future internet

Article

Forest Fire Detection and Notification Method Based on AI and IoT Approaches

Kuldoshbay Avazov [1], An Eui Hyun [1], Alabdulwahab Abrar Sami S [1], Azizbek Khaitov [2], Akmalbek Bobomirzaevich Abdusalomov [1] and Young Im Cho [1,*]

[1] Department of Computer Engineering, Gachon University, Sujeong-Gu, Seongnam-Si 461-701, Republic of Korea

[2] "DIGITAL FINANCE" Center for Incubation and Acceleration, Tashkent Institute of Finance, Tashkent 100000, Uzbekistan

* Correspondence: yicho@gachon.ac.kr

Abstract: There is a high risk of bushfire in spring and autumn, when the air is dry. Do not bring any flammable substances, such as matches or cigarettes. Cooking or wood fires are permitted only in designated areas. These are some of the regulations that are enforced when hiking or going to a vegetated forest. However, humans tend to disobey or disregard guidelines and the law. Therefore, to preemptively stop people from accidentally starting a fire, we created a technique that will allow early fire detection and classification to ensure the utmost safety of the living things in the forest. Some relevant studies on forest fire detection have been conducted in the past few years. However, there are still insufficient studies on early fire detection and notification systems for monitoring fire disasters in real time using advanced approaches. Therefore, we came up with a solution using the convergence of the Internet of Things (IoT) and You Only Look Once Version 5 (YOLOv5). The experimental results show that IoT devices were able to validate some of the falsely detected fires or undetected fires that YOLOv5 reported. This report is recorded and sent to the fire department for further verification and validation. Finally, we compared the performance of our method with those of recently reported fire detection approaches employing widely used performance matrices to test the achieved fire classification results.

Keywords: bushfire; fire detection; forest environment; YOLOv5; fire-like lights

check for updates

Citation: Avazov, K.; Hyun, A.E.; Sami S, A.A.; Khaitov, A.; Abdusalomov, A.B.; Cho, Y.I. Forest Fire Detection and Notification Method Based on AI and IoT Approaches. *Future Internet* **2023**, *15*, 61. https://doi.org/10.3390/fi15020061

Academic Editor: Paolo Bellavista

Received: 29 November 2022
Revised: 19 January 2023
Accepted: 29 January 2023
Published: 31 January 2023

1. Introduction

Given its direct impact on public safety and the environment, early fire detection is a difficult but crucial problem. To avoid harm and property damage, advanced technology requires appropriate methods for detecting fires as soon as possible [1]. According to UNEP, "Wildfires are becoming more intense and more frequent, ravaging communities and ecosystems in their path" [2]. Wildfires continue to burn for days without swift action, resulting in a climate crisis and the loss of lives. Due to the climate crisis, the world is starting to face anonymous fluctuations in water levels, changes in temperature, and the extinction of some protected animals, which will affect the balance of life in the future. Therefore, we must take wildfire problems seriously before they become catastrophic. Installing early fire detection in the forest and an automatic notification system to notify the fire department can reduce countless problems.

It has always been a challenge to control fires on a global scale. In 2019, there were 40,030 fires in South Korea, resulting in 284 deaths and 2219 injuries, according to the Korean National Fire Agency. In addition, property damage totaled KRW 2.2 billion as a result of 110 fires and 0.8 fire-related deaths daily. Two significant Korean cities experienced fires in 2020 that resulted in the deaths of over 50 people in each location. A 33-story tower block building in Ulsan burned down, and a warehouse fire broke out in Incheon [3].

However, these incidents are just in Korea. Imagine having a wildfire that is the size of a country continuing on for weeks. According to UNEP, the intensity and frequency of wildfires are increasing, wreaking havoc on the ecosystems and populations they pass through. These wildfires happen for many reasons. It is either due to bushfires that are caused naturally or to human errors. If bushfires are detected early, the spreading of wildfires can be prevented. Furthermore, if we can stop people from starting a fire as they light it, we can prevent further incidents. In our research, we found some studies on fire alarms that are installed in the forest. However, we found some challenges they may face in our own study.

○ Sensors are one of the widely used techniques to determine whether there is a fire or not. We found that just using a smoke detection sensor can lead to false alarms because some of the smells can come from different places or from someone smoking.

○ Remote cameras are used in some of the other fire detection systems to determine whether there is a fire or not. This kind of surveillance requires human employees to continuously monitor the cameras, which can sometimes lead to the employees falling asleep.

○ R-CNN-based approaches are also efficiently used to identify and eliminate fire catastrophes. However, this method might also make errors sometimes and incorrectly classify candidate fire regions as real fires. Therefore, they may also lead to false alarms.

Identifying these previous errors made by other research and companies gave us innovative ideas. When humans eliminate one of their sensory features, other sensory features become more enhanced. For example, when you blindfold someone, other senses in their body enhance. This realization came through serendipity. Having walked by an air purifier multiple times after a workout, the air purifier worked extra hard. Then, we realized the machine could not see but could smell us first. Our human body can smell things first before it can see. Another example is when leaving a pot of soup boiling for too long, you can smell the burnt scent before you can see it. However, we humans double-check to see what is burning. We wanted our ideas to do both so we converged the ideas from all the research into one product that would prevent wildfires from happening.

The rest of this manuscript is structured into five more sections. Section 2 reviews the literature on traditional and deep learning methods used to identify particular fire regions. Section 3 includes a description of how the fire detection system works. Section 4 presents the results of the experiment. Section 5 highlights certain limitations of the proposed method. Lastly, Section 6 covers the conclusion of the paper and the discussion of the proposed method's future directions.

2. Related Work

Forest fire detection technologies can be divided into two main categories: machine learning, deep learning, and computer vision methods, and sensor-based methods. In recent studies, it was observed that object-based detection in the industry has gained popularity from deep learning [4]. The most common approaches to detect objects in deep learning are image-based convolutional neural networks (CNNs) [5], fully convolutional networks [6], spatio-spectral deep neural networks [7], and faster R-CNNs [8].

2.1. Forest Fire Detection Using the Machine-Learning Approach

Toulouse et al. [9] developed a new method to detect the geometrical characteristics of a fire depending on its position, surface, and length. In this study, the fire color was categorized into pixels. Moreover, the pixels were classified based on the average intensity of the non-refractory images. Jian et al. [10] introduced an upgraded boundary-detection operator, and their model used a multistep operation. However, the abstraction of the model was only applied to simple and stable fire and flame images. Researchers worldwide have used a new algorithm based on FFT to detect fires. Turgay [11] developed a real-time fire detector that combined background and foreground color frames. However, the real-time color-based program did not provide a better output because of the smoke and shadow. In [12], based on the dynamic textures of smoke and flame, the fire was detected using dynamic systems (LDSs).

2.2. Forest Fire Detection Based on the Deep Learning Approach

Recently, deep-learning-based object detection has become more popular than sensor-based object detection. In [13], Park et al. proposed the ELSTIC-YOLOv3 model to detect small objects and, in the same study, they mentioned the dynamic fire tube, which is a characteristic of fire. The research team in [14] proposed a CNN-based model with an average precision accuracy of 98.7%. Furthermore, in [15–22], an approach to improve fire detection technology was presented. Fast region-based convolutional neural networks (R-CNN) for object detection utilized high-quality region proposals created by the Region Proposal Network (RPN), which was trained in the end-to-end process [23]. Liu, W. et al. introduced a single-shot detector (SSD) for multiple categories that was faster and significantly more accurate than the previous traditional works for single-shot detectors (YOLO), as they claimed in their paper [24]. Fast YOLO employs a neural network with 9 as opposed to 24 convolutional layers, and fewer filters in those layers. The only difference between YOLO's and Fast YOLO's training and testing parameters is the network's size [25–27]. Automated deep learning and IoT algorithms are becoming very active and attractive approaches in agriculture and industrial appliances because of accurate detection and predictions [28–30]. In CNN, the challenge is to achieve high accuracy by training with a large dataset, which is an expensive process. To overcome this problem, we prepared a large dataset publicly available on the Internet.

In summary, the main contributions of this study are as follows:

- The most common method to detect fire is the smoke alarm, which can commonly be found in a lot of projects or even at a store. However, having only smoke sensors can cause false alarms.
- Remote cameras are probably the most common methods used by park rangers. However, these require 24/7 human interactions. This can lead to human errors by falling asleep during a shift.
- R-CNN is now advancing more and more and is being introduced as a product. This allows real-time surveillance and it can detect whether there is a fire or not. This can be very useful in furthering our society in the future.
- The goal of this research is to converge Internet of Things devices with AI methods in the field of global wildfire prevention. Therefore, our research saves lives and government property.

3. Fire Detection Approach

3.1. Proposed High-Level System Architecture

An overview of the suggested method of forest fire detection and notification system is given in this subsection. A small error might result in unexpected wildfires, which can quickly lead to a catastrophic situation [31]. The major goal of this study is to develop a novel fire detection method based on IoT and AI that can reduce wildfires and a host of other issues. The system, which is seen in Figure 1, is designed to keep an eye on the risk of wildfires, accurately detect even the tiniest sparks of flame, and alert the fire brigade when necessary.

It is hard to detect fires in the forest until it is too late. This can lead to the loss of lives that are within the wildfire region. Our goal is to identify the fire and the location of the fire in order to safely evacuate the people within a 1-mile radius by sending a notification alarm through their phone, as we explained in detail in the refs. [32,33].

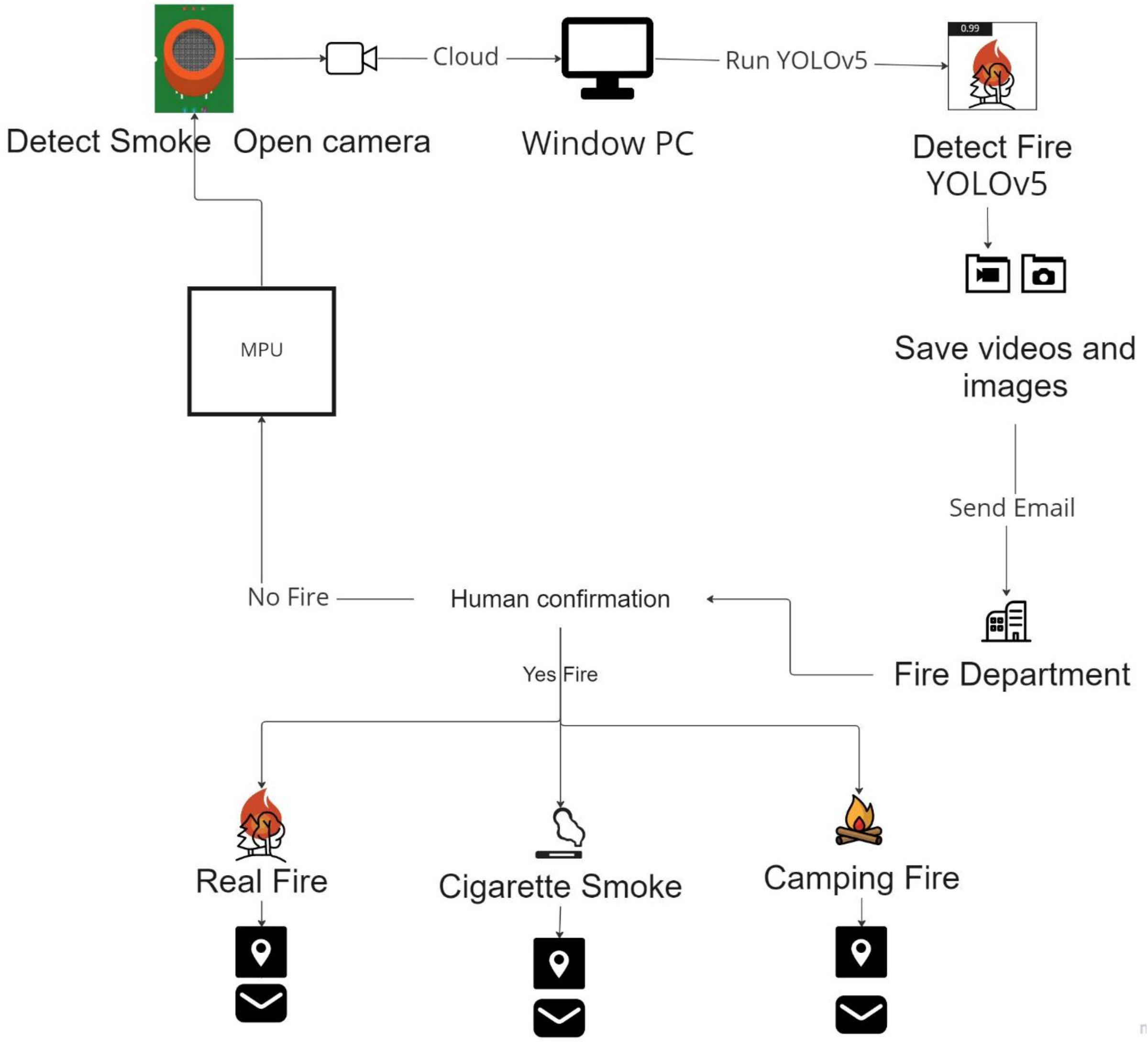

Figure 1. Overall flow chart of the system.

3.2. IoT Devices and Hardware Requirements

The device we used to perform our test was a Raspberry Pi 4 due to the size, portability, and accessibility to all the pins that are available for the sensors required for input. Furthermore, Raspberry has wire and wireless extensive variety for network accessibility. Accessing the network gave us a huge advantage by quickly sending data to the cloud for our main computer to identify whether it is a real fire or not. However, before the fire was detected, our MQ-2 flammable gas and smoke sensor was used to detect gas. The MQ-2 flammable gas and smoke sensor was connected to the Raspberry Pi, which was also connected to the breadboard. We connected the Raspberry Pi 4 pins to the breadboard and routed to the MQ-2 flammable gas and smoke sensor. Lastly, we connected the camera to detect the fire visually, which was easily accomplished with a USB cable. This unit was placed in a casing to protect it from being weathered. This unit would be installed on a fabricated tree. In Korea, you can see these trees when climbing a mountain, etc. These are cellphone towers hidden in the mountains. We can use these fabricated cellphone tree towers to install all of our electronics and power them as well. Once this unit detects the smoke via the MQ-2 flammable gas and smoke sensor, it then uses the camera to take multiple shots. These data are thensent off to the cloud for the main computer to analyze using AI.

Working with devices and sensors and testing out beta products, Raspberry Pi 4 or Arduino are used due to the easy accessibility to the pins and sensors. However, this can be bypassed by using a desktop Windows 10 Pro and its GPU to render video quality, and using the Raspberry Pi 4 or Arduino just for the smoke sensors.

Listed below are the hardware requirements:

- Raspberry Pi 4 or Arduino;
- Raspberry Pi 4 power cable;
- Raspberry Pi 4 Internet cable;
- Solid-state drive (SSD) or hard drive;
- Internet access;
- Desktop Windows 10 Pro (optional/recommended);
- MQ-2 flammable gas & smoke sensor;
- Pin wires;
- Breadboard;
- Camera.

3.3. IoT Devices and Sensor Models

Raspberry Pi 4 has its own OS that is pre-installed with anIntegrated Development Environment (IDE). This allows the user to conveniently use the python problem. Furthermore, the Raspberry Pi 4 is equipped with a set of pins, which you can easily access to test out sensors, etc. We tested the MQ-2 flammable gas and smoke sensor because it was the only thing that was available to us. However, there are a lot more similar sensors that can be used depending on the situation.

Sensors similar to MQ-2 are as follows:

- MQ-2: measures methane, butane, liquefied petroleum gas (LPG), smoke;
- MQ-3: alcohol, ethanol, smoke;
- MQ-4: methane, compressed natural gas (CNG);
- MQ-5: natural gas, LPG;
- MQ-6: LPG, butane;
- MQ-7: carbon monoxide;
- MQ-8: hydrogen gas;
- MQ-9: carbon monoxide, flammable gasses;
- MQ131: ozone;
- MQ135: air quality;
- MQ136: hydrogen sulphide gas;
- MQ137: ammonia;
- MQ138: benzene, toluene, alcohol, propane, formaldehyde gas, hydrogen;
- MQ214: methane, natural gas;
- MQ216: natural gas, coal gas;
- MQ303A: alcohol, ethanol, smoke;
- MQ306A: LPG, butane;
- MQ307A: carbon monoxide;
- MQ309A: carbon monoxide, flammable gas.

We chose to go with the MQ-2 not only because it was the only thing that was available to us, but also because it measures methane, butane, LPG, and smoke.

3.4. Dataset

Dataset collection and generation processes were obtained from the Robmarkcole and Glenn-Jocher databases, as shown in Table 1 [34,35]. For the experiment, we used a secondary dataset that contains a collection of indoor and outdoor fire images [36]. The dataset we used provides two folders: train and val. The full fire image dataset was split into training (75%) and test (25%) sets. The train folder was for the training images, and the val folder was for image validation. Both folders contained a set of unlabeled images as well as labeled images to train, test, and validate the model. Additionally, since we did not have wildfires, we used YouTube videos that had different shapes and types of fire to test our model and check its accuracy.

Table 1. Distribution of fire images in the dataset.

Dataset	Training Images	Testing Images	Total
Robmarkcole	1155	337	1492
Glenn-Jocher	1500	128	1628

3.4.1. Fire Identification

The suggested model may be thought of as a combination of all three existing ideas, whether it is from research or a product. As introduced in Section 1, people have made many different efforts to prevent wildfires from happening or spreading. We converged all three existing ideas into one, where we can use smell and sight by converging IoT and AI. IoT device refers to all the devices that are connected to the Raspberry Pi 4 listed in Section 3.2. Section 3.3 contains an MQ-2 flammable gas and smoke sensor that immediately activates the camera, which is operating and looking for fires whether or not the MQ-2 detects any smoke, when it detects any gas or strange odor. AI is then used to determine if a fire is present. AI is generally explained as artificial intelligence but, in this manuscript, it refers to the technique of YOLOv5. YOLOv5 is used to train an image and label it into classes to be identified through images. Moreover, the YOLOv5 correctly identifies the fire using its algorithm. Once the fire is correctly identified, it records and sends those highlighted images to the fire department. Then, the fire department can verify and validate if the recording and the images are an actual fire or just a false alarm. Once it is identified as a fire, the fire department hits confirm on their screen to send an alarm notification within a 1-mile radius of the recording. The notification sends a nearby emergency route to quickly guide anybody within that 1-mile radius away from harm. However, if it happens to be a camper trying to cook with fire, or if it is someone trying to smoke in the forest, a notification is sent within the 1-mile radius reminding the people with access to fire of the penalty of lighting a fire inside a forest and suggesting to turn off any fire hazard items.

3.4.2. YOLOv5

YOLOv5 is a recently released CNN that distinguishes between static and moving objects in real time, with notable performance and good accuracy. This model processes the full image region using a single neural network, divides it into different components, and then predicts the candidate-bounding boxes and probabilities for each component. The YOLOv5 network is an evolution of the YOLOv1-YOLOv4 network and is composed of three architectures: the head, which generates YOLO layers for multi-scale prediction, the neck, for enhancing information flow based on the path aggregation network (PANet), and the backbone based on cross-stage partial (CSP) integrated into the Darknet [37,38]. The data are given to CSPDarknet for feature extraction before being transferred to PANet for feature fusion. As seen in Figure 2, the YOLO layer uses three independent feature maps to produce detection results (class, score, location, and size).

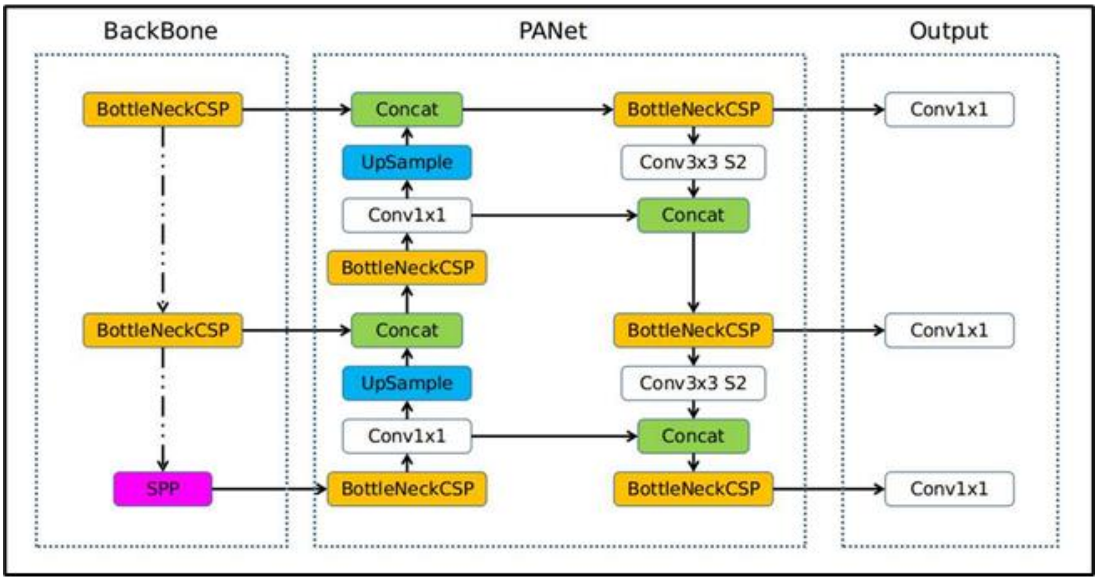

Figure 2. YOLOv5 network structure [39].

The YOLOv5 network is used as a backbone in Figure 2 to extract significant and practical characteristics from the input frame sequences with CSP. The YOLOv5 network builds feature pyramids using the neck model to enable models to successfully generalize in terms of object scale. As a result, it is simpler to recognize the same object at various scales and dimensions [40].

3.4.3. YOLOv5 Series

In the YOLOv5 series, there are five different models, ranging in size from the smallest and fastest model, the YOLOv5n nano, to the largest model, the YOLOv5x extra large [41]. The YOLOv5n nano model is good for mobile solutions, which is very efficient. YOLOv5s, which stands for small, is appropriate when using it with a CPU. YOLOv5m, which stands for medium-sized model, is fairly balanced when it comes to speed and accuracy. YOLOv5l, where l stands for large-sized model, is good where smaller items need to be found. The largest of these five models is called YOLOv5x. YOLOv5x contains more parameters than the others, but it is slower to execute.

As seen in Figure 3, it is clear that the YOLOv5x is superior among all five models. However, the pre-trained image data were annotated and trained using YOLOv5s. Even though someone already pre-trained the data, we decided to retrain it with YOLOv5x. We believe the previous contributor completed the pre-training of the dataset when the s model was used.

3.4.4. Fire Department

Our research team is not able to have a big influence on how the fire department operates. Therefore, this part of Section 3.4.4 is theoretical but possible with a simple collaboration with the fire department. Once the fire is confirmed by the YOLOv5, it then records the footage and sends it to the fire department by email using the Simple Mail Transfer Protocol (SMTP). SMTP is a very simple protocol; it uses email and password that originally come from Google account. In the code, we apply the SMTP method and write the receiver's email address and what we want to send. We can choose a file, folder, and type of image or video to be sent to the receiver. After the footage is sent, it is reviewed by the fire department, which then verifies and validates it, and acts accordingly. If the fire department confirms the video as hazardous fire, they send a notification alarm within the one-mile radius of where the footage was taken. If the fire is just a smoker or a camper trying to light a fire, the fire department sends a notification within the one-mile radius of where the footage was taken, notifying nearby hikers about the penalty of lighting a fire in the forest. All the messages are sent through the Wireless Emergency Alert (WEA) that all emergency stations have in Korea [32,33].

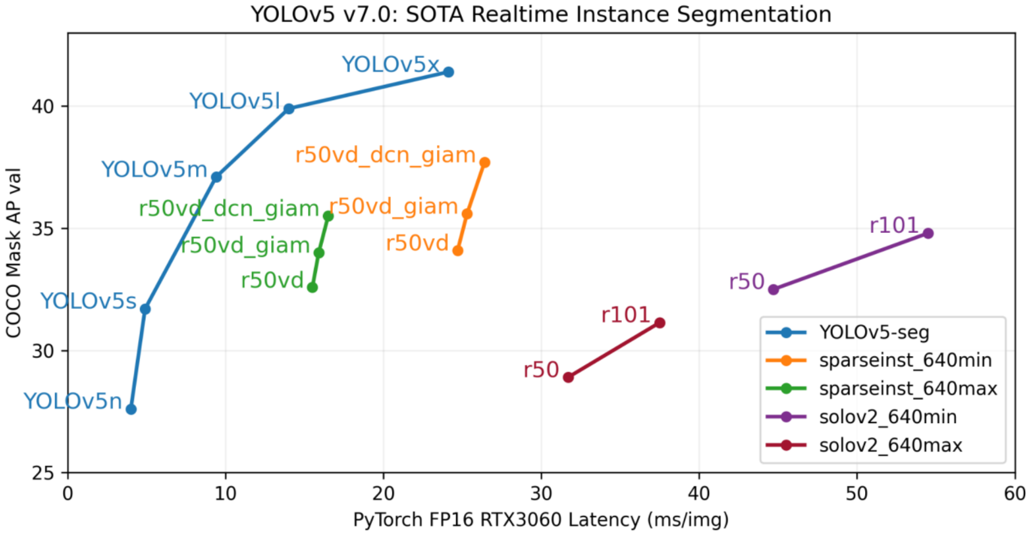

Figure 3. YOLOv5 SOTA Real-Time Instance Segmentation.

4. Experimental Results and Discussions

4.1. Environment

We implemented and tested our proposed idea configuration using two computers. One computer is Raspberry Pi 4 model B, and another computer is a personal computer that uses Windows OS, as seen in Table 2.

Table 2. Raspberry Pi 4 Model B specifications and Windows Desktop specifications.

Hardware	Detailed Specifications
Processor	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memory	1GB, 2GB, 4GB or 8GB LPDDR4
Connectivity	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE Gigabit Ethernet 2 USB 3.0 ports; 2 USB 2.0 ports.
GPIO	40 pin GPIO header
Camera	Webcam logitech c920
Storage	SSD 120GB
Operating System	Raspberry Pi OS
Power	5V 3A 1.5M
Processor	Intel i7 8700K 3.70GHz
Memory	32GB DDR4
Local Area Network	Internal port—10/100 Mbps External port—10/100 Mbps
GPU	1080 ASUS
Motherboard	TUF Z390-PLUS GAMING
Storage	500 GB M.2, 1TB SSD, 4TB Hard Drive
Operating System	Windows 10 Pro
Power	SuperFlower SF-1000 Watt

We used the Raspberry Pi 4 Model B to detect the smell and sight using the MQ-2 flammable gas and smoke sensor and a Logitech C920 webcam. Due to its portability, we were able to put this device anywhere to detect smoke and fire. Therefore, it could also be installed in the forest. After detecting the fire using the MQ-2 flammable gas and smoke sensor, it records it with the Logitech C920 webcam and sends it to the personal computer in Table 2. Then, with its high performance, it uses the Anaconda console to execute the video and automatically detects if there is a fire through YOLOv5. Once the fire is confirmed, the video is sent to the fire department for further action.

4.2. Proof of Concept Experimentation

After the MQ-2 flammable gas and smoke sensor detected the smoke, the camera immediately recorded the footage to be sent to the personal computer, as given in Table 2. As shown in Figures 4 and 5, our approach performed well and quickly identified fire accidents, even multiple fires and flames, in both indoor and outdoor environments.

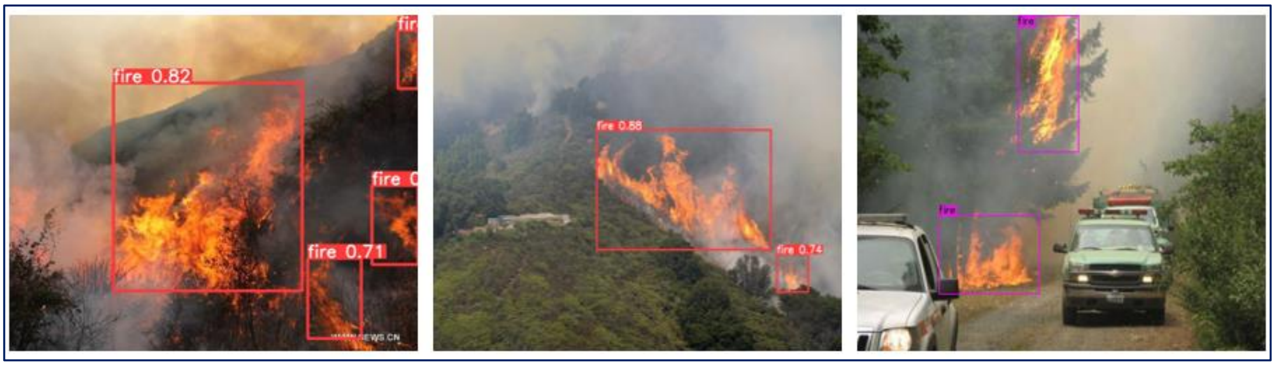

Figure 4. Visible experiment results in forest fire scenes.

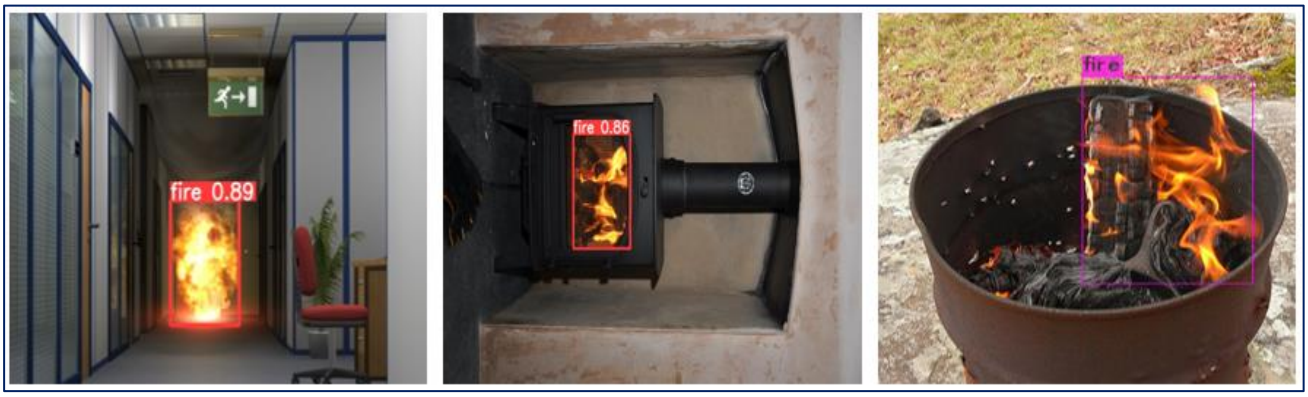

Figure 5. Visible experiment results in indoor and outdoor environments with fire scenes.

As seen in Figure 4, the confidence level was mostly above 80%. This is because we had to retrain the data to obtain the better output result. We retrained the data to YOLOv5x and set the batch size to 16 and the epoch to 3 at first. Then, we observed the accuracy level and it was average, as it can be seen here. There was a big difference between YOLOv5s and YOLOv5x. If we were to keep the pre-trained data, we would obtain approximately 30%. However, as it can be seen, there was a big jump in confidence level.

Figure 5 shows that, after increasing the epoch size to 10, the confidence level went up a little. This research shows and proves that the more you constantly train the data, the better your outcome is. These fire detection methods are conducted through some calculations. The IoU or Jaccard Index is used to determine if the prediction is correct to an object or not. It is defined as the predicted box intersection divided by the actual box intersection divided by their union [42,43]. In other words, it is an effective metric for evaluating detection results, and is defined as the area of overlap between the detected fire region and the ground truth divided by the area of union between the detected fire region and the ground truth (1):

$$IoU = \frac{groundTruth \cap prediction}{groundTruth \cup prediction} \tag{1}$$

The FM score and IoU value range is between 0 and 1, where these metric scores reach their best values at 1.

To understand mAP, the precision and recall calculation is needed, as we detailed in previous research [44]. TP stands for "true positives", FP stands for "false positives", and FN stands for "false negatives". Precision, or the percentage of true positive predictions among all positive forecasts, is the positive predictive value. The average precision and recall rates of the fire detection techniques can be calculated using the following equations:

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} \\ Recall &= \frac{TP}{TP+FN} \end{aligned} \tag{2}$$

We need to determine the AP for each class in order to calculate the mAP. However, we only have one class. A PR (precision–recall) curve is obtained by plotting these precision and recall values. Average precision is the region beneath the PR curve (AP). The PR curve has a zig-zag shape, as recall rises steadily while precision generally falls with intermittent increases. The AP, which in VOC 2007 was defined as the mean of precision values at a set of 11 equally spaced recall levels [0, 0.1, . . . , 1] (0 to 1 at a step size of 0.1), describes the shape of the precision–recall curve rather than the AUC. However, in VOC 2010, the computation of the AP changed so, instead of just taking 11, we take all points into account [45].

$$AP = \frac{1}{11} \sum_{r \in (0,0.1\ldots,1)} p_{interp(r)} \tag{3}$$

The greatest precision measured for a method for which the corresponding recall exceeds r is used to interpolate the precision at each recall level r.

Using quantitative and qualitative performance data, we rank the durability of previously introduced publications employing the suggested approach in many classifications, as shown in Table 3. In accordance with the scores, the proposed approach worsened when far- and small-region flames occurred, but it was able to successfully distinguish between fake or non-fire sceneries and actual fires with a quick processing time performance.

Table 3. Employing many characteristics to evaluate the effectiveness of fire detection.

Criterion	**Improved YOLOv3 [26]**	**Improved YOLOv4 [27]**	**Improved YOLOv4 BVI [33]**	**Proposed Method**
Scene Independence	standard	robust	standard	robust
Object Independence	standard	robust	robust	standard
Robust to Noise	powerless	robust	standard	robust
Robust to Color	standard	standard	powerless	robust
Small Fire Detection	robust	standard	robust	powerless
Multiple Fire Identification	standard	powerless	powerless	robust
Processing Time	powerless	standard	robust	robust

5. Limitations

Our research team faced a lot of limitations due to the time it takes to train the data. Therefore, we were not able to run enough training sessions because we were limited on computers and we needed to conduct other research to improve the accuracy. Therefore, some of the images came out as having a low confidence level. However, those images were outputting small fire images (Figure 6). Furthermore, YOLOv5 would mistakenly recognize red shirts or red blinking lights as fires. We are creating a sizable fire image dataset utilizing data augmentation methodologies that includes fire and non-fire photos for model training and testing in order to efficiently identify the target data and address the mentioned difficulties.

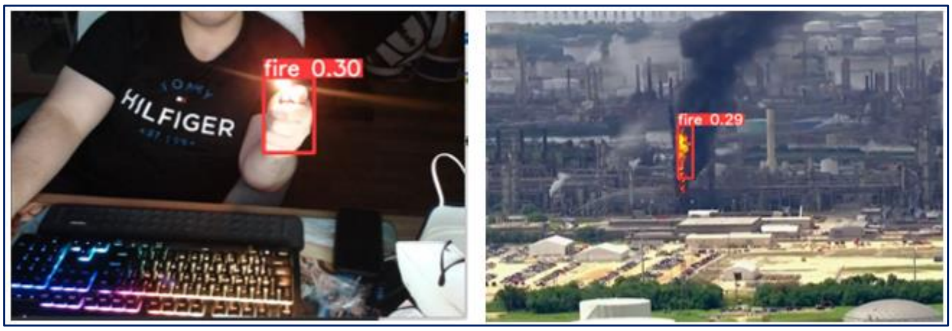

Figure 6. Small size fire region detected images.

The Internet of Things (IoT) has evolved into a free interchange of useful information between various real-world devices. Several technological issues must be overcome in order to improve fire detection and warning accuracy, which can be separated into five major issues: security and privacy, storage and cloud computing, energy, communication, and compatibility and standardization [46].

6. Conclusions

In this paper, we have introduced our new technology to reduce wildfires by using AI and IoT devices and sensors. Therefore, with our system, we believe that the proposed system can be effectively used to end the rapid increase in the world climate crisis and the loss of lives. The system can be installed in forests and start detecting smoke to let the AI model detect the exact fire location and notify the fire department to disallow the fire to continue for days. Finally, we hope that this technology will be effective in other countries to prevent wildfires worldwide.

Recent studies have shown that, in order to promote safety in our daily lives, it is critical to quickly identify fire accidents in their early phases. As a result, we hope to carry out more research in this area and enhance our findings. Our goal is to identify fire occurrences in real time with fewer false positives using the YOLOv6 and YOLACT models. Future goals include improving the accuracy of the approach and addressing wrongly detected situations in the same color cases with fire regions. Using 3D CNN and 3D U-Net in the IoT environment, we intend to create a compact model with reliable fire-detection performance and without communication issues.

Author Contributions: This manuscript was designed and written by K.A. and A.E.H. A.B.A. conceived the main idea of this study. A.A.S.S. wrote the program and conducted all experiments. A.K. and Y.I.C. supervised the study and contributed to the analysis and discussion of the algorithm and the experimental results. All authors have read and agreed to the published version of the manuscript.

Funding: This study was funded by Korea Agency for Technology and Standards in 2022, project numbers are K_G012002073401, K_G012002234001 and by the Gachon University research fund of 2021 (GCU-202106340001).

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to express their sincere gratitude and appreciation to the supervisor, Young Im Cho (Gachon University), for her support, comments, remarks, and engagement over the period in which this manuscript was written. Moreover, the authors would like to thank the editor and anonymous referees for their constructive comments toward improving the contents and presentation of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Korea Forest Service 2019, Korea Forest Service Website, Korean Government. Available online: https://english.forest.go.kr (accessed on 10 November 2022).
2. Nairobi 2022, Unified Nation Enviroment Programme Website. Available online: https://www.unep.org (accessed on 10 November 2022).
3. Korean Statistical Information Service. Available online: http://kosis.kr (accessed on 10 August 2021).
4. Mukhiddinov, M.; Muminov, A.; Cho, J. Improved Classification Approach for Fruits and Vegetables Freshness Based on Deep Learning. *Sensors* **2022**, *22*, 8192. [CrossRef] [PubMed]
5. Larsen, A.; Hanigan, I.; Reich, B.J.; Qin, Y.; Cope, M.; Morgan, G.; Rappold, A.G. A deep learning approach to identify smoke plumes in satellite imagery in near-real time for health risk communication. *J. Expo. Sci. Environ. Epidemiol.* **2021**, *31*, 170–176. [CrossRef] [PubMed]
6. Toan, N.T.; Thanh Cong, P.; Viet Hung, N.Q.; Jo, J. A deep learning approach for early wildfire detection from hyperspectral satellite images. In Proceedings of the 2019 7th International Conference on Robot Intelligence Technology and Applications (RiTA), Daejeon, Korea, 1–3 November 2019; pp. 38–45. [CrossRef]
7. Li, P.; Zhao, W. Image fire detection algorithms based on convolutional neural networks. *Case Stud. Therm. Eng.* **2020**, *19*, 100625. [CrossRef]
8. Tang, Z.; Liu, X.; Chen, H.; Hupy, J.; Yang, B. Deep Learning Based Wildfire Event Object Detection from 4K Aerial Images Acquired by UAS. *AI* **2020**, *1*, 166–179. [CrossRef]
9. Toulouse, T.; Rossi, L.; Celik, T.; Akhloufi, M. Automatic fire pixel detection using image processing: A comparative analysis of rule-based and machine learning-based methods. *Signal Image Video Process.* **2016**, *10*, 647–654. [CrossRef]
10. Jiang, Q.; Wang, Q. Large space fire image processing of improving canny edge detector based on adaptive smoothing. In Proceedings of the 2010 International Conference on Innovative Computing and Communication and 2010 Asia-Pacific Conference on Information Technology and Ocean Engineering, Macao, China, 30–31 January 2010; pp. 264–267.
11. Celik, T.; Demirel, H.; Ozkaramanli, H.; Uyguroglu, M. Fire detection using statistical color model in video sequences. *J. Vis. Commun. Image Represent.* **2007**, *18*, 176–185. [CrossRef]
12. Dimitropoulos, K.; Barmpoutis, P.; Grammalidis, N. Spatio temporal flame modeling and dynamic texture analysis for automatic video-based fire detection. *IEEE Trans. Circuits Syst. Video Technol.* **2015**, *25*, 339–351. [CrossRef]
13. Park, M.; Ko, B.C. Two-Step Real-Time Night-Time Fire Detection in an Urban Environment Using Static ELASTIC-YOLOv3 and Temporal Fire-Tube. *Sensors* **2020**, *20*, 2202. [CrossRef] [PubMed]
14. Abdusalomov, A.B.; Islam, B.M.S.; Nasimov, R.; Mukhiddinov, M.; Whangbo, T.K. An Improved Forest Fire Detection Method Based on the Detectron2 Model and a Deep Learning Approach. *Sensors* **2023**, *23*, 1512. [CrossRef]
15. Muhammad, K.; Ahmad, J.; Mehmood, I.; Rho, S.; Baik, S.W. Convolutional Neural Networks Based Fire Detection in Surveillance Videos. *IEEE Access* **2018**, *6*, 18174–18183. [CrossRef]
16. Pan, H.; Badawi, D.; Cetin, A.E. Computationally Efficient Wildfire Detection Method Using a Deep Convolutional Network Pruned via Fourier Analysis. *Sensors* **2020**, *20*, 2891. [CrossRef]
17. Li, T.; Zhao, E.; Zhang, J.; Hu, C. Detection of Wildfire Smoke Images Based on a Densely Dilated Convolutional Network. *Electronics* **2019**, *8*, 1131. [CrossRef]
18. Kim, B.; Lee, J. A Video-Based Fire Detection Using Deep Learning Models. *Appl. Sci.* **2019**, *9*, 2862. [CrossRef]
19. Wu, S.; Zhang, L. Using popular object detection methods for real time forest fire detection. In Proceedings of the 11th International Symposium on Computational Intelligence and Design (SCID), Hangzhou, China, 8–9 December 2018; pp. 280–284.
20. Imran; Iqbal, N.; Ahmad, S.; Kim, D.H. Towards Mountain Fire Safety Using Fire Spread Predictive Analytics and Mountain Fire Containment in IoT Environment. *Sustainability* **2021**, *13*, 2461. [CrossRef]
21. Gagliardi, A.; Saponara, S. AdViSED: Advanced Video SmokE Detection for Real-Time Measurements in Antifire Indoor and Outdoor Systems. *Energies* **2020**, *13*, 2098. [CrossRef]
22. Xu, R.; Lin, H.; Lu, K.; Cao, L.; Liu, Y. A Forest Fire Detection System Based on Ensemble Learning. *Forests* **2021**, *12*, 217. [CrossRef]
23. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [CrossRef]
24. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; ECCV 2016. Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; Volume 9905. [CrossRef]
25. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
26. Abdusalomov, A.; Baratov, N.; Kutlimuratov, A.; Whangbo, T.K. An improvement of the fire detection and classification method using YOLOv3 for surveillance systems. *Sensors* **2021**, *21*, 6519. [CrossRef]

27. Avazov, K.; Mukhiddinov, M.; Makhmudov, F.; Cho, Y.I. Fire Detection Method in Smart City Environments Using a Deep Learning-Based Approach. *Electronics* **2021**, *1*, 73. [CrossRef]
28. Sisias, G.; Konstantinidou, M.; Kontogiannis, S. Deep Learning Process and Application for the Detection of Dangerous Goods Passing through Motorway Tunnels. *Algorithms* **2022**, *15*, 370. [CrossRef]
29. Voudiotis, G.; Moraiti, A.; Kontogiannis, S. Deep Learning Beehive Monitoring System for Early Detection of the Varroa Mite. *Signals* **2022**, *3*, 506–523. [CrossRef]
30. Kontogiannis, S.; Asiminidis, C. A Proposed Low-Cost Viticulture Stress Framework for Table Grape Varieties. *IoT* **2020**, *1*, 337–359. [CrossRef]
31. Ahrens, M.; Maheshwari, R. *Home Structure Fires*; National Fire Protection Association: Quincy, MA, USA, 2021.
32. Mukhiddinov, M.; Abdusalomov, A.B.; Cho, J. Automatic Fire Detection and Notification System Based on Improved YOLOv4 for the Blind and Visually Impaired. *Sensors* **2022**, *22*, 3307. [CrossRef]
33. Abdusalomov, A.B.; Mukhiddinov, M.; Kutlimuratov, A.; Whangbo, T.K. Improved Real-Time Fire Warning System Based on Advanced Technologies for Visually Impaired People. *Sensors* **2022**, *22*, 7305. [CrossRef]
34. Robmarkcole 2022, Fire-Detection-from-Images, Github. Available online: https://github.com/robmarkcole/fire-detection-from-images (accessed on 10 November 2022).
35. Glenn Jocher 2022, Yolov5, Github. Available online: https://github.com/ultralytics/yolov5 (accessed on 10 November 2022).
36. Valikhujaev, Y.; Abdusalomov, A.; Cho, Y.I. Automatic fire and smoke detection method for surveillance systems based on dilated CNNs. *Atmosphere* **2020**, *11*, 1241. [CrossRef]
37. Redmon, J. Darknet: Open-Source Neural Networks in C. 2013–2016. Available online: http://pjreddie.com/darknet/ (accessed on 22 October 2022).
38. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
39. Mukhiddinov, M.; Abdusalomov, A.B.; Cho, J. A Wildfire Smoke Detection System Using Unmanned Aerial Vehicle Images Based on the Optimized YOLOv5. *Sensors* **2022**, *22*, 9384. [CrossRef] [PubMed]
40. Safarov, F.; Temurbek, K.; Jamoljon, D.; Temur, O.; Chedjou, J.C.; Abdusalomov, A.B.; Cho, Y.-I. Improved Agricultural Field Segmentation in Satellite Imagery Using TL-ResUNet Architecture. *Sensors* **2022**, *22*, 9784. [CrossRef]
41. Sharma, A. Training the YOLOv5 Object Detector on a Custom Dataset. 2022. Available online: https://pyimg.co/fq0a3 (accessed on 22 October 2022).
42. Ayvaz, U.; Gürüler, H.; Khan, F.; Ahmed, N.; Whangbo, T. Automatic speaker recognition using mel-frequency cepstral coefficients through machine learning. *Comput. Mater. Contin.* **2022**, *71*, 5511–5521. [CrossRef]
43. Nodirov, J.; Abdusalomov, A.B.; Whangbo, T.K. Attention 3D U-Net with Multiple Skip Connections for Segmentation of Brain Tumor Images. *Sensors* **2022**, *22*, 6501. [CrossRef]
44. Abdusalomov, A.; Whangbo, T.K. An improvement for the foreground recognition method using shadow removal technique for indoor environments. *Int. J. Wavelets Multiresolut. Inf. Process.* **2017**, *15*, 1750039. [CrossRef]
45. AlZoman, R.M.; Alenazi, M.J.F. A Comparative Study of Traffic Classification Techniques for Smart City Networks. *Sensors* **2021**, *21*, 4677. [CrossRef] [PubMed]
46. Pereira, F.; Correia, R.; Pinho, P.; Lopes, S.I.; Carvalho, N.B. Challenges in Resource-Constrained IoT Devices: Energy and Communication as Critical Success Factors for Future IoT Deployment. *Sensors* **2020**, *20*, 6420. [CrossRef] [PubMed]

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