sensors and advanced deep learning (DL) algorithms. Since forest conditions are not suitable for continuous maintenance, self-configurable, inexpensive, accurate, easy-to-deploy and energy efficient systems are required.

In order to achieve energy efficiency, we present a hierarchical framework in this paper. Two different types of sensors are considered for the hierarchy introduced:

- Scalar sensors: These sensors have limited computing, memory and sensing capabilities.
- Multimedia sensors: These sensors are equipped with multimedia sensors and they have significantly better computing, and memory capacities. This is why they are also referred to as smart sensors.

Scalar sensors are used at the first level of detection. Measurements from scalar sensors are thus used to trigger multimedia sensors for more accurate detection of fires. In addition, deep learning algorithms are used in nodes with multimedia sensors (also called smart sensors). In this sense, it is possible to say that edge computing is employed; however, unlike many of the existing studies, classification is performed on end nodes instead of a dedicated sink node close to the deployment area. To be able to deploy and run the deep learning algorithms for detection at the end devices, the model size and the computational requirements of the architecture employed are reduced. Although in terms of computational requirements and model size, the proposed model is superior, the accuracy of the proposed architecture is better than most of the existing studies and quite close to the architectures with highest detection accuracy. The edge computing is employed to prevent unnecessary data transmission, which increases the efficiency of the system because data transmission is the most energy-intensive activity. By reducing the transmission rate, the traffic load of the wireless channel is also reduced. This results in an increase in the packet reception rate since the probability of collision decreases. For the accuracy of the system, a lightweight DL model is proposed and tested. The proposed framework is implemented using a real testbed. The results obtained from the testbed are used in conjunction with OMNET++ [62], [48] based results to verify the energy efficiency and quality of service (QoS) measures. To train and test the DL model, in particular, a Convolutional Neural Networks (CNN) model, a new image dataset[1] is defined for forest fire applications. According to the results of the experiments, an accuracy of about 98% is

[1]The new image dataset established for forest fire detection is publicly available at https://doi.org/10.17632/g5nzp6j3bt.1

obtained and validated to adjust the parameters of a classifier. The main contributions of this study are summarized below.

- A novel, energy-efficient and accurate heterogeneous wireless multimedia sensor network framework is proposed for real-time surveillance applications.
- Although there are solutions based on Machine Learning (ML) in the literature, these approaches require significant computing resources and memory space. We propose a significantly more lightweight CNN model in this study for classification.
- To the best of our knowledge, this is the first study in which the CNN works in the smart sensors at the edge of the network for the detection of forest fires.
- A real-time testbed implementation is presented, and simulations are used for verification and extrapolation purposes.
- Our approach is as accurate as other competitive detection mechanisms and more efficient in terms of energy consumption, as well as computational and memory requirements.

The rest of the paper is organized as follows. Existing studies are discussed in Section 2. Section 3 explains the proposed forest fire detection framework. Section 4 presents our approach. The evaluation of our proposed system is presented in Section 5. Finally, Section 6 concludes the paper.

2 RELATED WORK

With efficient communication infrastructures and support of network protocols at various levels, WSN and WMSN have been widely integrated with multi-modal sensors and actuators in order to actively detect any abnormal event in terms of safety, surveillance and security.

2.1 Forest Fire Detection

Traditionally, forest fire detection has been conducted using human observations from watchtowers located in high-altitude areas. However, the working conditions of the surveillance staff are poor and the accuracy of this information is open to discussion [61]. Other previous implementations include video surveillance systems, and use of satellites which suffer from problems such as accuracy, delay, and energy efficiency.

Given the constraints of traditional methods, one of the most promising technologies for fire detection is wireless sensor networks with IoT-enabled applications [47]. In [14], a new architecture is presented for public protection and disaster relief networks. The architecture is based on cognitive communication and uses WSNs to establish what is called as "Low-Tier Application Network". As in our study, the simulation results are presented to demonstrate the effectiveness of the proposed architecture, considering a disaster scenario in which emergency situations such as a fire or bombing occurs in the center of an area. The results of the simulation presented are mainly related to QoS measures, such as cognitive network throughput. Another study of disaster scenarios is presented in [17]. A framework is presented for an industrial field based on the industrial WSN (IWSN). Fire emergencies in general and forest fires are presented as application areas of the new framework. The importance of energy efficiency is emphasized throughout the study. The proposed framework is not evaluated in terms of energy efficiency or QoS.

In [19], the fire sensing technologies and differences in the development of hardware as well as algorithms are considered critically. A modified fire sensing and control system concept is also proposed. The proposed framework makes use of an array of sensors, however, multimedia related features which could have improved the detection accuracy significantly are not considered. Furthermore, the proposed framework is at the stage of development and the quantitative results of evaluation are not presented. Similarly, in [18] WSNs are employed for disaster scenarios such as severe earthquakes, collapsing buildings and fires. The message synchronization capability of

WSNs is used for new opportunistic network infrastructures to achieve low delivery latency and high delivery rate.

In a recent study [30], a framework is presented for forest fire surveillance using WSNs, as IoT devices together with, fog and cloud computing paradigms. Instead of multimedia data, datasets of various meteorological factors such as temperature, relative humidity, precipitation, wind speed, and spatial coordinates of a potential fire site are used together with machine learning algorithms. The collected information is used more proactively to specify the susceptibility classes of an observation area (very-low, low, moderate, high and very-high) instead of real-time high accuracy detection through multimedia data. Artificial Neural Network (ANN), K-Nearest Neighbour (KNN), Decision Tree and Support Vector Machine (SVM) techniques are employed for classification. ANN outperformed the other classification techniques with an accuracy of 95.45%. The lifetime of the framework is presented in terms of battery levels for an observation of up to 300-400 minutes.

When reviewing existing work on forest fire detection, some studies fail to address energy efficiency since appropriate assessment methods are not employed. On the other hand, although some studies use simulation tools to show the energy efficiency of the proposed architectures, they do not consider the multimedia data with the assessment of the detection accuracy. Furthermore, systems that use machine learning or deep learning models do not perform classification at the edge of the network.

2.2 Multimedia Sensor Networks

Studies in the environmental monitoring application literature including forest fire detection, can be categorized as scalar sensor-based systems, multimedia sensor systems, and systems using both technologies as shown in Table 1.

Table 1. Literature Categorization

Study	Sensors Used	Evaluation
[32]	Multimedia	Energy
[68]	Multimedia	Accuracy
[10]	Multimedia	Energy & Accuracy
[16],[35],[5],[28],[51]	Scalar	Only System Design
[49],[4],[67],[41]	Scalar	Energy
[42]	Scalar	Review
[25]	Scalar	Security
[33]	Multimedia & Scalar	Energy & Accuracy

A study of a system in which Raspberry Pi and Arduino boards are used for environmental monitoring applications is proposed in [16]. Scalar sensor readings such as temperature and humidity are used to monitor the environment. A systematic review is presented for the wireless sensor network applications used to monitor coal mines in [42]. WSN-based forest fire monitoring is used to test the algorithms for detecting anomalies in [25]. The proposed method estimates the maximum number of malicious acts tolerated by the application. The design of the deployment of the sensors is automated using the proposed method. Similar studies are presented in [51] where in-depth analysis and validation real-time scheduling is performed in resource constrained hardware platforms such as Raspberry Pi and Arduino boards, and in [35], where body temperature, respiratory rate, heart rate and body movements are monitored using Raspberry Pi cards. A novel computing architecture is introduced [51], while a detailed system design is proposed in [35]. The study presented in [35] lacks the analysis of accuracy and energy efficiency. A traffic monitoring system is offered in [32]. A Raspberry Pi card and an HD camera are used for object detection and the system is analyzed in terms of energy consumption.

Classification of moving objects, real-time detection, recognition and understanding of activity from multiple sensors are considered in studies such as [55] and [10]. While a novel fusion framework is presented in [55] to combine data from multiple video sensors, a system architecture with scalar and multimedia sensors is presented in [10] for WMSNs to facilitate classification of moving objects. For both studies, experimental setups are implemented, and tests are performed to show the accuracy of the frameworks. Multimedia and scalar sensors are used together in [33] as well for the object detection and classification with visual and auditory data fusion. A practical implementation of a system is presented, and results of performance and energy consumption are discussed. Multimedia data, particularly in the form of surveillance video, is extensively used for event recognition. In most of the studies using surveillance video, while prioritising the accuracy of detection, the underlying frameworks become quite complicated. For example, in [38], a Spatio-temporal deep residual network with hierarchical attentions is employed for video event recognition. Residual Neural Network (ResNet), Faster Region-Based Convolutional Neural Networks (R-CNN), and Long Short Term Memory (LSTM) are used with a multi-layer architecture to improve the accuracy by 5.6% compared to similar existing architectures.

The literature presents numerous examples of monitoring and/or surveillance applications using multimedia sensors. Some studies focus on the accuracy of surveillance information. Some other studies consider energy efficiency alongside accuracy. However, none of these studies propose an autonomous system for forest fires. To the best of our knowledge, our study presented here is the first to attempt to combine the two approaches and evaluate their effectiveness and accuracy using simulations as well as real-world testbed for wildfire surveillance applications.

Recently, unmanned aerial vehicles (UAV) have become popular as an enabling technology of certain environment monitoring applications. In [2], a UAV based forest fire detection system is proposed. The battery life of the UAV is set as 18 minutes, which means that the system can work in an emergency for alert, instead of detecting the emergency. Similarly, in [56], an approach based on fog computing and deep learning with drones is proposed. A deep convolutional network architecture is employed which achieves up to 95% accuracy for fire detection. In [8], a CNN-based forest fire detection system is presented where images of the forest are collected with a drone. The approach presented tests different CNN architectures, reaching up to 86% accuracy.

UAV-based environment detection systems have their advantages, however there are many limitations to using UAVs both proactively and reactively for forest fire detection. Firstly, due to its limited battery, UAVs can patrol the given area at specific intervals and cannot provide continuous monitoring. In that sense, if a proposed UAV-based system employs a large number of ready-to-use batteries for certain applications, this will significantly affect the cost of the entire system. In addition, a UAV has limitations on the payload to maintain a safe and proper flight, and has less adaptability to different weather conditions. Therefore, they may be preferred for certain specific applications, but not for continuous monitoring for the early detection of possible disasters.

2.3 WMSNs and Machine Learning for (Forest) Fire Detection

Studies on forest fire detection using multimedia data typically includes studies of image processing, video processing, and, rarely, machine learning techniques. However, following their success in classification, object recognition, and detection, approaches based on machine learning (especially deep learning) have also proven successful in the recognition of events. As a result, many of the existing studies on event analysis rely heavily on deep learning architectures [1].

A wildfire detection framework based on image processing is presented in [20]. Scale invariant feature transform is used for feature selection, and SVM and KNN classifiers are applied for fire classification, as fire and non-fire images. A combination of CNN and Recurrent Neural Networks (RNN) is used to create a DL model for detecting fires from video sequences in [21]. On average, an accuracy of 92% is obtained. In [43] a CNN-based early detection method is proposed where the processing capabilities of CCTV cameras are used. The proposed method is applicable to indoor and outdoor environments. However, the model size is quite large (238 MB) for using the model in WMSN applications. Another CNN-based fire detection system is proposed in [36] where UAV images are used for training of the network. The system proposed in [36] achieves an accuracy of 99% with GoogLeNet.

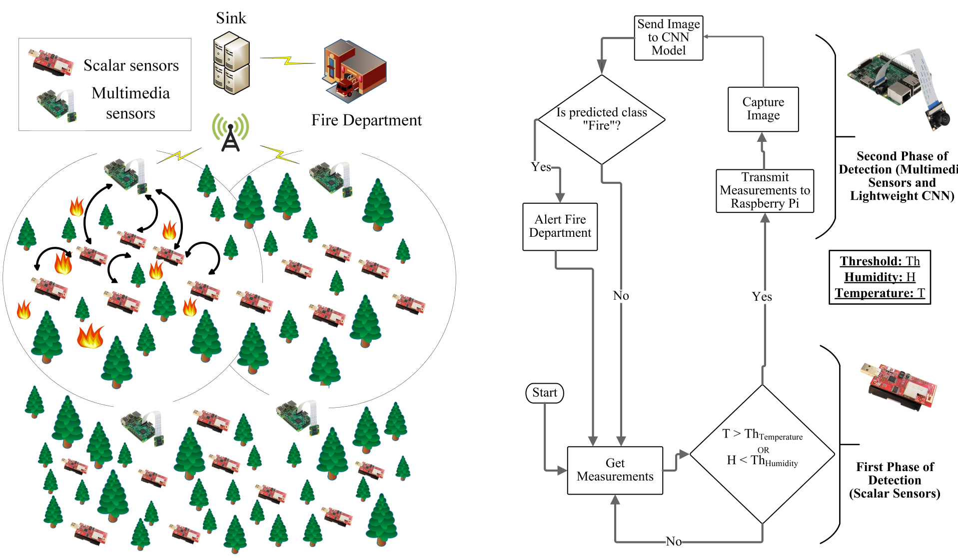
When the existing studies are analysed in detail, the cooperative use of multimedia and scalar sensors have been investigated and explored in various surveillance-based studies. However, there aren't as many studies focusing on forest fires and trying to maximise the energy efficiency and similar QoS measures together with forest fire detection accuracy. Considering the WMSN applications of forest fire detection in the literature, the majority of studies rely solely on image and video processing to detect forest fires. Only a few studies, such as [21], use machine learning models, including ANN, SVM, and RNN. On the other hand, the proposed machine learning models are not suitable for use in IoT edge devices because they require relatively large memory and computing power. In this study, a lightweight CNN model is developed which can be used on edge devices with limitations in terms of computing, memory, and power resources. The proposed lightweight CNN model and the hierarchical forest fire detection framework provide a much more efficient solution in terms of energy consumption without compromising detection accuracy. To the best of our knowledge, this is the first study in which CNN is used in the smart sensors at the edge of the network for the detection of forest fires. The proposed model is trained using a newly created forest fire dataset.

3 FOREST FIRE DETECTION FRAMEWORK

The proposed effective forest fire detection framework uses scalar sensors for energy efficiency and multimedia sensors (for capturing snapshots) to improve the accuracy with the aid of DL and edge computing. XM1000 sensors and Raspberry Pi (RPi) 3s are used in this study as scalar and multimedia sensor nodes respectively. An example scenario for the proposed framework is shown in Figure 1a, and a flowchart is presented in Figure 1b to illustrate the flow of information. In Figure 1b, T, H, $Th_{Temperature}$, and $Th_{Humidity}$ are used to denote the current temperature, the current humidity, the threshold of temperature and the threshold of humidity values respectively.

The area to be monitored is divided into regions. Each region has one multimedia sensor node with Raspbian OS and a number of scalar sensors using Contiki OS [13]. Multi-hop communication is considered between scalar and multimedia sensors, and, the built-in CC2420 transceiver is employed with ZigBee 802.15.4 protocol in the radio layer. In Radio Duty Cycling layer, ContikiMAC is used for low-power listening with energy efficiency. In the MAC and network layers, CSMA/CA and Routing Protocol for low power and lossy networks (RPL) are used, respectively in order to deal with possible packet collisions.

In the first phase of the detection, scalar sensors are used to monitor the forest environment by measuring the temperature and humidity. The measured values are compared to the threshold values. The seasonal mean temperature and humidity of the deployment area (Cyprus in our case) are used in calculating the thresholds for humidity and temperature. We can denote the seasonal mean temperature and humidity of the deployment area as $Mean_{temperature}$, and $Mean_{humidity}$ respectively. For the temperature, the threshold is computed by considering the mean seasonal temperature with a tolerance of an additional 10% ($Th_{Temperature} = Mean_{temperature} \times 1.1$). Similarly, for humidity,



(a) Forest fire detection system architecture (b) The flowchart of the framework

Fig. 1. Forest fire detection system architecture and information flow chart for first and second phase of detection.

seasonal mean humidity -10% is considered as the threshold ($Th_{Humidity} = Mean_{humidity} \times 0.9$). If the measured temperature is greater than the threshold value of temperature or the measured humidity is less than the humidity threshold, all the measured values are transmitted to the multimedia sensor node in the region via intermediate nodes. The transmitted packet includes values of temperature, humidity, light, voltage, id, number of hops, sequence number, packet size, date and time. The nearby sensors detect the fire quicker and transmit the triggering packet to the cluster head (multimedia sensor). The id of each sensor is assigned prior to the deployment and can be used to estimate the sensor's location and its cluster. The second phase of detection is initiated by the receiving node (the smart sensor which is RPi in this scenario). In the second phase, RPi activates the pi camera to capture an image of the environment. The camera of the multimedia sensor is turned on even in case a single activation packet is received from the scalar sensors for introducing a robust fire detection framework. The captured image is processed by the proposed lightweight CNN model at the edge to detect forest fire. If the result indicates a forest fire, the fire department is informed. With the proposed framework, multimedia data transmission is not necessary since the lightweight CNN model runs on the edge devices. In addition, snapshots are captured as many times as the scalar sensor notifications. Since the snapshots are not transmitted to the sink (processed on the edge with lightweight CNN), their effect on the lifetime of the sensors is not very significant. According to our experiments, compared to the idle state, RPi consumes 0.12 W extra power while running the proposed lightweight CNN. On the other hand, compared to the idle state, the additional power consumption of the RPi increases to 1.07 W while transmitting the images. Furthermore, in case the scalar sensors generate multiple notifications, the probability of false negative is reduced since each warning is taken into account. This, in turn, increases the reliability of the framework.

In order to achieve energy efficiency, three new features are being used in the proposed framework. The first feature is the hierarchy. Scalar sensor nodes are used at the first level of the hierarchy because they consume much less power than the multimedia sensor nodes. In the implemented testbed, scalar sensors monitor the environment by taking measurements in every 10 seconds. Instead of sending associated information (packets) every 10 seconds, the measurements are examined in each individual sensor node. No transmission is carried out by scalar sensors except in the cases where measured values are not within the threshold values.

The second feature used is heterogeneity. Heterogeneous wireless networks are more appropriate solutions for surveillance and disaster management applications. In our case, the conditions of the forest environment can be difficult to deploy, manually configure, and maintain sensor nodes. In case of fire, it is also quite possible to lose a large number of sensors. In this sense, it is preferable to use scalar sensors that are inexpensive, easy-to-deploy, self-configurable, and energy-efficient. In studies such as, [14],[17], [30], [42], [25], and [27] large numbers of scalar sensors are employed for environmental monitoring including detection of forest fires. On the other hand, the accuracy of fire detection is also crucial. By applying heterogeneity, the energy efficiency is achieved by scalar sensors and the accuracy is obtained by multimedia sensors.

The other approach used to achieve energy efficiency and accuracy includes the use of edge computing. For the second phase of detection, a lightweight CNN-based model is developed. The model developed uses the images captured by the RPi. Applying DL within the RPi reduces the number of transmissions needed from the multimedia nodes to the sink.

4 FOREST FIRE DETECTION USING DEEP LEARNING

The main objective of the proposed lightweight CNN model is to provide a robust approach for forest fire detection. As discussed in [44], CNN based image classification, especially in fire detection, provides substantial improvement in terms of detection accuracy which in turn minimizes disastrous results of fires. However, the main drawback of CNN based classification is high resource requirements which can become an important problem for resource limited edge devices. In order to mitigate these limitations, a lightweight CNN model is proposed in this study which can be used at edge devices with considerably high detection accuracy. Well-known image classification neural networks, and the best combination of hyper-parameters are investigated to adapt the most suitable architecture for the model.

Another important task for preparation of the model is the dataset creation.

Existing literature is rather limited in terms of datasets, and the ones available are typically used to detect fires from smoke images. However, our study specifically focuses on forest fires in various forest environments while aiming for detection from still images considering both smoke and fire. For the generation of datasets, close-up videos of forest fires are examined and the necessary frames are extracted and labeled. Labeling process is verified by re-checking the labeled images manually to avoid incorrect labeling.

4.1 Dataset

Since public image datasets for forest fire detection are not available, wildfire videos are collected on YouTube and sampled according to camera movement. Manual removal is performed to avoid incorrect labeling. The dataset generated includes 1111 fire images of 16 videos and 2,289 non-fire images from 9 videos, for a total of 3,400 images. Furthermore, different seasons and lightning conditions are taken into account. Fireless forest images are obtained for different seasons. While winter forest images are around 13% of the dataset, fall, spring and summer forest images are around 24%, 38%, and 25% of the dataset, respectively. On the other hand, forest fire images are obtained in different lighting conditions where 40% of images are at nighttime and 60% of images

Fig. 2. Fire Image Examples

Fig. 3. Non-Fire Image Examples

are at daytime. In addition, while 72% of the set used as fire images has both fire and smoke, 12% has only fire, 5% has only smoke. Various forest fires in Canada, California and Turkey are taken into account. Some examples of images in the dataset are shown in Figures 2 and 3. The division of training and testing data is performed with random sampling from each class with a specific ratio based on the validation method selected. To illustrate, in case of a 80-20% split, for validation set, 20% of each class is randomly selected and added into the validation set, and the remaining 80% is considered as a training set.

Considering some of the publicly available datasets such as [36], [31] and [60], or some studies which use images extracted from videos similar to our approach such as [20], [21], [56], [65], our proposed dataset contains more images than any existing dataset for forest fire detection. In [36] aerial videos are used for image collection since the main focus is fire detection using aerial vehicles. However, considering the system and infrastructure that we propose for forest fire detection, aerial images are not suitable to train our lightweight CNN. It is possible to further expand the dataset with additional images from different video streams. However, the size of the dataset is sufficient to represent the capability of the CNN model placed at the secondary level, since the primary focus is early detection unlike the other forest fire related attempts which try to predict forest fires, the direction of the fire, or their resulting damage [11].

4.2 CNN Model

A lightweight CNN model is proposed to detect forest fires accurately. Although DL is popular in various image classification tasks, forest fire detection as a specific domain offers only limited work with DL architecture. In addition, the computation time requirements of hardware platforms that can be used for forest fire detection systems should be considered when developing the model.

Therefore, this study proposes a DL model that can work on small computers such as Rasperry Pi, Orange Pi, and Hikey 970. The model is developed and tested using PyTorch and Keras libraries [9].

The proposed CNN model uses 64x64x3 input images. The CNN Architecture contains four convolutional layers each followed by a max pooling layer with 2x2 kernels for feature extraction, and uses three fully connected layers for classification. In addition, each fully connected layer contains a dropout layer with the rate of 0.25. As shown in Figure 4, the proposed CNN model is kept as shallow as possible because the deeper networks have higher time complexity and memory related requirements.

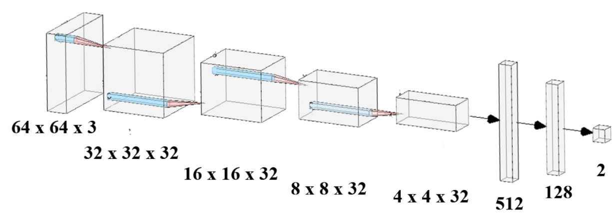


Fig. 4. CNN Model Architecture

The lightweight CNN model presented here is trained using the newly created dataset on forest fires. A distribution ratio of 80-20% of the train-test is applied. Stochastic gradient descent is used as an optimizer since it is widely used in CNN applications and studies such as [39], [64], and [66]. As an activation function, RELU (Rectified Linear Unit) function is used.

5 EVALUATION OF THE PROPOSED SYSTEM

The proposed framework is evaluated in terms of energy efficiency, detection accuracy and packet delivery rate using simulations and testbed implementations. OMNET++ is employed for simulation, and the newly introduced lightweight CNN is deployed in RPi for the testbed. Three different scenarios are compared. In the first scenario, only scalar sensors are used to detect a forest fire by measuring the temperature. If the measured temperature exceeds the specified threshold, the sink node alerts the fire department. Note that the fire threshold may be specified based on geographical location and the monitoring season. The second scenario uses only multimedia sensors. The Raspberry Pi hardware and the pi camera are used as multimedia sensors. The images captured by RPis are transmitted to the sink node via intermediate multimedia sensors. Then, the fire department is alerted if a fire is detected. The last scenario is the hierarchical and heterogeneous system proposed in this study. Please note that for WSNs, the correct deployment strategy depends on the application considered, as discussed in [3] and [15]. In [3], the results presented show that the uniform random deployment strategy provides better performance than the triangle grid, the hexagon grid and the tri-hexagon tiling (THT) deployment strategies in terms of coverage. Furthermore, in studies such as [54], random scattering of sensor nodes is identified as one of the most common deployment strategies for forest fires since the monitoring regions are considered in the category of large-scale open areas. As a result, for the three scenarios considered, it is assumed that the sensors are uniformly deployed to achieve maximum possible field coverage and connectivity between sensor nodes. The scalar sensors are, in turn, clustered around multimedia sensors. This assumption applies to simulation analysis as well as testbed based experiments. Once deployed, the scalar sensors periodically (every 10 seconds to allow early observation) capture the data (temperature and humidity reads) to monitor the environment while multimedia sensors capture the data (image) when scalar sensors trigger them.

5.1 Simulation

The approach introduced in [49] is used to calculate the energy consumption of scalar and multimedia sensors. It is possible to calculate the power consumption as:

$$P_{cons} = VI_m \tag{1}$$

where V is the supply voltage, I_m is the weighted average of current consumption, and P_{cons} is the consumed power. In order to calculate the weighted average of the current used, the states of the sensor nodes and the time fractions for each state are considered. A sensor node can be in one of three different states at a time. The idle state is the state where the sensor node neither receives nor transmits. In tx state, the sensor node transmits packets to other nodes in the communication range. In the rx state, the sensor nodes listen the communication medium for incoming packets. In each state, there is a constant current consumption which is available in data sheet of the sensor node. Then, the fractions of time for each node are calculated by experiments. The weighted average of current consumption is calculated as follows:

$$I_m = \delta_{idle}I_{idle} + \delta_{tx}I_{tx} + \delta_{rx}I_{rx} \tag{2}$$

where, δ_{idle}, δ_{tx}, and δ_{rx} are time fractions for idle, tx and rx states respectively. Moreover, I_{idle}, I_{tx}, and I_{rx} illustrates current consumption for the idle, tx, and rx states. According to [49], it is possible to assume that the sum of the time fractions is equal to one, which can be expressed as follows:

$$\delta_{idle} + \delta_{tx} + \delta_{rx} = 1 \tag{3}$$

This energy consumption model, is used in the simulations. The results of the simulations, are presented comparatively with the results obtained from the testbed implementation.

The duration of the simulations is different for each of the scenarios explained (scalar only, multimedia only, our approach) since the simulation runs until the first node dies. Sensors are deployed uniformly on a 50×50 m^2 square field, and there are six nodes in total, including the sink node for each scenario. The scalar sensors sense the environment once in every 10 seconds to allow early observation of potential calamities. The XM1000 packet size is 200 bytes to transmit the temperature value, while the size of the packets transmitted by RPi is 2 MB since the captured images are communicated. The initial energy of each node is 30,780 joules for XM1000 sensor node, which corresponds to the initial energy of two AA batteries, and 234,000 joules for RPi, which corresponds to the initial energy of 13,000 mAh power bank. The output power Tx of the nodes in each scenario is given as 0 dBm.

To calculate the power consumption of the XM1000, the radio duty cycle is calculated using the approach introduced in [49]. The supply voltage and the mean current consumed by the sensor are calculated. To calculate the mean drawn current, experiments are conducted on the XM1000 mote using "powertrace" library in Contiki OS and the time fractions for each state are extracted. Please note that in scalar-only scenario, since the transmission of packets is periodical, the time fractions for idle, Tx, and Rx states are not fire scenario dependent. The mean drawn current and the time fractions for idle, Rx and Tx states are computed as 2 mA, 17.4 mA, 18.8 mA, and 0.9619, 0.0075, 0.0306, respectively.

To compute the power consumption of the RPi, experiments are conducted using the available testbed. The results show that the RPi consumes 1.68W power at the idle state (pi camera not activated) and 2.75W power at tx/rx state. Additional experiments are conducted to obtain the fraction of time required and the amount of power consumed for image processing as well. The experimental results obtained show that multimedia sensors consume additional 0.12W power while processing images with proposed lightweight CNN. Multimedia nodes process packets of around 2MB once to decide whether there is a fire or not. The proposed lightweight CNN requires 0.3 seconds to process an image.

In the proposed framework, scalar sensors continue to periodically measure the environment every 10 seconds and they last much longer than the RPi multimedia node. RPi continuously listens the serial port to receive notifications. If it is triggered by a high temperature and low humidity reading, the pi camera is activated to capture an image. As a case study, the data on forest fires in Cyprus are used to find the fraction of the times the pi camera is activated. The data available in [12] includes a number of forest fires for the period of 2000-2017. Using the average number of forest fires, it is possible to calculate the fraction of time of the occurrence of a fire during the lifetime of RPi. The number of fire events that can occur during the lifetime of RPi is used in the configuration files of OMNET++ Castalia framework to generate fire events randomly for the specific simulation period considered. This way, since the number of events generated are taken from a real forest fire dataset, the number of times different functionalities of our framework are activated is as realistic as possible. The Castalia simulation tool [46] is used to simulate the three scenarios considered. Castalia is based on OMNET++ platform and it is generally used for the networks of low power sensor devices. It is possible to model different types of sensors in Castalia providing the corresponding configurations and protocols for each individual unit such as battery unit, communication unit, sensing unit, and processing unit. The XM1000 (scalar) and RPi (multimedia) sensors are modeled in the Castalia simulator providing appropriate configurations for each unit. Additionally, simulation results are presented along with actual benchmark results comparatively for validation.

In all simulation scenarios, the sensors are uniformly deployed in the fields considered. In both only-scalar and only-multimedia scenarios, all sensors are homogeneous, with the exception of one sensor selected to act as the sink. Packets are sent periodically in both scenarios, while scalar sensors send temperature and humidity readings and multimedia sensors send captured images. In the framework scenario proposed with the hierarchical setup, the sink node is a multimedia sensor and the source nodes are scalar sensors. The system is event driven, where scalar sensors take temperature and humidity measurements and require transmission only in case the readings are not within the pre-specified threshold values. Scalar sensor transmissions trigger the multimedia sensor to capture an image and use the proposed lightweight CNN model for forest fires detection. The packet reception rates are also presented using the OMNET++ simulation which keeps the records of numbers of packets transmitted, received and dropped.

When the first scenario (scalar sensors only) is considered using simulation implemented with the XM1000 sensor specifications, the lifetime (the first node dies) of the network is on average 43.55 days with two AA batteries. Since only scalar data is transmitted in this scenario, the system works well in terms of energy efficiency.

The results of the simulations for the second (only multimedia sensors) and third scenarios (proposed framework) show that the lifetime of the system considered in the second scenario is approximately 11.16 hours, whereas our proposed model has a lifetime of about 15.93 hours with a power bank of 13000 mAh. According to the simulation results, our proposed framework is 29.94% more efficient than the scenario in which only multimedia sensors are used. The heterogeneous nature of the proposed framework in which the scalar sensor readings are used to trigger the camera dramatically improves the lifetime of the multimedia nodes. In addition, the introduction of a lightweight CNN model that enables edge computing on RPi reduces the unnecessary transmission of video data and increases the system efficiency. Furthermore, additional simulation experiments show that using 33, 100 and 200 Ah batteries (each one of 12 V), it is possible to achieve the lifetimes of 4.70, 14.25 and 28.50 days respectively.

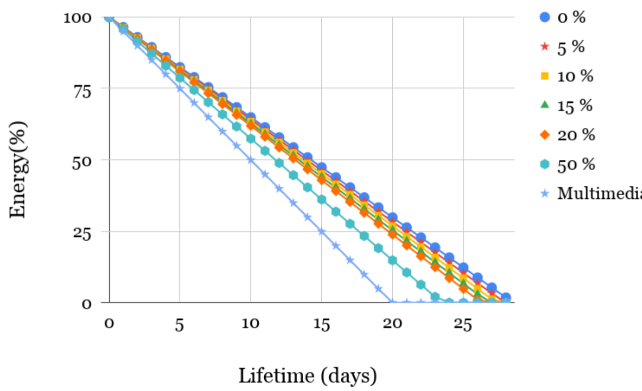


Fig. 5. Effect of False Alarm Rate on Lifetime

Please note that the false positive information from scalars where the multimedia sensors are switched on although there is no fire, can shorten the lifetime of the system. The effect of these false alarm rates on the lifetime is studied in Figure 5. The lifetime of the proposed system is shown for systems using 200 Ah batteries with different rates of false alarms. Scenarios are considered for cases where 0%, 5%, 10%, 15%, 20%, and 50% of the incoming notifications to activate the cameras are incorrect. In addition, the scenario where only multimedia sensors are employed is illustrated for a comparative analysis. The results presented in Figure 5 clearly show that the proposed system outperforms multimedia scenario even when 50% of the camera activation notifications are incorrect. This is reasonable since the camera is always-on in the multimedia scenario. However, in the proposed system, the camera does not operate unless it is triggered by an event. Please also note that the values used for threshold can also affect the false alarm rates. Therefore, the threshold values should be decided according to the temperature and humidity profile of the environment that will be monitored. In this work, we have conducted experiments considering Cyprus as a case study, and we have used available forest fire, temperature, and humidity data [12] to specify the threshold values.

Instead of sending all the messages to the base station for decision making, the proposed system uses edge computing which can reduce the overall traffic load significantly. Therefore QoS metrics should also be considered when evaluating the system. Instead of sending all the messages to the base station for decision making, the proposed system uses edge computing which can reduce the overall traffic load significantly. Therefore QoS metrics should also be considered when evaluating the system.

Since the unnecessary packet transmissions are avoided, the channel traffic decreases and the packet loss rate is reduced. Thus, by applying heterogeneity, machine learning and edge computing, the proposed system becomes more accurate, more energy-efficient and more effective in terms of QoS compared to the existing systems. The scalability of the proposed system is studied in Figure 6. The lifetime and the successful packet reception rate of the multimedia nodes are presented as functions of the number of nodes considered in each region. The scenario in which only the multimedia sensor nodes are used is considered, as well as the scenario with the proposed framework.

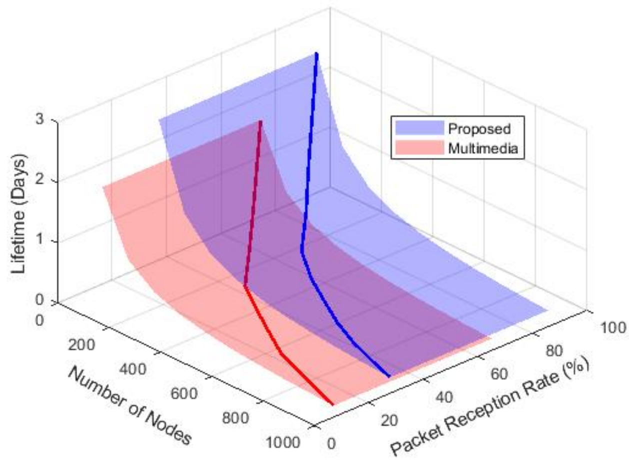


Fig. 6. Lifetime and packet reception rate of the proposed framework

For forest fire detection applications, successful delivery of transmitted packets is very critical. The results clearly show that the proposed approach is suitable for emergency applications since the reception process is more reliable and energy efficient.

5.2 Testbed Implementation

The three scenarios considered for the energy efficiency assessment are used to conduct benchmarking experiments. The main purpose of the testbed is to show the effectiveness of the proposed approach comparatively and to validate the simulation results presented for the lifetimes of the sensor nodes. The detection accuracy is considered in the section on the evaluation of the proposed deep learning model. The testbed based evaluation approach employed is similar to the testbed based scenarios (referred to as environment simulation) presented in studies such as [10], [16], [33], [51]. A small scale representative is deployed for development and testing purposes. Forest fire warnings are injected to the system by considering forest fire probability similar to the approach used for the simulation to show that with close numbers of forest fire notifications, the simulation and the testbed results confirm each other in terms of energy efficiency. Figure 7 shows the scalar and multimedia sensors used in the testbed implementation. The multimedia sensor with the attached scalar sensor is the sink node as a whole in our setup. The scalar sensor attached is not used for sensing. Instead, it is used as a communication interface to collect incoming packets from the scalar sensors to the sink.

In order to monitor the voltage levels, a library “battery-sensor" is used in Contiki OS. According to the results of the testbed, when the first scenario is considered (only scalar sensors), the XM1000 sensor nodes discharges two AA batteries in 42.50 days, which is very close to the results of our simulation computed as 43.55 days. The discrepancy is less than 3.5%. For the second scenario, (only multimedia sensors), the lifespan of the multimedia sensors is approximately 11.88 hours, which is very close to the simulation results obtained as 11.16 hours. The difference between the simulation results and those of the testbed is this time less than 6.1%.

The power sources used for the proposed framework are the same as those of previous experiments for scalar and multimedia sensors. Voltage levels are monitored to specify the lifetime of

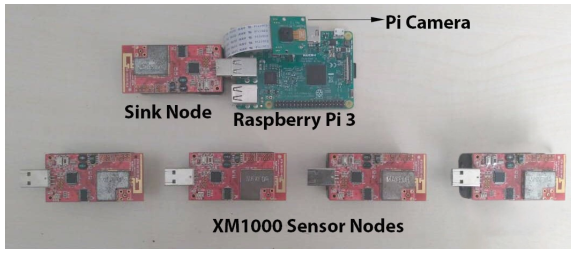


Fig. 7. Experimental Setup

the nodes. The testbed validates the simulation results for the third scenario as well. While the lifetime of the proposed model is 15.93 hours in the simulation, it is 15.66 for the testbed. The difference between the results is less than 1.8%. Results obtained for all three of the scenarios can be summarized as presented in Table 2.

Table 2. Summary of Results

	Scalar	**Multimedia**	**Proposed Framework**
Testbed	42.50 days	11.88 hrs	15.66 hrs
Simulation	43.55 days	11.16 hrs	15.93 hrs
Discrepancy	< 3.5%	< 6.1%	< 1.8%

The evaluation carried out using the testbed assumes that the deployment of the sensors is performed in a way to allow the framework to function as effectively as possible. However, please note that although similar assumptions are used for testbeds in some studies such as [10], [16], [33], [51], the correct deployment of the scalar and multimedia sensors is very important for all the frameworks proposed for wildfire detection. The results presented comparatively for the testbed employed, and the simulation shows a good agreement with less than 3.5%, 6.1%, and 1.8% discrepancies for scenarios with scalar sensors only, multimedia sensors only, and the proposed framework, respectively. The results obtained from the testbed implementation also show that systems based on scalar sensors consume less energy than multimedia sensors. However, as stated in many studies dealing with disaster scenarios, we live in a time where multimedia data can be processed autonomously for much greater accuracy, especially when the correct classification is essential. On the other hand, since the proposed framework uses the efficiency of scalar sensors together with the accuracy of multimedia sensors, the power consumption of the system is reduced by about 29% when compared to the solutions which are solely based on multimedia sensors.

5.3 Evaluation of the Proposed Deep Learning Model

The new dataset which contains the forest images with and without fire is used to evaluate the proposed lightweight CNN model. The model proposed is created by testing different hyperparameter settings with the consideration of the model's lightness. The best performing CNN model proposed has convolution with three filters in the first and 32 filters for the rest of the layers. In order to ensure that the model presented is not overfitting, 10-fold cross validation is performed. In the preliminary tests (a single pass with mixed train-test split), the model reaches to an accuracy of 99.12%. For single pass and 10-fold cross validation processes, stochastic gradient descent is used as an optimizer. Rectified Linear Unit (RELU) is used as an activation function. Each layer contains a dropout at the rate of 0.25. The model is trained for 100 epochs with a learning rate of 0.01 using cross-entropy loss, and early stopping with patience of 10 epochs. In other words, if the model no longer improves for 10 consecutive epochs, the training process is terminated and latest version of the model is saved.

The representation used for the captured images is critical for the whole system as it directly affects the complexity of the model. Complexity, in turn, specifies the computational and storage resources required in smart sensors. Although the input sizes can be reduced, the detection accuracy must be at acceptable levels. In the proposed framework, the images are resized to 64x64x3 in order to reduce the computational cost. This reduction in the input size does not significantly affect the learning process since the proposed framework mainly focuses on early detection. It is sufficient to use the input for classification as fire or non-fire. Other features needed for localization of the fire or for the detection of other factors such as the direction of the fire, are not necessary for early detection. While the input size is reduced as explained, the model proposed in this study comes with a detection accuracy of 98.28%. There are similar studies in the literature which use well-known models such as [21], [43], and [36] (GoogLeNet architecture) with fire detection accuracies of around 92%, 94.39%, and 99% respectively. Compared to these approaches, our proposed model works almost as accurately as the best performing model, but requires reduced amounts resources in terms of computation and storage.

When the characteristics of the model we propose are compared with the characteristics of the models used in existing machine learning based fire detection systems in more detail, we see that only a few of the studies such as [20] focus on forest fire detection. Studies such as, [21], [36], and [43] mainly consider fire detection in various environments. Similarly, while [20] uses SVM and KNN as machine learning algorithms, [21], [36], and [43] use CNN based approaches. In [21], RNNs are employed in addition to the CNN based architectures. All of these studies mainly focus on detection accuracy with well-known architectures such as GoogLeNet. They do not consider the QoS or energy efficiency related issues of the underlying systems. Instead, in our proposed system, by using a hierarchical approach, we can improve the underlying infrastructure in terms of QoS and energy efficiency. The size of the employed machine learning architecture can be quite critical in order to make sure that we can use the presented architecture in various types of multimedia-enabled smart devices. The model size of the deep learning architecture used is provided only in [43] as 238 MB. The CNN architecture proposed in this study comes with a model size of 1.4 MB. Therefore the improvement in terms of edge computability is quite significant.

In order to show that the proposed model is lightweight, it is compared to well-known CNN architectures in terms of float-point operations (FLOP) [6]. While the proposed model have less than 0.55 GFLOPs, the majority of well-known models have more than 1 GFLOPs in terms of computational requirements. For example, RESNET-50 has about 4 GFLOPs and VGG-13, 16 and 19 have more than 10 GFLOPs [6]. In order to further emphasize the efficacy of the proposed model, we have also tested the well known light weight models such as Sufflenet [40], Squeezenet [24], MNasnet [58], Alexnet [34], Mobilenet [53], Resnet [22], and Inception [57] using our forest image dataset. Table 3 shows the number of parameters, FLOPS, model size, and accuracy for each model comparatively with the model proposed in this study.

When the results presented in Table 3 are considered in terms of the numbers of parameters, the MNasnet, and Mobilenet models are comparable to the proposed model. However, the proposed model is superior to these models in terms of accuracy since Mnasnet and Mobilenet have accuracies of 64.05% and 97.58%, respectively, as opposed to 98.28% accuracy of the proposed model. In terms of the FLOPS, the only model comparable to our proposed model is the Sufflenet. Although the accuracy of the Sufflenet is slightly better than our proposed model, in terms of the model size, while our proposed model is around 1.4 MB, the the Sufflenet model is around 9 MB. When the

Table 3. Comparison of the proposed model with other light weight approaches

Model Name	Parameters (G)	FLOPS (G)	Accuracy (%)	Model Size (MB)
Proposed	**0.0040**	**0.5362**	**98.28**	**1.40**
Sufflenet	0.0230	0.4516	98.81	9.02
Squeezenet	0.0120	1.7368	30.5	4.89
MNasnet	0.0044	0.9284	64.05	17.33
Alexnet	0.0611	5.0179	99.00	238.68
Mobilenet	0.0035	0.8992	97.58	13.84
Resnet18	0.0117	4.7833	99.00	45.72
Inception	0.0272	183.425	99.72	106.35

model size is considered, our proposed model is significantly more lightweight than the rest of the models. In terms of size, the closest model to the one we propose is Squeezenet. However, its size is still more than three times the size of the proposed model (4.89 MB), and its accuracy is significantly low (30.5%).

The size of the model and the required computational resources are critical for the proposed framework as these are the main factors affecting the ability to perform early detection at smart end nodes. The use of deep learning architectures with various types of boards for edge computing is also becoming very popular [7], [23], [52]. However, one of the most significant limitations in using various boards to run lightweight deep learning algorithms is the availability of resources, especially in terms of memory requirements. Therefore, these devices are referred to as "Memory-Constrained Edge Devices" [7]. For example, STM32F2 series comes with up to 120 MHz CPU speed, 1 MB of Flash, and up to 128 kB of SRAM [29], whereas Arduino MEGA 2560 supports clock speeds up to 16 MHz, with 256 KB of flash program memory and 8 KB of SRAM [26]. We, therefore, think that all of the metrics presented for the model evaluation play crucial roles. In terms of FLOPs, the proposed approach is at least 1.6 times better than all other approaches except Sufllenet, which performs similar to ours. The proposed approach is superior to all other models in terms of model size, which can be a significantly limiting factor depending on the type of end devices being used. Our proposed architecture is more than three times smaller than the next smallest architecture (Squeezenet) and more than six times smaller than Sufflenet. Finally, in terms of accuracy, the best models are Inception, Resnet18, and Alexnet with accuracies of 99.72%, 99.00%, and 99.00%, respectively, as opposed to the accuracy of the proposed model which is 98.28%. Although the accuracy of the proposed model is quite close to the highest accuracy, in terms of the number of parameters, FLOPS and model size, the proposed approach is significantly better especially compared to Inception, Resnet18, and Alexnet approaches. Please also note that for relatively deeper models such as Inception, images of size 300x300x3 should be used since 64x64x3 images vanish during the process of the forward pass due to the pooling and padding operations.

We end this section with a summary of the performance of the proposed model in terms of accuracy. The results of the evaluation carried out show that with an accuracy of about 98.28%, the proposed model is superior to studies such as [21], [43], [30], [56], which respectively report accuracies of 94.39%, 95.45%, 95%, and 86%. The study presented in [36] performs slightly better than the proposed architecture with an accuracy of 99%; however, it is primarily based on GoogLeNet, which is a 22-layers convolutional neural network. On the other hand, the well-known architectures Mnasnet, Mobilenet, Sufflenet, Inception, Resnet and Alexnet, respectively provide accuracies of

64.05%, 97.58%, 98.81%, 99.72%, 99.00% and 99.00%. In terms of accuracy, the proposed approach performs close to the approaches with the highest accuracies.

6 CONCLUSION

In this study, an effective and energy-efficient framework is proposed for the detection of forest fires. Unlike existing studies, multimedia sensors employing deep learning models are used alongside scalar sensors, and detection is performed using the fusion of information at different levels of the system architecture. In addition, the efficiency of communication is improved by the introduction of edge computing for decision making. The proposed framework uses the efficiency of scalar sensors and the accuracy of multimedia sensors. Through the establishment of a hierarchical infrastructure, the new framework balances energy efficiency and detection accuracy and provides a sustainable emergency surveillance system.

The results of the simulation and testbed experiments are presented and discussed in a comparative way. According to the experimental results, an improvement of 29.94% is obtained for energy efficiency (up to 28.50 days). In addition, a new lightweight CNN model is proposed to enable processing on edge devices. According to the results of the experiments, an accuracy of 98.28% is achieved. When the existing fire detection studies are taken into account, our approach is able to perform better than any other in terms of accuracy with the exception of the study included in [36]. Although [36] uses GoogLeNet, a 22-layer deep convolutional neural network, the performance of our approach is close to this architecture. Since the existing approaches do not take into account the limitations of edge computing, in this study, the existing lightweight architectures are also used in the experiments performed for comparison. The proposed approach performs very similar to the lightweight architectures with the best performance in terms of accuracy. However, the proposed approach's main advantage is its suitability for edge computing on resource limited sensor nodes. To be able to use the CNN architecture in multimedia-enabled smart end devices, the model presented is set to be as light as possible in terms of the required computations and the model size. In terms of FLOPs, the proposed approach is at least 1.6 times better than all other approaches except Sufllenet, which performs similar to ours. The proposed approach is superior to all other architectures in terms of the model size, which can be a significant limiting factor depending on the type of end devices being used. The efficiency in terms of FLOPs and model size makes the proposed architecture the best candidate for smart nodes with limited capacities enabling early detection at edge nodes.

For future work, we plan to use our lightweight, high-accuracy detection mechanisms in other application areas. We believe that the lightweight architectures presented in this paper can be practical and useful, especially in eHealth domain.

ACKNOWLEDGMENT

This work was supported by NU Faculty-development competitive research grants program, Nazarbayev University, Grant Number-110119FD4543.

REFERENCES

[1] Kashif Ahmad and Nicola Conci. 2019. How deep features have improved event recognition in multimedia: a survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 15, 2 (2019), 1–27.
[2] Abdulla Al-Kaff, Ángel Madridano, Sergio Campos, Fernando García, David Martín, and Arturo de la Escalera. 2020. Emergency Support Unmanned Aerial Vehicle for Forest Fire Surveillance. *Electronics* 9, 2 (2020), 260.
[3] Ibtihal Alablani and Mohammed Alenazi. 2019. Performance Evaluation of Sensor Deployment Strategies in WSNs Towards IoT. In *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*. IEEE, 1–8.

[4] Yunus Emre Aslan, Ibrahim Korpeoglu, and Özgür Ulusoy. 2012. A framework for use of wireless sensor networks in forest fire detection and monitoring. *Computers, Environment and Urban Systems* 36, 6 (2012), 614–625.
[5] Anand S Bhosle and Laxmikant M Gavhane. 2016. Forest disaster management with wireless sensor network. In *2016 Int. Conf. on Electrical, Electronics, and Opt. Tech. (ICEEOT)*. IEEE, 287–289.
[6] S. Bianco, R. Cadene, L. Celona, and P. Napoletano. 2018. Benchmark Analysis of Representative Deep Neural Network Architectures. *IEEE Access* 6 (2018), 64270–64277. https://doi.org/10.1109/ACCESS.2018.2877890
[7] Alessandro Capotondi, Manuele Rusci, Marco Fariselli, and Luca Benini. 2020. CMix-NN: Mixed low-precision CNN library for memory-constrained edge devices. *IEEE Transactions on Circuits and Systems II: Express Briefs* 67, 5 (2020), 871–875.
[8] Yanhong Chen, Youmin Zhang, Jing Xin, Guangyi Wang, Lingxia Mu, Yingmin Yi, Han Liu, and Ding Liu. 2019. UAV Image-based Forest Fire Detection Approach Using Convolutional Neural Network. In *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2118–2123.
[9] François Chollet et al. 2015. Keras.
[10] Muhsin Civelek and Adnan Yazici. 2017. Automated moving object classification in wireless multimedia sensor networks. *IEEE Sensors Journal* 17, 4 (2017), 1116–1131.
[11] Paulo Cortez and Aníbal de Jesus Raimundo Morais. 2007. A data mining approach to predict forest fires using meteorological data. (2007).
[12] Forests Department. 2018. FOREST FIRE STATISTICS FOR 2000-2017, Ministry of Agriculture Rural Development and the Environment. http://www.moa.gov.cy
[13] Adam Dunkels, Oliver Schmidt, Niclas Finne, Joakim Eriksson, Fredrik Österlind, and Nicolas Tsiftesand Mathilde Durvy. 2011. The contiki os: The operating system for the internet of things. *Online], at http://www. contikios. org* 605 (2011).
[14] Ozgur Ergul, Ghalib A Shah, Berk Canberk, and Ozgur B Akan. 2016. Adaptive and cognitive communication architecture for next-generation PPDR systems. *IEEE Communications Magazine* 54, 4 (2016), 92–100.
[15] Mohammed Farsi, Mostafa A Elhosseini, Mahmoud Badawy, Hesham Arafat Ali, and Hanaa Zain Eldin. 2019. Deployment techniques in wireless sensor networks, coverage and connectivity: A survey. *IEEE Access* 7 (2019), 28940–28954.
[16] Sheikh Ferdoush and Xinrong Li. 2014. Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications. *Procedia Computer Science* 34 (2014), 103–110.
[17] Jun-Song Fu, Yun Liu, Han-Chieh Chao, and Zhen-Jiang Zhang. 2016. Green alarm systems driven by emergencies in industrial wireless sensor networks. *IEEE Communications Magazine* 54, 10 (2016), 16–21.
[18] Xiuwen Fu, Giancarlo Fortino, Wenfeng Li, Pasquale Pace, and Yongsheng Yang. 2019. WSNs-assisted opportunistic network for low-latency message forwarding in sparse settings. *Future Generation Computer Systems* 91 (2019), 223–237.
[19] Anshul Gaur, Abhishek Singh, Ashok Kumar, Kishor S Kulkarni, Sayantani Lala, Kamal Kapoor, Vishal Srivastava, Anuj Kumar, and Subhas Chandra Mukhopadhyay. 2019. Fire Sensing Technologies: A Review. *IEEE Sensors Journal* 19, 9 (2019), 3191–3202.
[20] Nargess Ghassempour, Ju Jia Zou, and Yaping He. 2018. A SIFT-Based Forest Fire Detection Framework Using Static Images. In *2018 12th International Conference on Signal Processing and Communication Systems (ICSPCS)*. IEEE, 1–7.
[21] JiSeong Han, GwangSu Kim, ChanSeo Lee, YeongKwang Han, Ung Hwang, and SungHwan Kim. 2019. Predictive Models of Fire via Deep learning Exploiting Colorific Variation. In *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 579–581.
[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
[23] Pan Hu, Junha Im, Zain Asgar, and Sachin Katti. 2020. Starfish: resilient image compression for AIoT cameras. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 395–408.
[24] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv preprint arXiv:1602.07360* (2016).
[25] Vittorio P Illiano, Andrea Paudice, Luis Muñoz-González, and Emil C Lupu. 2018. Determining Resilience Gains From Anomaly Detection for Event Integrity in Wireless Sensor Networks. *ACM Transactions on Sensor Networks (TOSN)* 14, 1 (2018), 5.
[26] Riley Jackson, Jonathan Gresl, and Ramon Lawrence. 2021. Efficient External Sorting for Memory-Constrained Embedded Devices with Flash Memory. *ACM Transactions on Embedded Computing Systems (TECS)* 20, 4 (2021), 1–21.
[27] Mian Ahmad Jan, Priyadarsi Nanda, Xiangjian He, and Ren Ping Liu. 2018. A Sybil attack detection scheme for a forest wildfire monitoring application. *Future Generation Computer Systems* 80 (2018), 613–626.
[28] Benamar Kadri, Benamar Bouyeddou, and Djillali Moussaoui. 2018. Early Fire Detection System Using Wireless Sensor Networks. In *2018 International Conference on Applied Smart Systems (ICASS)*. IEEE, 1–4.

[29] Amer Kajmakovic, Konrad Diwold, Nermin Kajtazovic, Robert Zupanc, and Georg Macher. 2019. Flexible soft error mitigation strategy for memories in mixed-critical systems. In *2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*. IEEE, 440–445.

[30] Harkiran Kaur and Sandeep K Sood. 2019. Fog-assisted IoT-enabled scalable network infrastructure for wildfire surveillance. *Journal of Network and Computer Applications* 144 (2019), 171–183.

[31] Ali Khan. 2020. Dataset for Forest Fire Detection. https://doi.org/10.17632/GJMR63RZ2R.1

[32] Michal Kochláň, Michal Hodoň, Lukáš Čechovič, Ján Kapitulík, and Matúš Jurečka. 2014. WSN for traffic monitoring using Raspberry Pi board. In *2014 Federated Conference on Computer Science and Information Systems*. IEEE, 1023–1026.

[33] Murat Koyuncu, Adnan Yazici, Muhsin Civelek, Ahmet Cosar, and Mustafa Sert. 2018. Visual and Auditory Data Fusion for Energy-Efficient and Improved Object Recognition in Wireless Multimedia Sensor Networks. *IEEE Sensors Journal* 19, 5 (2018), 1839–1849.

[34] Alex Krizhevsky. 2014. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997* (2014).

[35] R Kumar and M Pallikonda Rajasekaran. 2016. An IoT based patient monitoring system using raspberry Pi. In *2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16)*. IEEE, 1–4.

[36] Wonjae Lee, Seonghyun Kim, Yong-Tae Lee, Hyun-Woo Lee, and Min Choi. 2017. Deep neural networks for wild fire detection with unmanned aerial vehicle. In *2017 IEEE international conference on consumer electronics (ICCE)*. IEEE, 252–253.

[37] Jian Li, Baoqiang Yan, Mingjiang Zhang, Jianzhong Zhang, Baoquan Jin, Yu Wang, and Dong Wang. 2019. Long-Range Raman Distributed Fiber Temperature Sensor With Early Warning Model for Fire Detection and Prevention. *IEEE Sensors Journal* 19, 10 (2019), 3711–3717.

[38] Yonggang Li, Chunping Liu, Yi Ji, Shengrong Gong, and Haibao Xu. 2020. Spatio-Temporal Deep Residual Network with Hierarchical Attentions for Video Event Recognition. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 16, 2s (2020), 1–21.

[39] Jiawei Liu, Zheng-Jun Zha, QI Tian, Dong Liu, Ting Yao, Qiang Ling, and Tao Mei. 2016. Multi-scale triplet cnn for person re-identification. In *Proceedings of the 24th ACM international conference on Multimedia*. ACM, 192–196.

[40] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*. 116–131.

[41] Antonio Molina-Pico, David Cuesta-Frau, Alvaro Araujo, Javier Alejandre, and Alba Rozas. 2016. Forest monitoring and wildland early fire detection by a hierarchical wireless sensor network. *Journal of Sensors* 2016 (2016).

[42] Lalatendu Muduli, Devi Prasad Mishra, and Prasanta K Jana. 2018. Application of wireless sensor network for environmental monitoring in underground coal mines: a systematic review. *Journal of Network and Computer Applications* 106 (2018), 48–67.

[43] Khan Muhammad, Jamil Ahmad, and Sung Wook Baik. 2018. Early fire detection using convolutional neural networks during surveillance for effective disaster management. *Neurocomputing* 288 (2018), 30–42.

[44] Khan Muhammad, Jamil Ahmad, Zhihan Lv, Paolo Bellavista, Po Yang, and Sung Wook Baik. 2018. Efficient deep CNN-based fire detection and localization in video surveillance applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49, 7 (2018), 1419–1434.

[45] Khan Muhammad, Jaime Lloret, and Sung Wook Baik. 2019. Intelligent and energy-efficient data prioritization in green smart cities: Current challenges and future directions. *IEEE Communications Magazine* 57, 2 (2019), 60–65.

[46] Khoa Anh Ngo, Trong Thua Huynh, and De Thu Huynh. 2018. Simulation Wireless Sensor Networks in Castalia. In *Proceedings of the 2018 International Conference on Intelligent Information Technology*. ACM, 39–44.

[47] Linh Nguyen and Hoc T Nguyen. 2020. Mobility based network lifetime in wireless sensor networks: A review. *Computer Networks* (2020), 107236.

[48] OMNeT++. [n.d.]. OMNeT++, Discrete Event Simulator. https://omnetpp.org/.

[49] Athanasia Panousopoulou, Mikel Azkune, and Panagiotis Tsakalides. 2016. Feature selection for performance characterization in multi-hop wireless sensor networks. *Ad Hoc Networks* 49 (2016), 70–89.

[50] Andy Parsons, Russell Goldman, and Matthew Abbott. 2020. Why the Fires in Australia Are So Bad. https://www.nytimes.com/2020/01/01/world/australia/fires.html

[51] Partha Pratim Ray, Dinesh Dash, and Debashis De. 2019. Internet of things-based real-time model study on e-healthcare: Device, message service and dew computing. *Computer Networks* 149 (2019), 226–239.

[52] Ramon Sanchez-Iborra and Antonio F Skarmeta. 2020. TinyML-Enabled Frugal Smart Objects: Challenges and Opportunities. *IEEE Circuits and Systems Magazine* 20, 3 (2020), 4–18.

[53] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4510–4520.

[54] Vikrant Sharma, RB Patel, HS Bhadauria, and D Prasad. 2016. Deployment schemes in wireless sensor network to achieve blanket coverage in large-scale open area: A review. *Egyptian Informatics Journal* 17, 1 (2016), 45–56.

[55] Lauro Snidaro, Ingrid Visentini, and Gian Luca Foresti. 2012. Fusing multiple video sensors for surveillance. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 8, 1 (2012), 1–18.

[56] Kethavath Srinivas and Mohit Dua. 2019. Fog Computing and Deep CNN Based Efficient Approach to Early Forest Fire Detection with Unmanned Aerial Vehicles. In *International Conference on Inventive Computation Technologies*. Springer, 646–652.

[57] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.

[58] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. 2019. Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2820–2828.

[59] Jasmine Taylor-Coleman. 2018. *Wildfires around the world: The photos that explain the flames.*

[60] Tom Toulouse, Lucile Rossi, Antoine Campana, Turgay Celik, and Moulay A Akhloufi. 2017. Computer vision for wildfire research: An evolving image dataset for processing and analysis. *Fire Safety Journal* 92 (2017), 188–194.

[61] FIRE LOOKOUT TOWERS. 2018. B.C. FIRE LOOKOUT TOWERS. http://www.firelookout.com/bc.html.

[62] András Varga and Rudolf Hornig. 2008. An overview of the OMNeT++ simulation environment. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*. ICST (Institute for Computer Sciences, Social-Informatics, 60.

[63] Denghui Wang, Jian Liu, Dezhong Yao, and IEEE Member. 2020. An Energy-efficient Distributed Adaptive Cooperative Routing Based on Reinforcement Learning in Wireless Multimedia Sensor Networks. *Computer Networks* (2020), 107313.

[64] Linnan Wang, Yi Yang, Renqiang Min, and Srimat Chakradhar. 2017. Accelerating deep neural network training with inconsistent stochastic gradient descent. *Neural Networks* 93 (2017), 219–229.

[65] Wonjae Lee, Seonghyun Kim, Yong-Tae Lee, Hyun-Woo Lee, and Min Choi. 2017. Deep neural networks for wild fire detection with unmanned aerial vehicle. In *2017 IEEE International Conference on Consumer Electronics (ICCE)*. 252–253. https://doi.org/10.1109/ICCE.2017.7889305

[66] Lingxi Xie, Jingdong Wang, Zhen Wei, Meng Wang, and Qi Tian. 2016. Disturblabel: Regularizing cnn on the loss layer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4753–4762.

[67] Yi-Han Xu, Qiu-Ya Sun, and Yu-Tong Xiao. 2018. An Environmentally Aware Scheme of Wireless Sensor Networks for Forest Fire Monitoring and Detection. *Future Internet* 10, 10 (2018), 102.

[68] Chi Yuan, Zhixiang Liu, and Youmin Zhang. 2017. Fire detection using infrared images for UAV-based forest fire surveillance. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 567–572.