ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ

«ΠΛΗΡΟΦΟΡΙΚΗ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΑΣΦΑΛΕΙΑ, ΔΙΑΧΕΙΡΙΣΗ ΜΕΓΑΛΟΥ ΟΓΚΟΥ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΠΡΟΣΟΜΟΙΩΣΗ»

«Αναγνώριση καπνού και φωτιάς σε βίντεο με μηχανική μάθηση»

Κωνσταντίνος Ελεμένογλου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέποντες Καθηγητές:
Γεώργιος Σταμούλης
Κωνσταντίνος Κολομβάτσος

Λαμία, Απρίλιος 2023

UNIVERSITY OF THESSALY

SCHOOL OF SCIENCE

INFORMATICS AND COMPUTATIONAL BIOMEDICINE

«COMPUTER AND TELECOMMUNICATION SYSTEMS SECURITY, BIG DATA MANAGEMENT, SIMULATION»

«Smoke and fire detection in video based on machine learning»

Konstantinos Elemenoglou

MASTER THESIS

Supervisors:

Georgios Stamoulis

Konstantinos Kolomvatsos

Lamia, April 2023

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ

ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ

ΚΑΤΕΥΘΥΝΣΗ «ΠΛΗΡΟΦΟΡΙΚΗ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΑΣΦΑΛΕΙΑ, ΔΙΑΧΕΙΡΙΣΗ ΜΕΓΑΛΟΥ ΟΓΚΟΥ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΠΡΟΣΟΜΟΙΩΣΗ»

«Αναγνώριση καπνού και φωτιάς σε βίντεο με μηχανική μάθηση»

Συγγραφέας: Κωνσταντίνος Ελεμένογλου
Επιβλέπων καθηγητής: Γεώργιος Σταμούλης
Επιβλέπων καθηγητής: Κωνσταντίνος Κολομβάτσος

Τριμελής Επιτροπή:

1.
2.
3.

Επιστημονικός Σύμβουλος:

«Υπεύθυνη Δήλωση μη λογοκλοπής και ανάληψης προσωπικής ευθύνης»

«Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων και γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα και ενυπογράφως ότι η παρούσα εργασία με τίτλο «Αναγνώριση καπνού και φωτιάς σε βίντεο με μηχανική μάθηση» αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές από τις οποίες χρησιμοποίησα δεδομένα, ιδέες, φράσεις, προτάσεις ή λέξεις, είτε επακριβώς (όπως υπάρχουν στο πρωτότυπο ή μεταφρασμένες) είτε με παράφραση, έχουν δηλωθεί κατάλληλα και ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.»

Ο ΔΗΛΩΝ

Κωνσταντίνος Ελεμένογλου
24/04/2023

Υπογραφή

Acknowledgements

I would like to express my deepest gratitude to my professors and supervisors, Georgios Stamoulis and Konstantinos Kolomvatsos, for giving me the opportunity to do my thesis in a great project and get involved with some revolutionary technologies. A major thanks for all their help and guidance throughout this trip. I am extremely grateful to my parents, my brother, and my friends, for helping me and believing in me. They all knew that it was a difficult journey, but their positive energy was overwhelming. Words cannot express my gratitude to Despoina for her invaluable patience and understanding. I could not have undertaken this journey without her support. In the end, I cannot thank her enough for the fact that I finish this postgraduate program as a different person than I started it, with my little boy Iosif now in my life.

Abstract

Fires can endanger lives, destroy private properties, and cause huge damages to the environment and the public infrastructure worldwide. Disaster information services are very critical to detect and prevent fire in its early stages. In this thesis, firstly we examine the phenomenon of fire and smoke and their composition. The development of effective and fast detection methods on the base of those features, can confront with greater success this phenomenon. Then, we will summarize all the intertemporal technologies that have been used in various fire and smoke detection methods, along with their techniques and methodologies, mainly focusing on the Machine Learning approaches. Furthermore, we will check the architecture and infrastructure of Edge Computing, the process of moving some of the network computational load towards the edge nodes. Additionally, we will check all the different concepts of it, the emerging technologies that can be implemented into this, and finally make a survey of all the different fire and smoke detection techniques with edge computing technologies.

Keywords

Data science, artificial intelligence, machine learning, deep learning, neural networks, convolutional neural networks, deep convolutional neural networks, CNN, DNN, edge computing, edge computing types, cloud computing, network infrastructure, edge device, edge node, edge server, edge data centers, edge computing technology, edge artificial intelligence, internet of things, internet of everything, big data, fire, smoke, fire detection, smoke detection, prevention, suppression, fire alarm, smoke alarm, physical sensors, sensing technology, video-based fire detection, image processing, image classification, image recognition, image segmentation, image-based detection, object detection, computer vision, color, texture

Table of contents

List of figures

List of tables

CHAPTER 1

INTRODUCTION

1.1 Background

The problem of forest fires especially during the summer period has a great impact in everyday life. It can endanger human and animal lives, and cause irreparable damage to the environment, the atmosphere, and the ecological balance. It can threaten the economy, the social and public development by destroying residents, transport and working facilities and damage tourism and health services in general. Every year hundred thousand square kilometers of forests are lost. The diversity in the characteristics of fire and smoke, like color, direction, density, lighting, development, and appearance, make it essential to develop recognition methods in the early stages of the fire in order to control and minimize the damage with the less possible consequences [20].

Fig. 1.1. *Various appearances of fire*

1.2 An overview of detection technologies

Over the years there have been efforts to fight the fire phenomenon, either with great success or less. Various methodologies have been introduced, tested, formulated, and implemented to detect fire accidents and minimize the damage. The dynamic growth of smoke and fire requires effective learning approaches in order to be confronted successfully. Such new techniques can establish an optimal surveillance environment with much better real time monitoring and more reliable results. As Data Science grows rapidly and new smart systems and technologic methodologies dramatically evolve through the years, we see that Artificial Intelligence can play a huge and effective role in that part in comparison to conventional smoke and fire detection technologies. Machine learning overall is an approach of artificial intelligence which is trained on various datasets and learns to identify trends and patterns, improve its learning behavior, and make predictions on the basis of computational intelligence techniques [20].

Deep learning is the field of Machine Learning where more complexed tasks can be solved with the use of deep neural networks, without human dependency, by processing unstructured data, and doing unsupervised learning. Especially with the rapid growth of the Internet of Everything, we have a huge number of smart end devices which are connected to the network and can provide big scale data. Deep Convolutional Neural Networks is a key technology to implement smart 5G applications on mobile devices, and to assist in Computer Vision problems like detection[20].

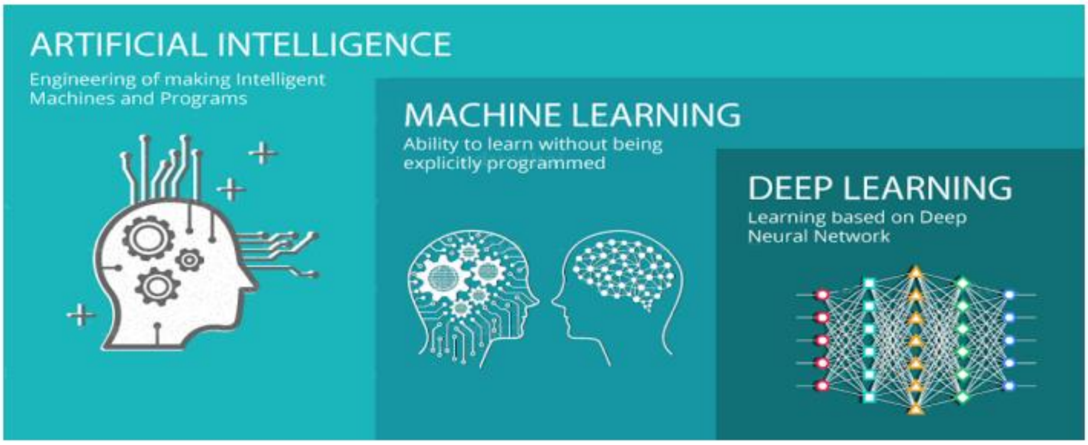


Fig. 1.2. *Conceptual hierarchy of Artificial Intelligence and its subsidiaries*

The computing of the diverse information of image processing transformation is based on complex activities, fast calculations, and the effective performance of emerging technologies like edge computing. Unlike centralized big data processes, the aim is closer to the source of the data, distributed to the edge of the network, allowing it to gain important advantages and achieve higher efficiency. In this thesis we are going to examine all the different methodologies which are used to detect fire and smoke in image and video based on machine learning and computer vision in general. Furthermore, we will review and focus on a very specific area, fire and smoke detection and management techniques combined with the edge computing technologies [15].

CHAPTER 2

THE PHENOMENON OF FIRE

2.1 Composition of fire and smoke

Oxygen, heat, fuel, and the chemical reaction are the four elements which combined together are known as the "fire tetrahedron". Fuel is the combustible substance, like wood, paper, gasoline, that is being burned, while oxygen, the gas responsible for the combustion, combined with the fuel produces the heat, which is the energy that sustains the combustion. Fire is the rapid oxidation of a fuel material in the exothermic chemical process of combustion, releasing heat, light, electrical sparks, while the chemical reactions produce various reaction byproducts such as smoke, ash, carbon dioxide $CO2$, methane $CH4$, nitrous oxide $N2O$ and other pollutants. Fire is hot because the conversion of the weak double bond in molecular oxygen $O2$, to the stronger bonds in the combustion products carbon dioxide and water releases energy (418kJ per 32g $O2$). Uncontrollable fires can result in long term environment effects like huge amounts of carbon dioxide and smoke in the air and weather pattern changes. The composition of smoke can vary, depending on many factors like fuel, oxygen, and temperature. Some of the basic components of smoke are tiny solid particles, gases, water vapor, and volatile organic compounds (VOCS) which are a series of organic chemicals that are released in the atmosphere during the burning [27].

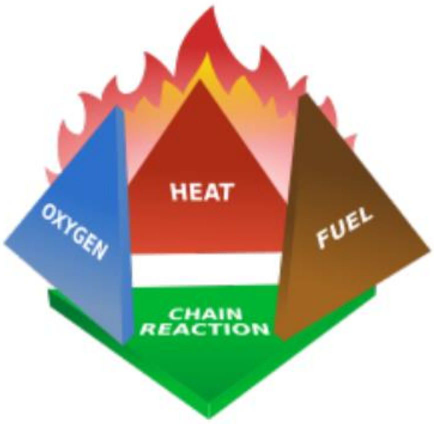


Fig. 2.1. *Fire tetrahedron*

The flame is the visible part of the fire. Flames are mainly composed of carbon dioxide, water vapor, oxygen, and nitrogen. If they are hot enough, the gases can ionize to produce plasma. Depending on the hazardous substances and any external impurities, the color of the flame and the intensity of the fire will be different. A flame is a mixture of reactant gases and solids that emit visible, infrared, and sometimes ultraviolet light, the frequency spectrum of which depends on the chemical composition of the fuel, the temperature of the flames and intermediate reaction products. In many cases, such as the burning of organic matter, for example wood, or the incomplete combustion of gas, solid incandescent particles called soot produce the familiar red-orange glow of a 'fire'. This light has a continuous spectrum.

Complete combustion of the gas has a faint blue color due to the emission of wavelength radiation from various electron changes in the excited molecules formed in the flame. Oxygen is usually involved, but hydrogen burning in chlorine also produces a flame, producing hydrogen chloride (HCL). Other possible combinations that produce flames are, among many, fluorine and hydrogen, hydrazine, and nitrogen tetroxide. The structure of a flame can be divided into three parts, the inner cone is where the fuel is actively burned, the outer cone is the part which contains the unburned fuel and byproducts, and the non-luminous zone is where there is incomplete combustion, and the unburned fuel and byproducts are released into the atmosphere [21].

2.2 Prevention and suppression of fires

For centuries there have been some key factors for preventing and suppressing fires. Firstly, quick and effective detection historically has been an important factor in fighting fires and stop them from spreading, while at the same time taking the proper actions like early response, extinguishing or evacuating. Additionally, focused prevention is essential to eliminate or minimize the reasons for a fire to start in the first place like safety procedures, proper storing, disposing, and handling of dangerous materials, and maintenance of electrical systems. As far as the suppression there are several methods used to stop fires, like water, fire fighters, fire extinguishers, fire blankets and sprinkler systems. Water is the most common extinguishing agent that cools the fuel and stops the heating, trained firefighters with specialized equipment, fire extinguishers are portable devices with various extinguishing agents like water, foam or dry chemicals, fire blankets are made of flame-resistant materials and can suppress small fires or wrapped around a person, and sprinkler systems are automatic devices releasing water automatically [13].

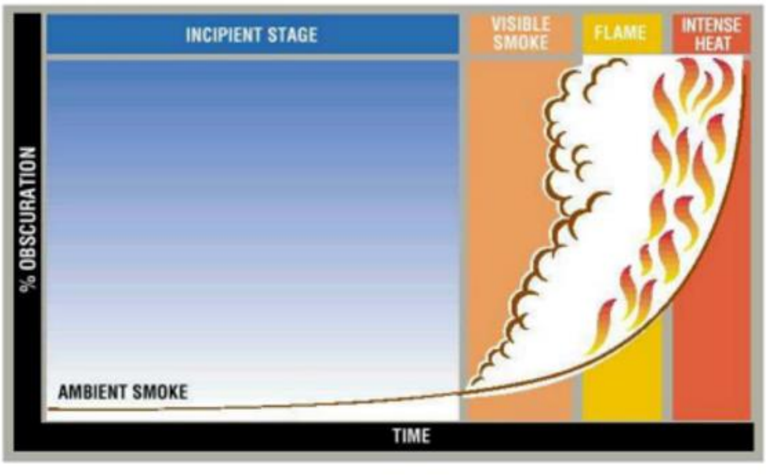


Fig. 2.2. *Duration of the forest fire period*

Other additional strategies against fires can be education, training seminars and public awareness campaigns that provide understandings of the risks of fires, the establishment of fire codes and regulations that force specific safety measures for buildings and other areas, proper building designs and constructions with the use of fire resistance materials, fire alarms and smoke detectors, as long as building layouts for easy evacuation and emergency planning with effective evacuation routes, designating meeting places and practicing crisis drills. Some methods used frequently since decades by authorities and individuals to successfully in time detect and prevent fires are the fire weather forecasts and estimates of fuel and moisture, watch towers, lightning detectors which detect the coordinates of the strike, optical smoke detection, spotter planes, water tankers, infrared vision, controlled burning of areas close to the fire to eliminate fuel, direct communication via phone calls and educational programs for Fire Safety and Fire Protection [13].

2.3 Consequences

The negative consequences of fire can be severe and long-lasting, and cause disasters beyond measure and description like global warming. Fires can have devastating effects on wildlife and the surrounding landscape. They can endanger life and property, air pollution and water contamination, the atmospheric carbon-dioxide can no longer be absorbed, significant economic impact, displacement of the affected, phycological impact and post-traumatic stress disorders, long term health effects, and finally the tragic loss of human and animal lives. If fire destroys protective vegetation, heavy rainfall can lead to increased soil erosion by water. Large areas of productive agricultural and forest land can be lost, along with the loss of biodiversity where species of the flora and fauna are diminished. Eventually fires lead to failures of the entire ecosystem [13].

CHAPTER 3

EDGE COMPUTING

3.1 Overview

3.1.1 Benefits

In comparison to the centralized model of data centers with high demands on communication and computational infrastructure in order to process large scale data generated by edge devices, the model of edge computing is to distribute the workload of the performing computations to the edge nodes of the network and closer to the source of the data so it can improve the response time and also for bandwidth reasons. User gadgets, mobiles, tablets, smartphones music players wearables, game controllers, Internet-of-Things devices etc. are referred as edge devices, while nodes through which network traffic is directed like sensors, cameras, routers, switches, gateways, servers, small/macro base stations are referred as edge nodes. There are various needs that actuate the use of edge computing in a wide range of technologies like image and video processing and detection, the Internet of Things, mobile computing etc. Such needs are the low latency computing in decentralized environments, the smart computational methodologies which require minimum computational capabilities of the edge node resources, the decrease of energy consumption when the computational tasks are performed on the edge nodes, the distribution of the network traffic as long as the dealing of the enormously increasing rate of volume of data, and the first level filtering of data from the frontend edge devices [14].

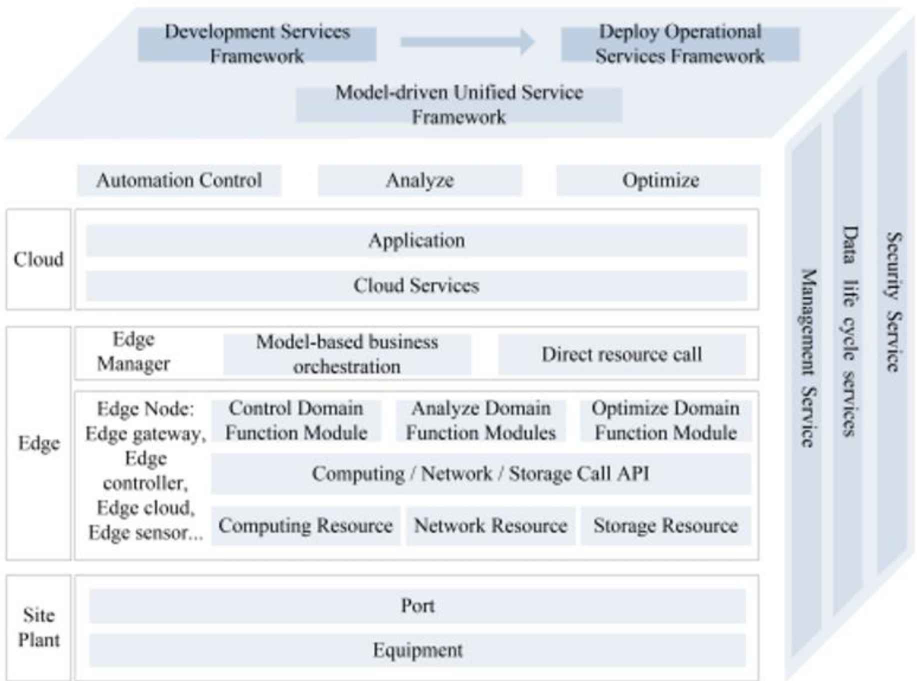


Fig. 3.1. *Edge computing architecture*

3.1.2 Challenges and opportunities

These needs however come along with emerging challenges some of which are the actual computing capabilities of the existing edge devices, the decentralization of the computing environment and the partitioning of the task, the managing of the distributed resources which could be located in multiple locations connected in different networks, the exploiting of the distributed network in order to find the appropriate edge nodes, the security and safety of a public environment, the high demands of creditability to ensure Quality-of-Service(QoS) and Quality-of-Experience(QoE), and finally the accessibility of the costs to maintain data and energy [14].

Despite all these challenges, this technology has all the potentials to achieve a most efficient way of distributed computing. Some of the opportunities that arise are the software frameworks, programming languages, deep learning algorithms, services, mobile containers, and toolkits that would be needed in a heterogenous environment of cross hardware in order to deploy and execute applications. Additionally micro-operating systems and virtualization, with a variety of defined standards, implemented benchmarks and reliable marketplaces could prove to be a feasible solution. Edge analytics would also need machine learning libraries and lightweight algorithms to be employed to edge node devices and support data processing in mixed distributed systems faster and more accurately [14].

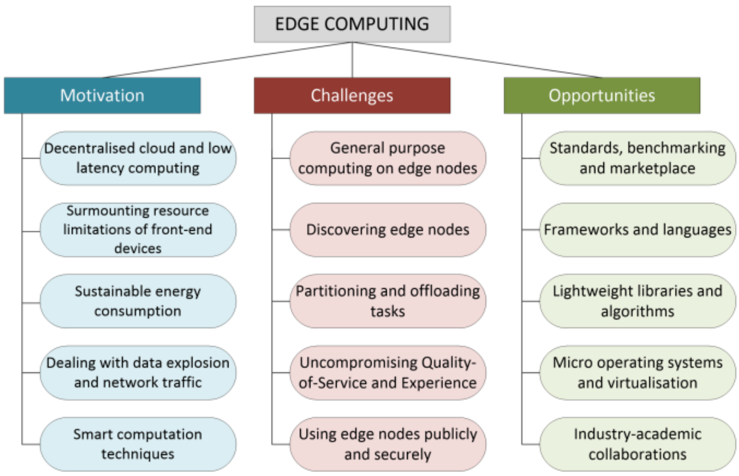


Fig. 3.2. *Motivation, Challenges and Opportunities in Edge computing*

3.2 Infrastructure in Edge computing

3.2.1 Hardware

There are various devices and tools that consist of and associate with edge computing hardware, which can be separated into two main categories, computation devices and network devices. In fact, these edge devices and edge nodes, are small devices located at the edge of the network that can enable edge computing by collecting and processing data before sending them to edge data centers for further analysis or process network traffic. These can be single board computers (SBC), which are small computers based on a single circuit board integrating various edge components together, or various commodity products. Edge data centers are small data centers also at the edge of the network, with the task to process and store data and their design allows them to be more efficient and more compact in comparison to traditional data centers. There, are also located the edge storage devices, such as solid-state drives (SSD) and memory cards, which can be used to provide quick accessibility to large amount of data locally and reduce the data network traffic to a centralized server and help the network be faster [18].

The communication between edge devices and edge data centers is possible due to some other devices, edge gateways and edge routers, which are acting as the bridge between those two. Their task is to perform a part of the processing of the data and assist in filtering and reducing the amount of data which are transmitted over the network. The more complicated and demanding part of the data processing on the edge of the network is done by the edge servers. These are small servers that are used to host applications, perform analytics, and provide real time responses. Also, there are other network devices like the Wireless Access Points (Wi-Fi APs) and Edge racks, which is the packing into a single rack of all the network computing and storage resources [18].

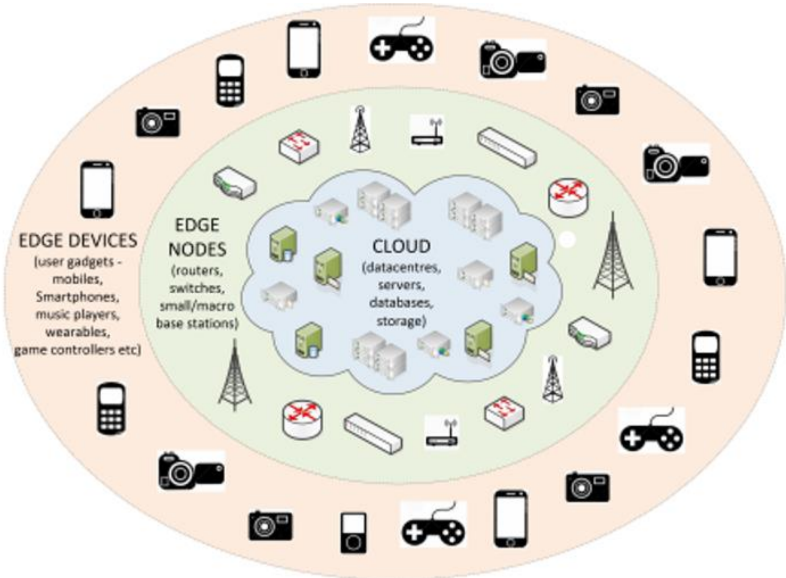


Fig. 3.3. *Edge devices and Edge nodes in relation to the cloud*

3.2.2 Software

The software applications that are designed to run at the edge of the network and include data processing and analysis tools, are the Edge Applications or Edge-native applications. Edge Computing Standards are several protocols and standards, related to security, communication, operation, and data formats, which have been developed to ensure interoperability and compatibility between different components and systems. A set of practices, protocols and security measures that are used to secure edge devices and networks are called Edge Security, and include features like firewalls, authentication, encryption, prevention systems and intrusion detection [18].

Moreover, there are plenty platforms and software tools, known as edge computing software, which are commonly used in these environments and are designed to enable edge computing. Some of them are open-source platforms like Kubernetes and Docker, while there are also commercial platforms like AWS Greengrass and Microsoft Azure IoT Edge. Also, containerization is a technology that allows applications to be packaged in lightweight, portable Containers, making it much easier to deploy and operate apps in such distributed and decentralized environments. Additionally, in order to reduce the hardware costs and optimize resource utilization, multiple virtual machines are running on a single edge server, a technology known as virtualization, where a set of virtualized resources are emulating a physical computer. Such software administrative entities are the Software Defined Networking (SDN), the Network Function Virtualization (NFV) and the Overlay Network. SDN is responsible to design the network, manage network traffic from deciding the destination and forwarding the traffic to it, and NFV is responsible for routing and firewalls, allowing functions to run on virtual machines. Those two are combined together, but also can be independent from each other, allowing flexible and efficient networking management, a very useful aspect in complex and dynamic environments. Last, there is the Overlay network, which is a virtual network based on an underlying physical network, that provides network services where nodes are connected by virtual links, so they create new virtual data paths instead of physical links [18].

3.2.3 Middleware

Middleware runs on an operating system and is responsible to provide complementary services, like monitoring, coordination, orchestration, and communication facilities. An important implementation of a useful technology in edge computing is artificial intelligence. Edge AI is a branch of AI where machine learning models are deployed at the edge of the network and not in cloud, and AI algorithms run on edge devices that need to collaborate with each other, allowing real time analysis and fast and accurate decision making without the need of a central server for data processing. This is proved to be a huge assistance to various applications which require real time processing of data, including image detection and recognition, natural language processing, predictive maintenance, and autonomous vehicles. A similar approach to that technology is Distributed Artificial Intelligence (DAI), which focuses on the development of AI algorithms and systems that can operate in distributed and decentralized environments and make decisions based on the data collected from those environments. Edge-to-cloud orchestration is a process of managing and coordinating workloads across various levels of the edge infrastructure, including edge nodes, edge servers and cloud servers as well. This way workloads are executed effectively and efficiently. A set of tools and processes that are used to design solutions, provide device configuration, manage edge devices, monitor infrastructure remotely, asset tracking and control systems are referred as Edge Management. Finally, the processing of the collected data at the edge nodes and the real time analysis before transmitting them to centralized data centers or to cloud servers is called Edge Analytics [18].

3.3 Edge computing types and related methods

There are various types of edge computing, which can offer a series of different benefits over the traditional cloud computing and centralized data processing and analysis. Furthermore, they can provide more efficient data analysis and processing, higher data privacy and security, faster real time processing, greater scalability, and flexibility, as well as improved network latency and high bandwidth connections to edge devices. A key method factor in edge computing is mobility management, that supports in the resource discovery and the mobility of the applications based on the geographical distribution of the resources, ensuring the access to the services during the movement without off times. Edge computing also includes other important technologies like traffic offloading, caching acceleration and network control. Fog computing is a hierarchical architecture that involves placing resources, like data storage and processing power, closer to edge devices which have more computing power than traditional devices, but less than edge servers, resulting in speeding up data processing and reducing network latency. In the edge devices of that distributed model, the data processing and filtering is initially performed and then the information is sent to different levels of the network into more powerful servers for further processing, allowing greater scalability and flexibility. Cloudlet Computing is a distributed type of edge computing, which provides small data centers or "cloudlets" in the edge of the network, that allows edge devices to be provided storage services and data processing with low latency, so it improves the network performance and reduce data transfers over the network [15].

Other key methods in this technology, ensure data security privacy like access control schemes, some of which are attribute-based access control and role-based access control, that provide flexible control and management through mapping mechanisms. Another important factor is the data security, consisting of data confidentiality, secure data sharing, integrity audit and searchable encryption. Furthermore, there is privacy protection for data, location, and identity, and also there is identity authentication, either in a single domain, cross domain or switching. Multi Access Edge Computing is a standard architecture for computing across different layers of networks and devices, developed by the European Telecommunications Standards Institute, aiming to provide better performance and latency results, in order to meet the requirements of the 5G network. 5G, the latest generation of cellular networks, is designed to provide faster and more reliable data transmission. On-device Edge Computing requires running the data processes and computing tasks only locally, directly on edge devices, without the need of external computing resources, assisting overall in reducing network traffic, improving data processing, and enhancing data security and privacy [15].

Computing offloading refers to resource limited edge devices which partially or fully transfer computing processes from mobile devices to close network components with much higher computing resources. Mobile Edge Computing(MEC) which uses computing offloading, is related to the needs of mobile networks and provides low latency services to mobile devices such as tablets and smartphones, like location-based services, image and video streaming, applications, data processing tasks and augmented reality. It deploys computing and communication resources like processing power, data storage and network connectivity at the edge nodes of the mobile network, on base stations or other edge devices. In order to provide resource efficient, like Quality of Service, it is important for MEC to take into consideration the task offloading generated by the devices, the resource allocation in the device-server communication, and the handling of the inter-server communication. A similar approach is the Hybrid Edge Computing which combines lots of different types of computing technologies such as Fog computing, MEC and Cloudlet computing, in order to create a hybrid infrastructure that meets all the requirements of multiple scenarios and use cases [24].

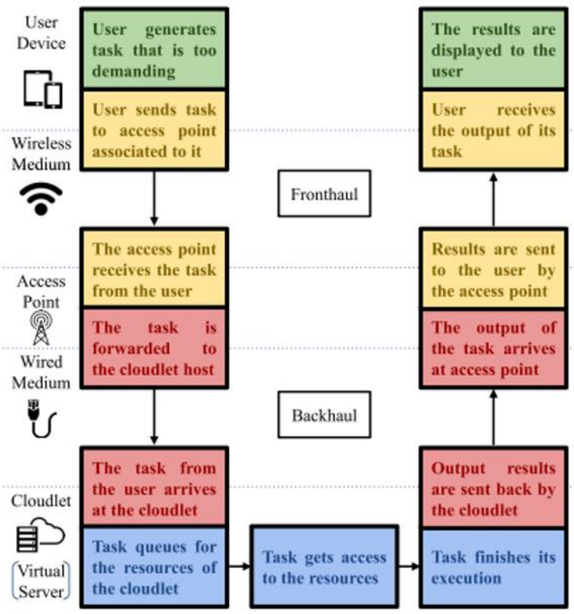


Fig. 3.4. *View of MEC service model(although Access Point and Cloudlet are virtually different entities,they are usually physically together and always at the edge)*

Energy efficient computing is a computing technique focusing on minimum power consumption without performance reduction. It is very important in edge computing environment to achieve that edge devices and edge servers can operate without recharging or maintenance for long time periods. Computing offloading consists of two specific parts, offloading decision and resource allocation. In the first, the user equipment determines three cases, the quantity that needs to be offloaded and the energy that needs to be consumed decided by the system parser, the specific tasks and contents that need be to offload decided by the application type and code data partition, and finally whether to offload or not decided by the decision engine. The second part has to do with the way that the computing tasks will be divided and allocated to various servers [15].

3.4 Emerging technologies along with Edge computing

There are various other technologies which can be used in conjunction with edge computing to provide more efficient data analysis and processing, as well as improved latency and high bandwidth connections to edge devices. Mesh networks are networks where devices communicate with each other directly without the need of a centralized system or a central server, which could prove to be sometimes unreliable and dependable. Digital twins are virtual representations of physical objects, or systems, that consist of data from the real network environment, like network topologies, schedulers, and channels, and from fundamental rules from theoretical studies, like adjustments in information and queuing theories. Digital twins perform real time monitoring and analysis of physical systems enabling various types of optimizations and maintenance based on prediction, like training the DNN. Edge to Cloud integration is the concept of integrating edge computing to cloud computing, resulting in a unified computing environment, to allow applications to move between edge and cloud based on their needs [23].

In the edge computing scenarios with many server and applications and parallel users, complex problems occur due to high dimensionality, complicated efficient configuration, and high amount of data to be processed. A utilization of machine learning algorithms could allow the edge computing services to address these problems, by drawing conclusions and make predictions based on the networks existing data, providing optimal solutions. Machine learning can lead edge computing operations to more efficient networks and cloud services, allow more complex systems to be built in this environment, and provide higher quality of service. There are also two relatively new technologies which can be implemented in edge computing environments to provide real time immersive experiences, augmented reality(AR) and virtual reality(VR) and can prove to be very useful in edge computing and specifically to applications related to surveillance, training, education, and remote assistance [24].

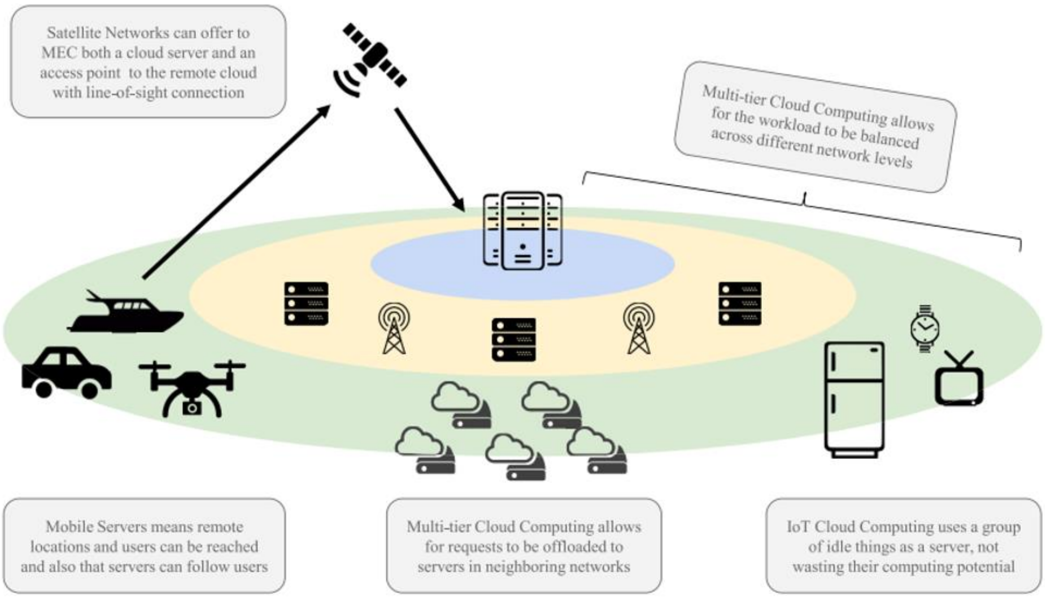


Fig. 3.5. *Future researched MEC applications implemented along with ML*

As edge computing technology continues to evolve, new emerging platforms, frameworks, infrastructures, and tools are developed to support it. The specific models used, will depend on the requirements and the needs of the applications and the related use case. Such a computing model is Serverless Computing, where the cloud providers are the ones who manage the scaling and the workloads of the various levels of the computing infrastructure, the applications are defined by the actions and the events that trigger them, while the application developers mostly emphasize on coding, by decomposing large applications into small functions, allowing software components to scale individually. There can be also software frameworks like Internet of the Things platforms, which can include various applications, functions, and features, and provide all the necessary services and tools for data processing and real time data analytics. Another approach in the mobile network infrastructure is the Open Radio Access Network (Open RAN) that enables the use of nonproprietary subcomponents from a variety of different vendors, allowing open standards and the interoperability between those vendors, providing flexible and low latency connectivity to the edge devices. Furthermore, there is also Hybrid Cloud approach in the network infrastructure, which is a combination of public and private cloud infrastructure, that combined with edge computing environment can provide flexibility and scalability by allowing workloads to be distributed across different levels of the network based on various requirements like data privacy, cost, and performance [25].

A part of the emerging blockchain technology, the distributed ledger, can also be used in such distributed environments to provide consensus of replicated, shared, and synchronized digital data spread across all the nodes of the network, ensuring replication, security, reliability, information protection, decentralized storage, and fast process of data. Those features are very important for applications that need to have high standards of transparency, data integrity and security. Furthermore, microservices could run in those environments and aid to integrate distributed processes into edge platforms and deploy them into serverless pipelines. An additional revolutionary technology that can be mutually developed and integrated into edge computing is Quantum computing, by using two important concepts of quantum physics, superposition and entanglement. The latter technology can allow the edge nodes to perform complex calculations and processes, and simulation in real time, secure and reliable, further promising additional advantages to such distributed edge networks [26].

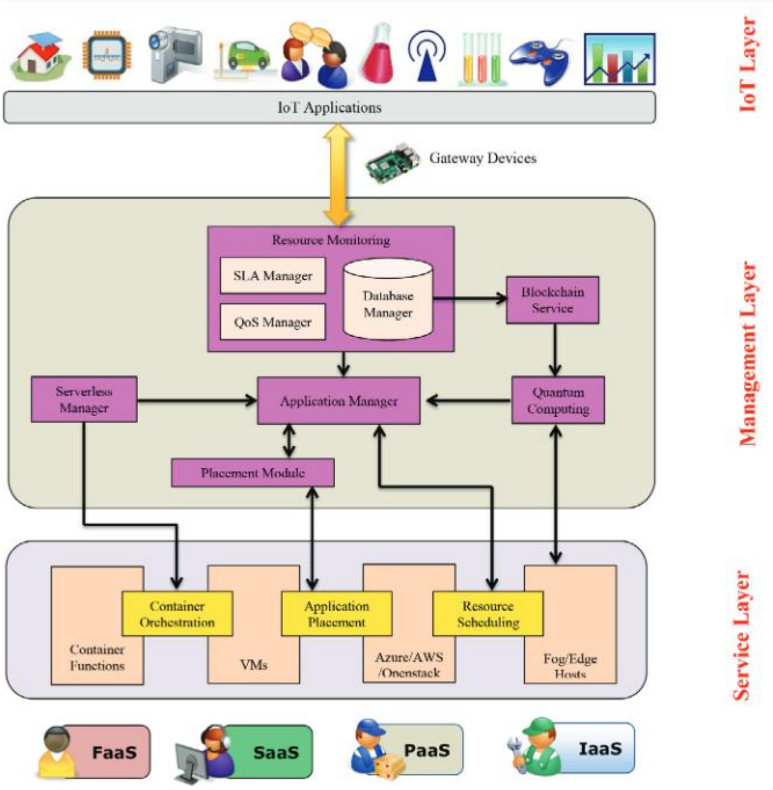


Fig. 3.6. *Quantum and blockchain based edge computing model*

CHAPTER 4

SURVEY OF THE VARIOUS TECHNIQUES USED FOR FIRE AND SMOKE DETECTION

4.1 Traditional Approaches with human intervention

The first effort to detect fire and smoke was done by traditional fire alarm systems, which employed physical sensors and sensing technologies. The most common of them are thermal detectors, flame detectors such as optical(ionization) and photoelectric, smoke detectors and combinations of different sensor-based tools. The parameters that were tracked down from the sensors, were given as input to a processor which then on took decisions based on that. But they are proved to be slow responsive, with false alarms, not much accurate and less capable of providing evidence for the route of fire. Additionally, they need human intervention for the confirmation of the fire accident, as they cannot operate without continuous surveillance coverage. In an open environment they cannot precisely measure the size and the location of fire, they mostly operate close to the place installed and need more time to trigger an alarm. Also, conventional sensors could prove not to be trustworthy in extreme weather and location conditions as they are more sensitive to damages and may malfunction. A possible general loss of confidence in the fire and smoke detection systems due to false alarms could cause big economic damages and eventually real casualties [1].

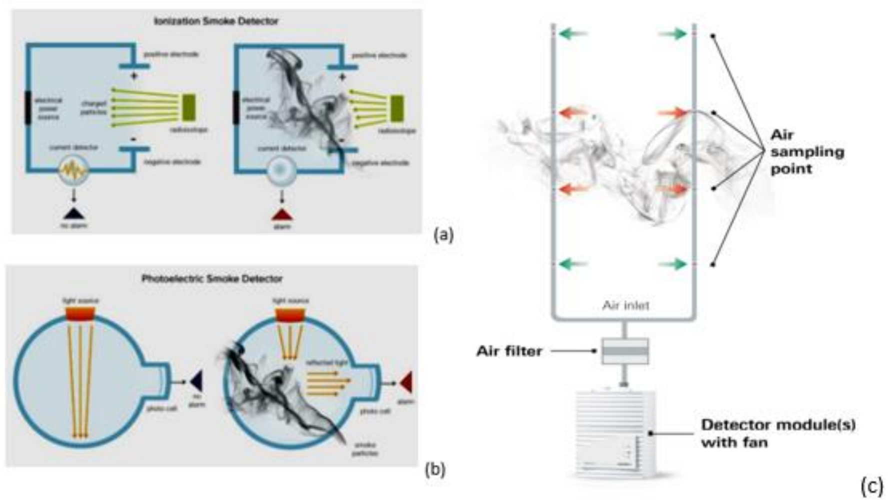


Fig. 4.1. *Smoke detectors: a) ionization, b) photoelectric, c) air sampling*

4.2 Classic Methodologies in Image processing

4.2.1 Image recognition and segmentation

Image processing then came to overcome such issues in the fire and smoke detection as crucial data could be extracted from the images, where the digital cameras steadily produce images. These approaches offered earlier fire detection, faster response, and more information on the fire progress. Image processing is a fundamental subfield of computer vision, and as a methodology is considered the analysis, the manipulation, and the feature extraction via computational techniques of useful information from images, in a wide range of application using software tools[17].

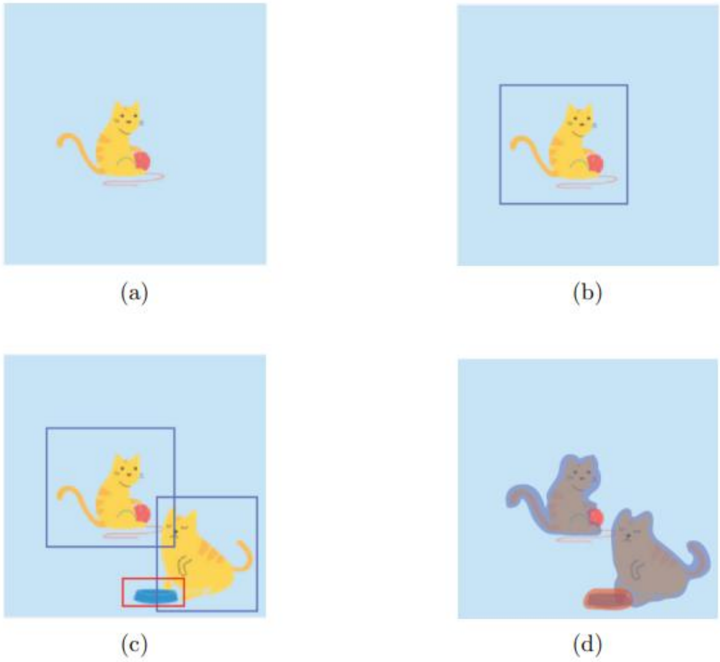


Fig. 4.2. *Computer vision main tasks: a) classification, b) classification & localization, c) multi object detection, d) segmentation*

Firstly, image is digitally imported, then analyzed and manipulated accordingly. Other steps of this process can be the visualization, in order to find not initially visible objects, sharpening and restoration, as to enhance it. Image recognition is the process to assign a label to an image. Image segmentation is the process where an image is broken into many segments of elements which are assigned one pixel, based on features like color and texture. Feature extraction is the identification and extract of key features like corners, edges or text patterns. Classification it the part where the input data, set of features, are categorized into different classes and categories, while the localization is the part where the objects are identified based on position and orientation. The processing can track motion changes and by checking the changed pixels and other discontinuities and dissimilarities in the local neighborhood, then sends the output to the system algorithm to decide on the smoke and fire glow detection [17].

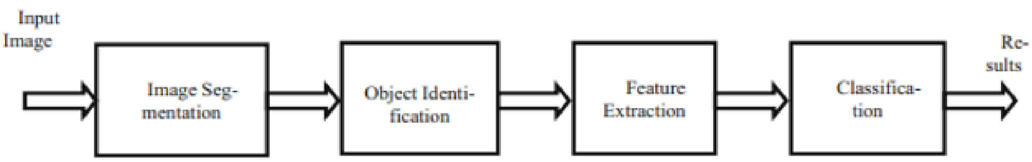


Fig. 4.3. *Outline of image segmentation*

Image technology is evolving through time, and the equipment used is also more affordable and with better quality and efficiency, like cameras and processing units, providing more detailed visual information. However, most of the related methodologies can be applied only to specific images as they require steady and clear shots of fires, flames, and smokes. Initially there are various traditionally image processing techniques through video, like motion models or like RGB (red, green, and blue), HIS (hue, saturation, and intensity) and YCbCr (luminance with two color difference signals) color models. These do not require historical data and supervision for training but have specific limits because of the restricted features in images such as complex scenarios, occlusion, illumination, and clutter effect. Additionally, all of these models can be more vulnerable to extreme weather conditions as their performance could be reduced dramatically over events like lightnings and various other distortions [3].

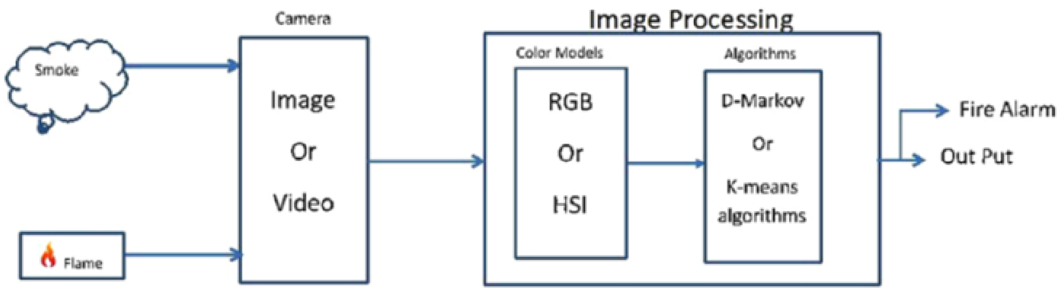


Fig. 4.4. *Block diagram of fire and smoke detection by image processing*

Depending on whether fire and smoke detection techniques primarily focuses on flame or smoke characteristics, then we have either video image flame detection or video image smoke detection. Both detection methods mostly use color, motion, energy, and other disorder information, although the lack of consistency in the fire and smoke pattern and the features of those like color, shape, and transparency, can cause false alarms. Regarding fire detection, one of the initial fire detection techniques was color detection, which used a color-based detection mechanism. Then there was the moving object detection, based on the detection of possible motion. Other techniques that appeared later were flicker detection and energy analysis, both based on the temporal behavior of flame and smoke, the spatial difference, based on spatial color variation of fire, the smoke disorder analysis, based on the variation of turbulence, the boundary roughness, and the randomness of area size. Also, there are three modules which are used in the detection process, sub-blocking, for the reduction of measurement disturbances, training, for the creation of the background and the color models, and clean-up post processing, for the removal of outliers and the grouping neighboring elements [8].

	Color Detection	Moving object detection	Flicker \ Energy Analysis Temporal difference analysis	Spatial difference analysis	Disorder Analysis	Subblocking	Training	Cleanup post-processing	Localization / Propagation	Flame Detection	Smoke detection
W. Phillips et al.	RGB		x	x			x	x		x	
T.-H. Chen et al.	RGB/HSI	x			x					x	x
J. Ebert, J. Shipley	RGB/HSV		x	x				x		x	
Z. Xu, J. Xu		x	x		x					x	x
P. Piccinini et al	RGB	x	x				x				x
T. Celik et al.	YCbCr/RGB/HSV									x	x
Z. Xiong		x	x		x						x
G. Marbach et al.	YUV		x		x					x	
R. Yasmin	RGB/HSI	x			x	x			x		x
F. Gomez-Rodriguez et al.		x	x		x						x
P. V. K. Borges et al.	RGB				x					x	
C.B. Liu, N. Ahuja	HSV		x		x					x	
B.U. Toreyin et al.	YUV	x	x		x						x
B.U. Toreyin et al.	RGB	x	x	x						x	
T. Celik et al.	RGB	x			x		x	x		x	
B. Lee, D. Han	RGB	x					x	x		x	x
S. Calderara et al.	RGB	x	x			x	x				x
F. Gomez-Rodriguez et al.		x	x		x						x

Table 4.1. *State of the art VFD techniques on relevant papers*

4.2.2 Smoke detection models

There have been many methods to detect fire and smoke based mostly on color, motion and other space and time attributes. These methods can apply to either flames, smoke or both combined. The reason for the various attributes used in different methods of detection is caused due to the very specific characteristics fire has, in comparison to other objects. Regarding smoke detection we have Stationary Wavelet Transform (SWT), Color Features and Motion Detection, Finding Feret's Region, Covariance Descriptors and Dual Dictionary Modelling. Smoke Detection based on SWT functions in three parts. The image is resized in two dimensions, converted to grey scale, and then indexed to group colors of similar intensity. In the second part SWT is used and since smoke turns to smooth an image, only high frequency components are deleted while the rest unchanged parts under smoke, are reconstructed by applying the inverse SWT. Two indexed images of different decomposition levels are saved and then performed the detection algorithm. The Region of Interest (ROI) of those four images, grey scaled and indexed, is picked. This selection of the high intensity area is done by matching up the images and choosing the common pixels, which then are compared to pixels of non-smoke frames. The final part is to use the smoke verification algorithm to the common pixels, which form the smoke region, in order to detect smoke and eliminate false alarms [1].

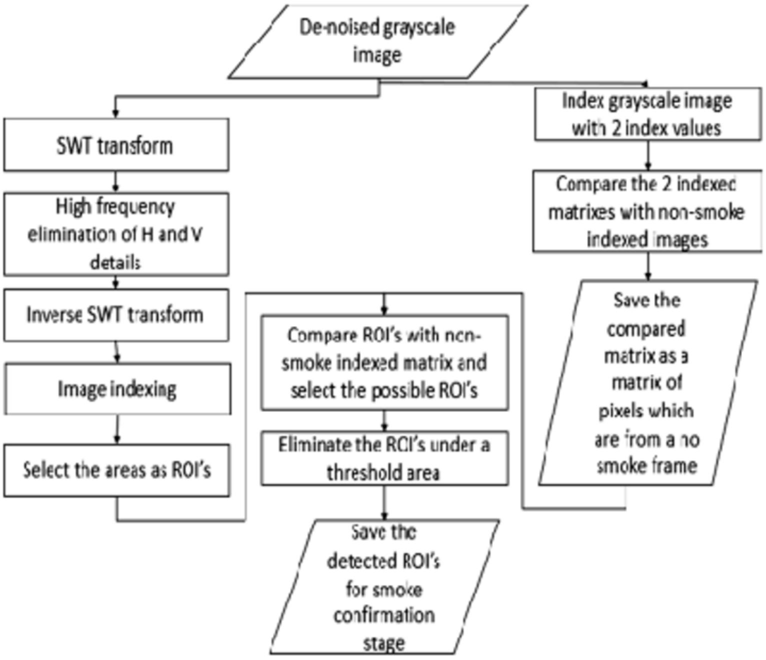


Fig. 4.5. *Smoke detection algorithm using SWT*

In the smoke detection method using Color Features and Motion Detection, pixels are split into fire pixels, smoke pixels and background pixels and then clustered. The classification is done based on the difference of the behavior of those features, with the use of five modules. Change Detection and Motion Detection first, in order to eliminate background pixels. Then Fire Feature Extraction based on YCbCr color space and Smoke Feature Detection based on LAB color space, to identify pixels which contain fire and smoke. Then in the Region Growing these pixels are united into big regions, which are sent to the Chaotic Motion Analysis. Finally, a data fusion process is applied using multi-layer perceptron (MLP) to erase false alarms [1].

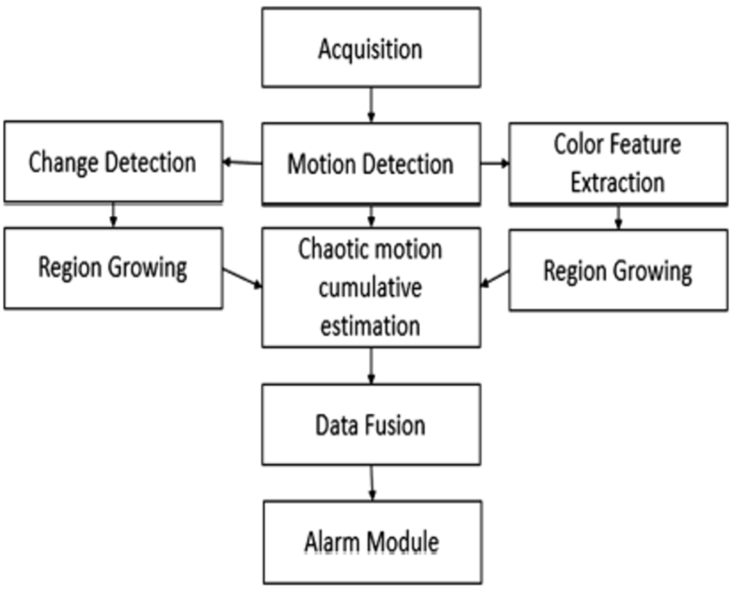


Fig. 4.6. *Smoke detection algorithm using Color features & motion detection*

In the next detection method, Feret's Region is the shape and region of the object which is found with the use of the Feret's Diameter, the distance of its opposite sides. We need first to isolate the moving objects, which happens in the next five parts. Image subtraction and accumulation, to find the proper region of the moving parts and collect subtracted images. Image binarization, to turn it into a binary image. Morphological operation, to exclude noise in the binary image. Extraction of Feret's diameter, to get the object's shape and finally the use of an image mask to calculate the region of the moving object inside the frame. These moving objects are classified as smoke or not smoke regions, by Support Vector Machine (SVM) which is a learning module, trained by learning algorithms like the previous ones. SVM can also classify fire based on covariance features which are computed from spatial-temporal blocks. In the first place, 3D regions are extracted from images, and then covariance descriptors with the use of color masks and other spatial and domain information generate these blocks [1].

Smoke in comparison to fire or haze is a scattered part in an image. Smoke detection using Dual Dictionary Modelling is used to represent smoke in each block. Sparse representation generates data as a linear combination of points deriving from a dictionary of elements. Two dictionaries are trained independently with real samples, and they classify pure smoke images to the smoke dictionary and clear images to the none smoke dictionary. Additionally, are used regularizations parameters, a threshold to check convergence and the initial block image [1].

Article name	Author	Methodology	I. Pros II. Cons
"Wavelet-Based Smoke Detection in Outdoor Video Sequence," 978-1-4244773-9/10/$26.00 ©2010 IEEE	R. Gonzalez-Gonzalez, V. Alarcon-Aquino, R. Rosas-Romero, O. Starostenko, J. Rodriquez-Asomoza	Uses Stationary wavelet transform	I. Robust and reduces false alarms II. Not as effective in closed spaces
"Early Fire and smoke detection based on color features and motion Analysis" 978-1-4673-2533-2/12/$26.00 ©2012 IEEE	Pietro Morerio, Lucio Marcenaro, Carlo S. REgazzoni, Gianluca Gera	Color feature extraction	I. Good range II. Not effective in low light
"A novel Detection Method Using Support Vector Machine," 978-1-4244-6890-4/10/ $26.00 ©2010 IEEE	HIdenoriMaruta, Akihiro Nakamura, Fujio Knrokawa	Using Feret's Region to obtain possible regions of smoke	I. Exact region of smoke is found II. Not efficient when there are discontinuous regions of smoke
"Flame Detection method in video using covariance descriptors," IEEE International Conference on Speech and Signal Processing, pp. 1817–1820, 2011	Y. Habiboglu, O. Gunay, A. Cetin	Covariance descriptors are used for texture classification	I. Lesser computational cost and is not affected by the random behavior of fire II. Efficient only when in close range and visible clearly
"Detection and Separation of Smoke from Single Image Frame," 1057-7149 ©2017 IEEE	Hongda Tian, Wanqing Li, Philip O. Ogunbona, Lei Wang	Dual dictionary technique	I. Differentiate smoke from fog and hue II. Highly Complicated

Table 4.2. *Table comparing the methodology, pros/cons of proposed methods*

4.2.3 Visual descriptors

In most of methodologies, we see the same similarities in the detection process where it is essential to use a large number of datasets, which results in the creation of a huge amount of visual content to be processed. This visual information is described by a vector, which is produced in a standardized way. As visual descriptor is defined the algorithm which has as input the visual features of an image and as output the vector. As the requirements to describe an image differ, so do the algorithms that produce the characteristic vectors. Visual descriptors can produce descriptions for the whole image or can extract information for specific domains like an object or event in the scene. Such descriptors can be used in many fire and smoke detection and classification approaches. In Covariance approach, spatial and domain data are used along with covariance descriptors, to detect smoke close to the camera. Mehdi Torabnezhad approach is based on vision and thermal information to effectively detect smoke. Fractal encoding method, apply fractal encoding techniques to detect smoke images when smoke regions are not clear. Another approach for smoke is based on machine vision for image quality improvement, where a hyper parameter of support vector is established for online identification. As far as an approach for flame identification and classification, there is the extraction of statistical and spectral features of the flame in gasifiers, in order to achieve better analysis of the flame characteristics [9].

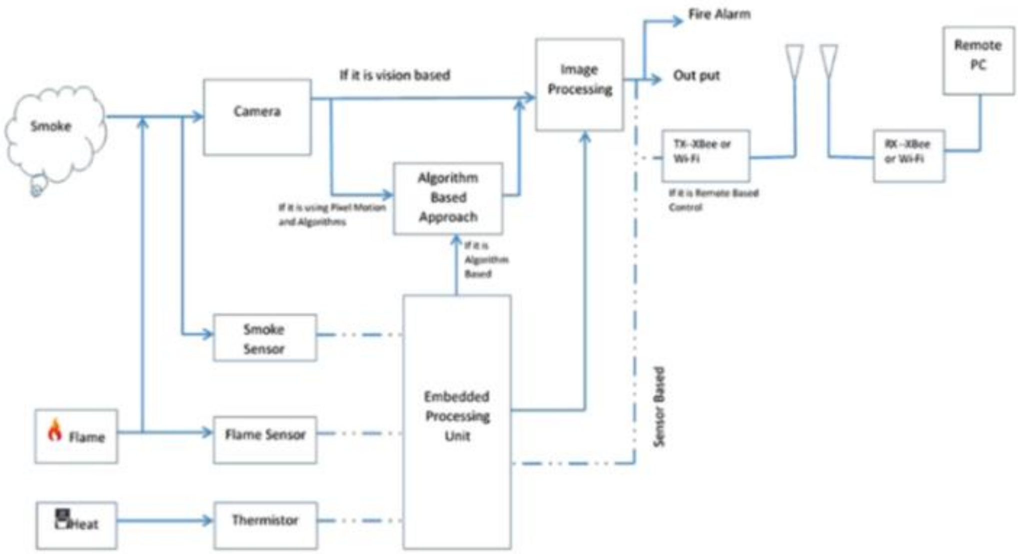


Fig. 4.7. Block diagram of different types of smoke and fire detection approach

There are many descriptors that cover various divergent local features. Some of which are color, texture, shape, motion, and location. Color and texture are the most widely used features for detecting and recognizing objects. Both features, which can be calculated efficiently, are suitable for real-time systems. Color is used in almost all fire detection methods as a separate feature, but in a different color space. There are also other color areas such as YUV (intensity of brightness with red and blue projections), HIS (hue, saturation and intensity), HSV (hue, saturation and value) and LAB (lightness with two color components as dimensions), however there is not many research on theoretical analysis and empirical comparison of color selection, but from the current status LAB histogram is the one that usually has the best performance in many SVM kernel settings [11].

	dense sampling	keypoint sampling
SURF	**84.8485**	76.431
RGB + SURF	**86.1953**	79.1245
HSV + SURF	84.1751	**85.1064**
LAB + SURF	**92.2559**	84.8484
YUV + SURF	**85.1852**	80.4713

Table 4.3. *Precision comparison of sampling strategies*

Texture is another important local distinctive feature. Many texture descriptors like Scale Invariant Feature Transform (SIFT) and Histograms of Oriented Gradients (HOG) have been widely used with success in image-based matching and in many different object and scene recognition under real-world conditions. But their high computational complexity sets limits to their implementation into monitoring and data retrieval. To ensure that detecting systems are functioning in real time, Speeded Up Robust Features (SURF) descriptor are adopted in many methods. SURF is pretty similar to SIFT, they both operate only on gray scale images, also regarding efficiency, as it is a feature vector of responses in the close area of the point of interest but is much quicker as far as computations. Moreover, SURF is parallel computing, so it can benefit from systems with multi core processors and GPUs. SURF-128 is an extended SURF descriptor, that instead of the standard 64-dimensional SURF, it has 128 dimensions, and in general it is more individual but much slower in the matching step [11].

	64-dimension	128-dimension
SURF	84.8485	**85.5218**
RGB + SURF	**86.1953**	85.5218
HSV + SURF	84.1751	**86.8687**
LAB + SURF	92.2559	**92.5925**
YUV + SURF	85.1852	**86.532**

Table 4.4. *Accuracy comparison of SURF and SURF-128*

4.2.4 Proposing, classification, and verification of fire region

The fire detection system consists of three main parts. First, the candidate fire areas are proposed as a background/font model (candidate fire region proposing), then fire areas are classified by characteristics of color, texture, and dictionary of optical words (feature extraction and region classification) and finally, we have the time verification of smoke and fire (temporal verification). In the candidate fire region proposing, systems detect fire areas or pixels directly. Based on observations and statistical experiments, areas of fires are relatively light, and the values of pixel intensities are higher than a certain limit, which later assists in finding new fire region candidates and building the fire detection process. In the initialization stage of the region classification, a codebook is used to construct a K - D tree, which allows features to be quickly found and searched. For each incoming blob, an extended sampling is performed, producing the descriptors. Objects such as streetlamps and car lights are similar in appearance to fire and are difficult to exclude based on a single frame. Therefore, the blobs classified as fire in the previous step should be further verified by statistical time variation, since fire mainly shows variation only in shape, not in other features over the time. For each new blob, the stability of the area is calculated in consecutive frames, without having to calculate its exact representation, by using three parameters, perimeter, area, and spatial distribution. In the experiments conducted in real time fire detection applications, there can be used either key point sampling methods in both codebook construction and classification, or dense sampling [12].

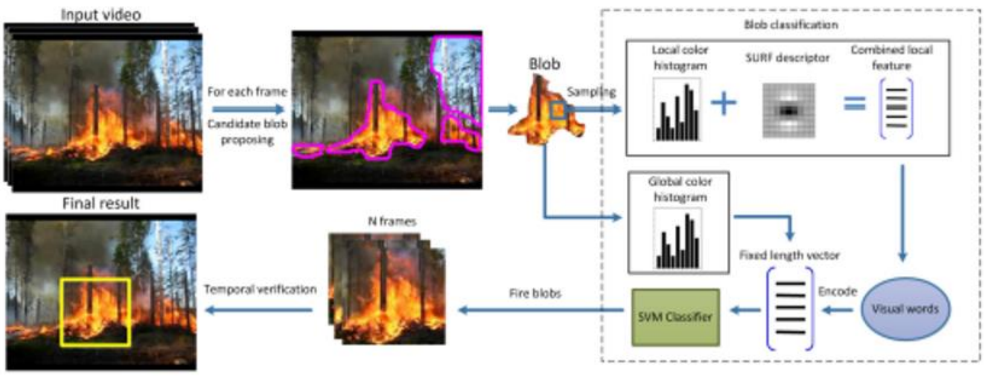


Fig. 4.8. *Framework of proposed fire detection system*

The sorting processes initially detect and isolate moving pixels by using algorithms to remove background, and then they apply color models to find the color areas of the fire. Then these areas are further analyzed spatially and temporally in order to detect the irregular and trembling characteristics of the fire. Since motion is the main feature, these methodologies can only function properly with steady cameras in surveillance scenarios. Most of the models use image level evaluation, the process where image is evaluated if it meets the criteria to identified and sorted as fire or not fire. Some of them also refer to patch level detection accuracy, where algorithms automatically extract related attributes from the output of a detection system. Both the accuracy of the image level classification and the correction are considered evaluation criteria on the frame level [4].

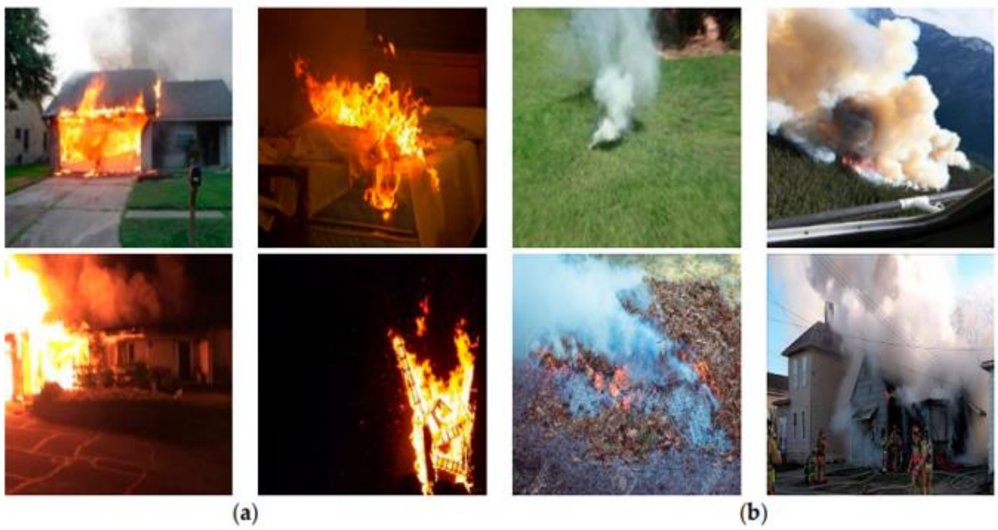


Fig. 4.9. *Examples of images showing fire(a) or smoke(b)*

4.3 Image and video algorithms

4.3.1 Optical systems

Furthermore, video and image algorithms appeared to deal with irregularities of the previous methods either on static or dynamic rules. Video based fire detection algorithms are introduced to the most visible features of fire. Initially mostly about the flames, and in a second time mostly towards to the smoke. This happened because in most of the occasions smoke in comparison to flame is produced first, so it can prove to be a significant sign of early fire. Besides, it can be detected from great distance, and it spreads faster. The autonomous forest fire detection system uses black and white cameras to detect temporal differences of smoke. Other optical systems use intelligence analysis of the atmosphere. In general, color and motion models are used in detection algorithms, and then can be also implemented in both fixed and pan/tilt/zoom cameras(PTZ). Firstly, trained color models along with pixel-wise temporal analyzers transform the color space of the video data, determine the candidate regions for fire and reduce fire like false alarms. Then these systems make the temporal and spatial analysis of the optical flow vectors, modeling the motion information. Afterwards, combined color and motion models determine what is the fire probability, and finally the decision made about the presence of fire or not, is made only if the above possibility of fire is higher than the predefined criterion in the system [10].

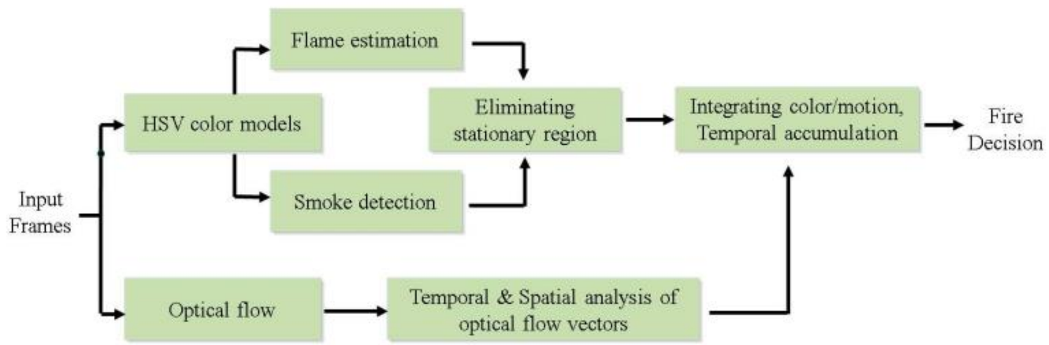


Fig. 4.10. *Overall structure of proposed fire detection algorithm*

Various detection sensors were introduced into many optical systems, such as 360° rotating cameras, thermal cameras based on the fire heat flow, raging systems and light detectors, black and white color camera detectors, color day vision cameras, ultralow light gray scale night vision cameras, cameras sensitive to smoke, clustering motions in smoke detectors, IR spectrometers for smoke and detectors for chemical composition in the atmosphere. These systems beside the sensors and cameras, in order to function are produced along with other parts while software is developed with various algorithms which process the input data into output decisions. These internal system parts include power systems, weather stations, monitors, GIS equipment, communication units like microwaves, satellites, 3G, 4G, 5G, data transfer units and receivers. As it proved, such systems did not prove to be as trustworthy as they should, based on the high cost they have. The topology and topography of the covered areas is hard to be calculated with minor localization errors. There are many false alarms due to various weather conditions that cause moving columns of smoke, cloud shadows or wind tossed trees. Another reason is the big topology differences, cloud motion, atmosphere variations and day and night changes that various landscapes have worldwide, so it is very difficult to design specific patterns in order to detect smoke and fire. Furthermore, big parts of the landscape are out of sight because mountains, hills, and trees are blocking the optical systems to have a clear view of fires [13].

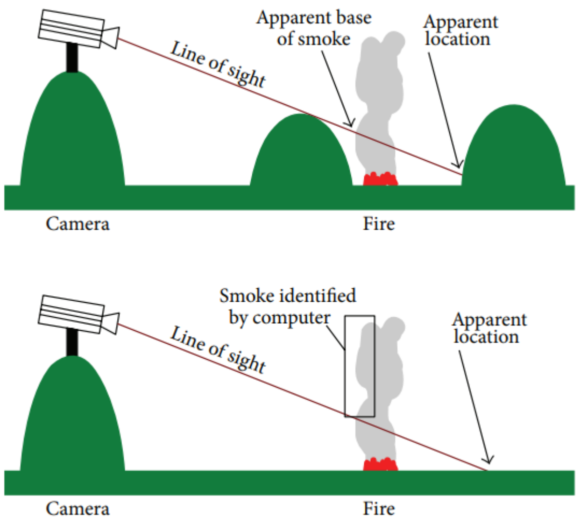


Fig. 4.11. *Localization errors*

4.3.2 Sensor networks, CCTV and GIS systems

Wireless Sensor Networks(WSNs), an IoT node arrangement, consist of circuit boards, wireless transmitter and receiver, memory, buffer, network of sensors computing data, IP cameras, and batteries, all combined together to conform to algorithms and operate in a self-organized and self-healed environment. Physical information is gathered, computed, and processed into data, while this system also reacts to environmental changes. Such environmental parameters are temperature, wind speed, humidity, pressure, nitrogen dioxide, carbon monoxide/dioxide and various others. These are transformed into electrical signals and sent asynchronously in many locations to be deployed by different applications. WSNs can provide at any time with high accuracy and reliability all relevant information about environmental conditions with minimum delays. Since they can be scalable, they afford to cover from small to huge size areas. Furthermore, they have big connectivity and deployment advantages since they link to numerous low-cost devices and nodes in order to observe and track down numerous various physical parameters, without the need of highly expensive communication channels or complicated detection methods and devices. On the other hand, this technology requires a huge number of sensors deployed in order to have some reliable results, so it’s use shows some specific disadvantages. It functions only in places where sensors have been deployed so it cannot perform in unknown places and provide full coverage. All devices have a specific life circle since they depend on batteries, making it impossible to replace or recharge all of them in a wide cover area. Finally, it cannot exactly predict or detect with maximum precision and reliability [13].

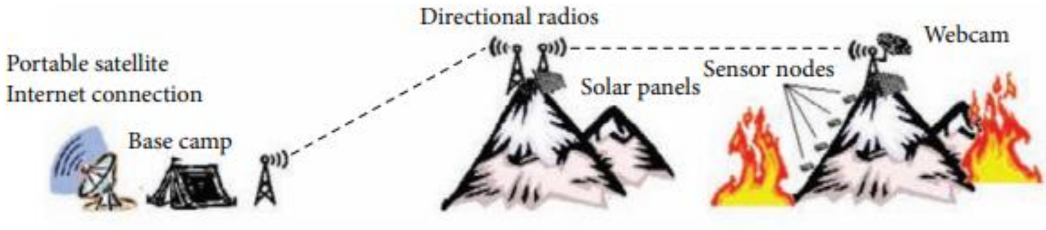
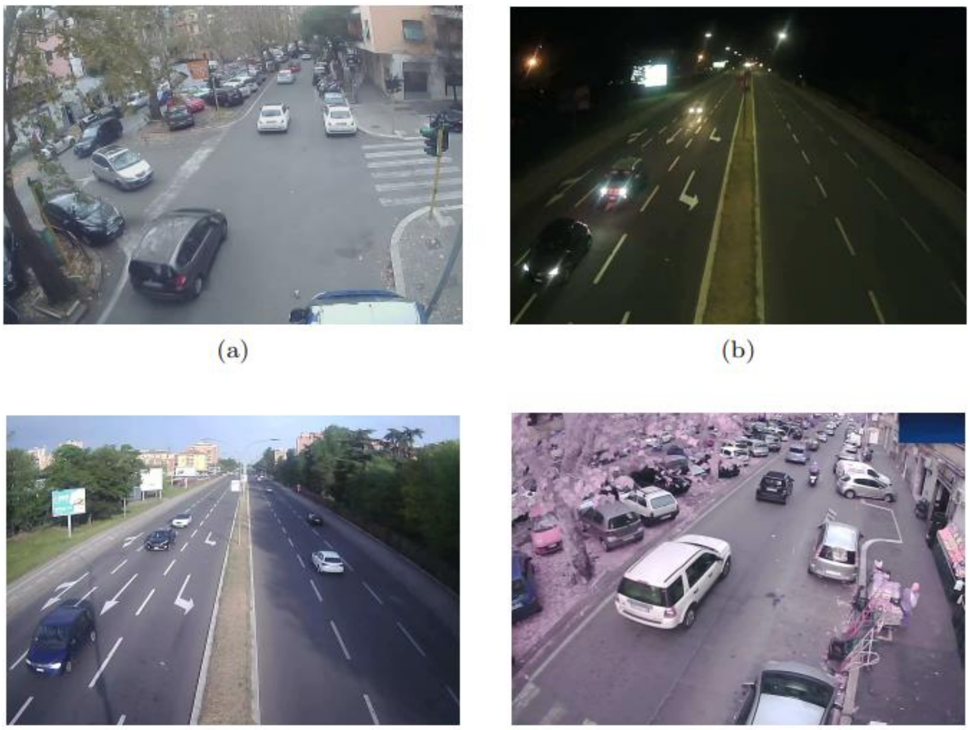


Fig. 4.12. *Wireless sensor network*

As Closed-Circuit Television (CCTV) systems are installed at indoor and outdoor large monitoring areas, videotaped fire detection techniques are suitable for analyzing the behavior of fire and smoke, while performing a three-dimensional detection. These algorithms take decisions based on visual properties of fire and extract structural and statistical features. Some of those properties are color and motion features, time and shape aspects, edge blurring and flicker, flow direction and magnitude, GIS (Geographic Information System) and geometry of smoke regions etc. But they mostly operate on moving pixels based on steady cameras. The dynamic growth of fire and smoke, and the flexibility in shape and unfixed structure, set severe limitations to the conventional fire detection methods as they rely on limited features. Also, the color of fire or smoke as feature, is heavily affected by the quality of the camera. For instance, due to sharpness, resolution or white balance. So further resources are needed to design additional algorithms to track the close or long range of the identified smoke areas. These limitations of these methodologies lead to failures, unsolved problems or provide detection solutions with poor performance results [2].



(a) (b) (c) (d)

Fig. 4.13. *Four sample images of outdoor areas from CCTV*

Pure satellite-based solutions had also been deployed, in order to provide uninterrupted and full satellite coverage. Either orbiters or air floating devices, have been designed to do a variety of different functions, so it was not difficult to add another extra functionality, of fire detection, to the existing ones without increasing the costs significantly. Although programming another function was a simple task, the lack of equipment suitable for fire and smoke observation, such as transponders, antennas, and downlink transmission, could affect the efficiency. Besides, laws, legal agreements and regulations between different countries, could create further obstacles in the detection process via satellites [13].

Comparison	Human based observation	Satellite system	Optical cameras	Wireless sensor networks
Cost	Low	Very high	High	Medium
Efficiency and practicality	Low	Low	Medium	High
Faulty alarms repetition	Low	Low	Medium	Medium
Fire localising accuracy	Low	Medium	Medium	High
Detection delay	Long	Very long	Long	Small
Fire behaviour information	—	Yes	—	Yes
Can be used for other purposes	No	Yes	No	Yes

Table 4.5. *Comparison of proposed systems*

4.4 Learning architectures

4.4.1 Object detection and feature extractors

Conventional fire and smoke detection methods based on video face the problem of relying on expertise to build relevant feature extractors. Experts are needed, for manual intervention, to build rule-based models and distinctive features. A different approach to standard datasets is to use a learning algorithm for training forest fire and smoke detector, where a fire detection benchmark is built. The method in that approach is to extract the useful features instead of using a system to generate them, along with computer vision-based detection techniques which can operate with either steady surveillance systems or non-stationary cameras. Deep learning architectures can learn and extract such useful features for forest fire and smoke video detection. Convolutional neural networks are a variant of deep learning that can extract topological properties from an image. Such methodologies can be fully automated and more over designed to offer effective generalization for unseen datasets [3].

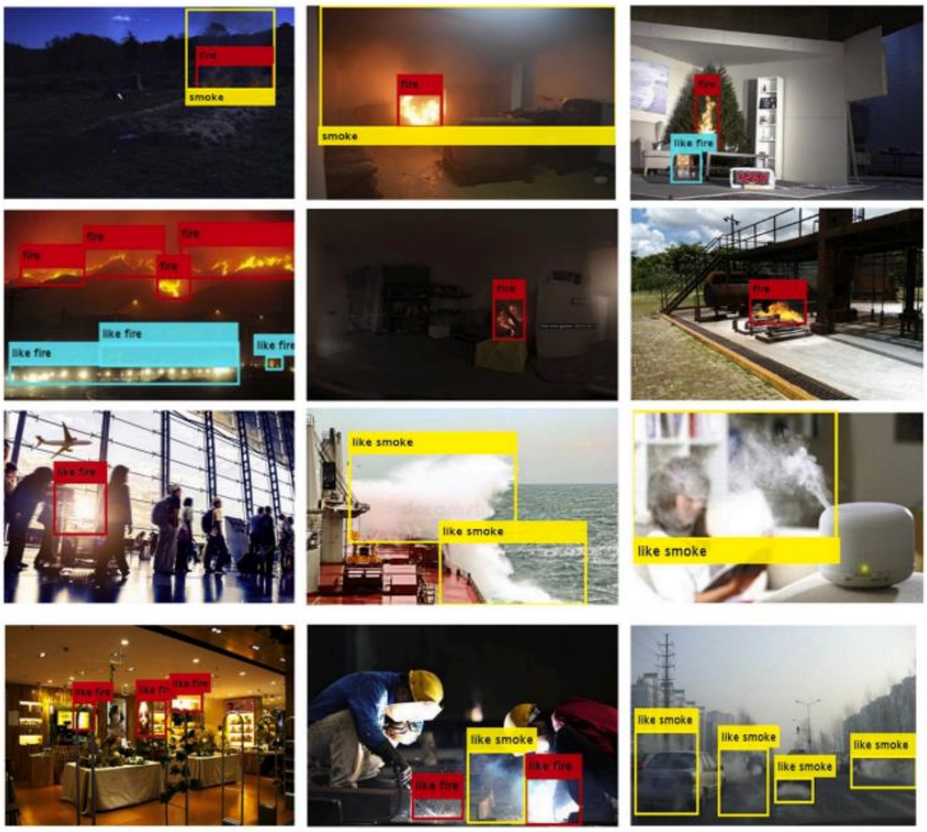


Fig. 4.14. *Images in dataset classified as 'fire/smoke' & 'like fire/like smoke'*

The change from conventional sensors and image processing to computer vision techniques brought revolutionary results to object detection. Object detection is the process that predicts inside the image where each object stands and what specific label should be assigned and can be broken down into machine learning-based networks and deep learning-based networks. In more traditional machine learning approaches, objects are being identified by moving pixels and regions which are extracted and grouped based on image properties using various models, further analyzed, and then fed into a regression model predicting the location and label of the object, all the above relied on machine learning software image recognition software and cloud workflows for importing and scanning images. A further step is the creation of a convolutional layer that uses as detector a dilated convolution operator, where subsets of data are used and evaluated for the learning process. Such applied methods can be an adaptive piecewise linear unit (APL units), proposed by Abdulaziz and Cho, or relevant methods with common CNN architectures, such as AlexNet, SqueezeNet, GoogleNet, MobileNetV2, VGG16(Very Deep Convolutional Networks for Large-Scale Image Recognition) and VGG19 [3].

CNN	Accuracy	Classification Time	Training Time
AlexNet	94.8 %	7.743 sec	3.5 hours
GoogLeNet	99.0 %	11.657 sec	3.1 hours
Modified GoogLeNet	96.9 %	10.017 sec	1.5 hours
Modified VGG13	96.2 %	7.951 sec	6.05 hours
VGG13	86.2 %	10.215 sec	17.7 hours

Table 4.6. *Evaluation results of methods with common CNN architectures*

4.4.2 CNN Architectures

In the three main parts in the fire and smoke image detection algorithms, image preprocessing, feature extraction and fire detection, the second is considered the main part of the algorithm. Firstly, the detection CNN takes as input an image and via pooling and convolution it processes it, and then gives as output the region proposals. Then the region-based detection CNN decides on those region proposals, if there is absence or presence of smoke and fire, via convolution layers, fully connected layers, pooling layers and others. The convolution layer inside the feature extraction, which is considered the core of the algorithm, is a set of convolution kernels which are the image transform filters responsible for the generation of feature maps of the original images. In comparison to the traditional algorithms, where there happens manual selection of fire features and machine learning classification, the modern image recognition algorithms based on CNNs, can automatically learn and extract complex image features with great success in real scenarios. These algorithms aim to achieve high detection rate, minimum false alarms, high speeds in image processing and overall fast computation speeds, but at the same time, and since they require large disk sizes to operate, they face the challenge to keep as low as possible the costs when they are implemented into embedded systems or they are deployed on low cost IoT devices [5].

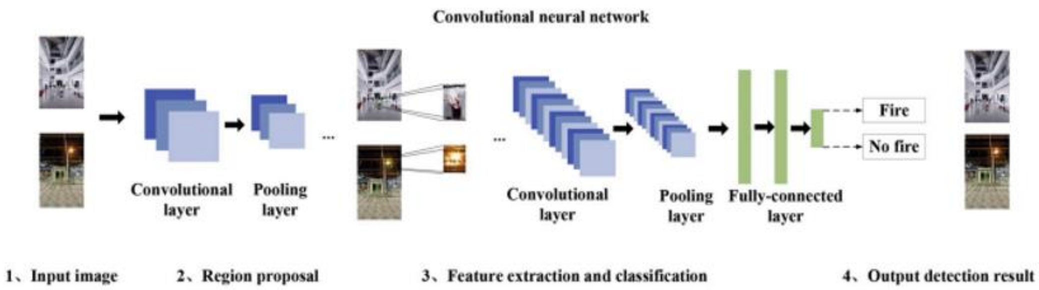


Fig. 4.15. *Flow chart of image detection algorithms based on detection CNNs*

Such image detection algorithms, that are based on advanced object detection convolutional neural networks, are Faster-RCNN, R-FCN, SSD, You look only once (YOLO) v2 and YOLO v3. A standard preprocessing procedure of such algorithms can consist of the following steps. Load the collection of relevant fire and smoke images resized, label the fire and smoke objects of interest via custom algorithm tools and export those data to a workbench to obtain arrays of labeled features. Afterwards in the training process, the related fire and smoke classifier is trained, using annotated samples from scratch or apply a fine-grained classifier to find the precise location in the image, and then the epochs are defined. The classifiers can be linear or nonlinear, and the epochs are the number of times that the algorithm will run the training dataset, where it is important to keep an optimal number and avoid having the model overfitting and overconfident in its predictions. Then follows the validation process, where in a separate test bench, a new dataset of various resized images is provided, and have them each validated as image or as patch with presence or absence of fire and smoke [6].

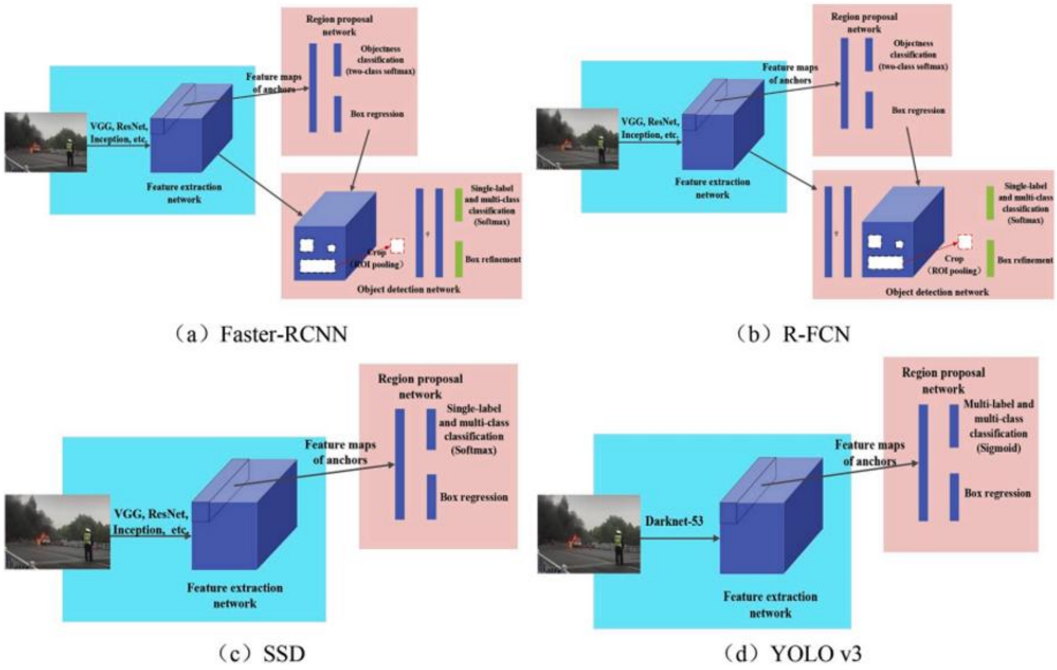


Fig. 4.16. Diagrams of fire detection algorithms based on above CNNs

Next step is the experiment, which is carried out in testing datasets of many videos in the relevant variety of indoors and outdoors conditions, and accordingly comes the final classification. At the end, the efficiency of all approaches can be evaluated, and the effectiveness is measured, in order to compare all the detection algorithms [6].

No.	Class	Algorithm		Missed detection rate (%)	False alarm rate (%)	Accuracy (%)
1	Manually extraction features	Chen		11.76	14.29	87.10
2		Celik		29.41	0	83.87
3		Raffiee		17.65	7.14	87.10
4		Habibuglu		5.88	14.29	90.32
5		De Lascio		13.33	0	92.86
6		Foggia		11.67	0	93.55
7	CNN	Muhammad	(Alexnet)	9.07	2.13	94.39
8		Muhammad	(GoogleNet)	0.054	1.5	94.43
9		Faster-RCNN		0.018	0.69	99.43
10		R–FCN		0.018	0.97	99.20
11		SDD		0.036	1.29	98.93
12		YOLO v3		0	0.46	99.62

Fig. 4.17. Comparison of different fire detection algorithms

There are modern deep learning networks which can perform end-to-end unsupervised detection, recognize patterns, and make decisions under uncertainty. The existing approaches of convolutional neural networks, propose various solutions like joined CNNs, deep CNNs with different classifiers, deep normalization and CNN model, CNN model for image classification at a different time or simultaneously. IoT/Edge devices equipped with cameras can be deployed strategically throughout hillsides, ridges, and high elevation areas, automatically monitoring for signs of smoke or fire. Drones and quadcopters can be flown above areas prone to wildfires, strategically scanning for smoke. Satellites can be used to take photos of large acreage areas while computer vision and deep learning algorithms process these images, looking for signs of smoke. Deep learning models have better performance than all existing methods, are more robust and offer object localization and object classification, but at the same time have big needs in training datasets and computation power of GPUs [4].

4.4.3 UAVs

Besides the standard approaches of detection in less challenging environments via fixed features and surveillance, since fire is a bending object with dynamic shapes, there can be also other approaches with higher difficulties like moving cameras on unmanned aerial vehicles(UAV) for high-scale forest fire detection. In such methods, deep learning approaches for training detectors are used, where the fire detection systems are mounted on UAVs. Despite the challenges, fire detection systems using UAVs which are developed using image and video recognition applications, on which deep CNNs have been implemented, can be a cost-effective solution against the wildfires and also overcome various limitations of other technologies. UAVs can provide up to date high spatial resolution aerial photographs and achieve high accuracy even in wide range images. Cameras are mounted on the bottom of the UAVs and are orientated to look towards the ground with an anterior specified angle, allowing them to avoid cloud obstacles [7].

Fig. 4.18. *Aerial photographs taken from a quadcopter in fire experiment*

In these approaches, fire detection rules can be designed for detecting and segmenting fire regions in the visual images taken from the onboard camera of the UAVs. Such an approach is the wildfire detection system, which involves joined deep convolutional neural networks (CNNs) where a full image and a fine-grained local patch fire classifier can be trained, and it utilizes UAVs. Firstly, the image is processed by the global image-level classifier and if the fire is identified, then the fine-grained patch classifier comes along to detect the exact location of fire patches. Such a detection architecture can mainly consist of a UAV Control System, an image analysis system where such a joined deep CNN is implemented, a disaster forecasting system, a web-based visualization system, an alert system, and a disaster response scenario database [19].

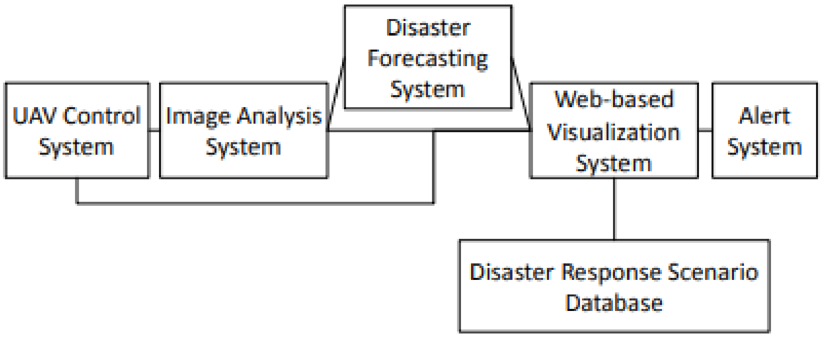


Fig. 4.19. Architecture of wildfire detection system

Accordingly, there are similar approaches for smoke detection systems. In order to drastically reduce false alarms due to environmental variations, different light conditions, background colors, supervisory fuzzy logic-based smoke detection methods are implemented. Firstly, in the processing phase, many images are captured from the digital camera in the bottom of the UAV. The camera used is a calibrated camera, in order to significantly reduce the deformation of the images caused by the distortions of the lens. Afterwards, the specified detection rule takes as input the RGB and HIS differences, and the smoke likelihood as output. Then an Extended Kalman Filter (EKF) is implemented to use both input and output of the smoke detection rule and assists with further regulation flexibility, and finally there is the segmentation of the smoke regions [22].

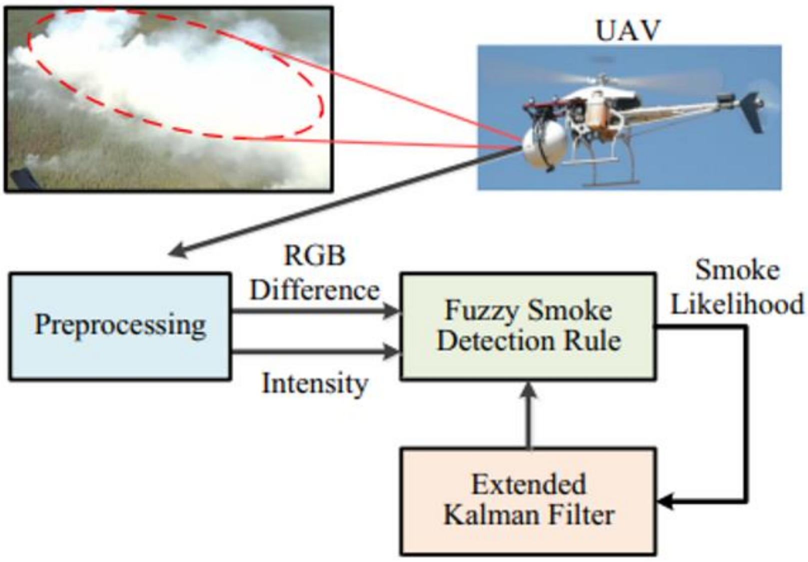


***Fig. 4.20.** Illustration of proposed learning-based smoke detection scheme*

4.4.4. Deep neural networks along with other technologies

Deep neural networks (DNN) are a powerful technology for solving problems in various fields and are also implemented along with computer vision in artificial intelligence applications and various other related machine learning adaptations. Moreover, they can be used together with IoT, in order to analyze the data generated by IoT devices, and optimize processes, improve operations, and make predictions. However, it is a difficult task for complex DNN based processes and complicated

DNN based activities to run on mobile devices, and that is where edge computing is introduced in order to assist with these issues by speeding DNN interference over Edge computing by Edge AI. Preserving Edge is a dynamic filtering approach for video images, as those filters, with their unique independent algorithm, are important tools for numerous tasks in image processing and transformation, while also Edge technology uses some of the DNN design features, like DNN partitioning and DNN right sizing[16].

CHAPTER 5

FIRE AND SMOKE DETECTION SYSTEMS WITH EDGE COMPUTING

5.1 Cyber Physical Social System

In order to confront shortcomings of previous adaptations, like limited energy resources and scarcity of computation capabilities, edge computing technology is implemented to provide computation resources to IoT devices, like a cluster of servers. The integration of fire and smoke detection systems with edge computing technologies can prove to be a revolutionary technique, that involves the deployment of sensors and cameras at the edge of a network where fire and smoke are detected, and AI algorithms which make the data analysis and decide on the presence or absence of fire in various environments. This analysis is done locally on the edge nodes or on nearby edge devices, rather than be sent for processing to a centralized unit, server or cloud. Overall, fire and smoke detection with edge computing is a promising technology that can improve the speed, accuracy, scalability, response times, and reliability of fire detection systems, while also reducing the cost, the energy consumption, the latency and bandwidth requirements of the system, and complexity of deploying and maintaining such systems, while also improving security and privacy[30].

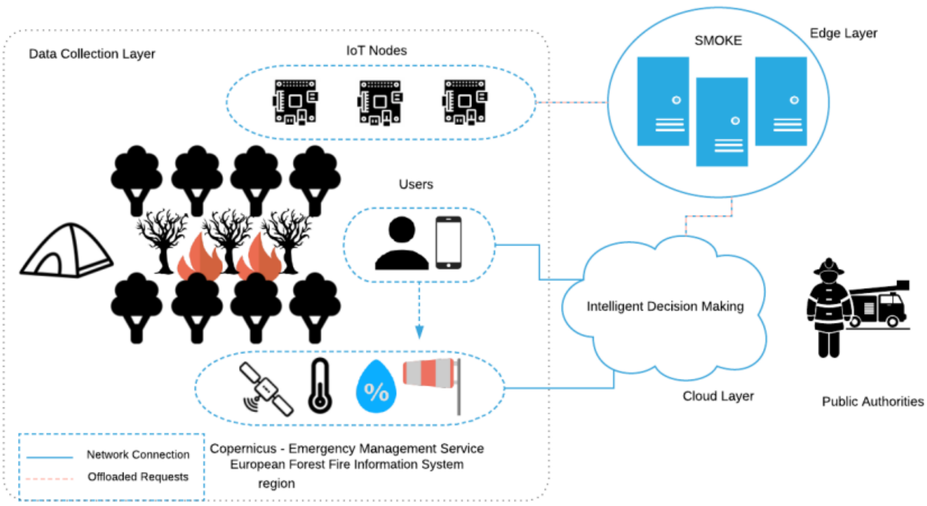


Fig. 5.1. *CPSS Architecture*

A system designed to adopt those emerging technologies into one entity, is the Cyber Physical Social System(CPSS) for early fire detection, which represents the integration of computer systems, physical world, and human parameter, consisting of three different level. At the bottom level, the architecture consists of static or mobile IoT nodes, which monitor and sense forest areas, while they also collect information from the environment, in order to detect fires in the early stages with the use of embedded cameras. Next in the middle level, a dynamic resource scaling mechanism for the Edge computing infrastructure called SMOKE(Scalable edge coMputing framewOrK for early firE detection), deployed at the edge of the network at a cluster of servers and which hosts two image classification services, is responsible to collect the snapshots taken from the IoT nodes and make the data processing. Finally at the top level, a cloud based decision-making algorithm, integrates previous classification results, user information on social media, weather information services and other related information, and then addresses the QoS requirements, evaluates the criticality of the fire conditions on time, and at last if needed notifies the relevant local authorities in case of emergency[30].

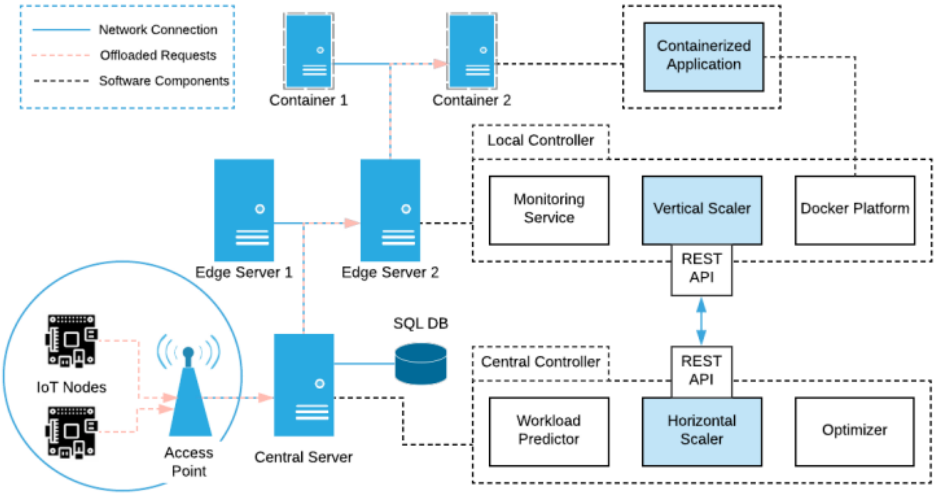


Fig. 5.2. SMOKE Architecture

5.2 AI Edge Inference Platform

A mechanism which serves the purpose to predict fire and smoke based on real world events, using trained DNN models, is AI Inference Platform. For the purpose of this platform, embedded sensor systems based on microprocessors are implemented and mounted on poles in outdoor environments, where these are equipped with 5G chipsets and WIFI modules, so they connect to the network, and are protected by waterproof equipment and thick metal cover against the sun. An embedded system is a device based on microprocessors, with each own components like integrated memory, camera, wireless communication modules or various other peripheral interfaces. In order to increase the performance of the hardware of those devices, vision processing units(VPU) are enabled. VPUs are AI microprocessors that interfere with machine learning models such as CNN and accelerate relevant tasks. After finishing the training on the DNN and getting the successful results, the trained model gets real time data from the IoT devices as input and based on those, locally on each device, it makes the predictions on the output, a process called Inference and more specifically on-device edge inference [28].

Fig. 5.3. Embedded System with Camera Sensor Mounted for Outdoor Use

Open Visual Inferencing and Neural Network Optimization(OpenVINO) is a toolkit, that mainly optimizes CNN-based deep learning inference at the edge, feeding pre-trained models to its own Model Optimizer and can be used alongside with Inference Engine which assist in the inference speed by the proper execution of the model on the edge devices. The Model Optimizer is a tool responsible for the transit from training environment to the development environment. It loads, reads, executes, and optimizes the model, and as a final step produces a valid and optimized Intermediate Representation, which is the format that can be accepted by the Inference Engine. The last, takes as input a set of rules and knowledge representation, and then by applying logical reasoning to the input, produces an output that may include new conclusions or actions. The sensor uses a script for periodic detection, captures live frames, and processes the images through the Inference Engine in order to bookmark the event as fire and smoke or not[28].

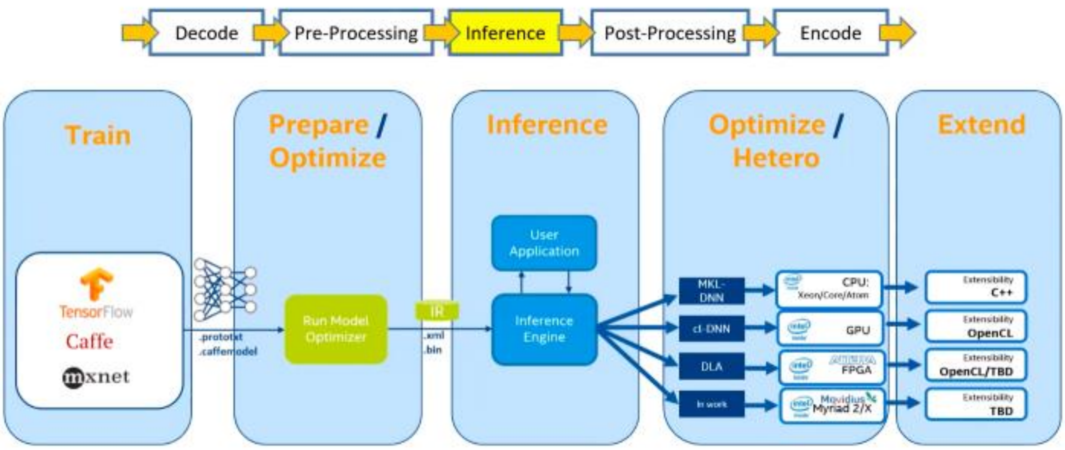


Fig. 5.4. Overview of OpenVINO

Additionally, the Message Queuing Telemetry Transport(MQTT) lightweight communication protocol is used, to implement messaging patterns, allowing applications to run independently over the network. MQTT protocol is integrated into the platform, for sending the image along with relevant data, like sensor location, topic, and timestamp in a JSON message. Specified TCP ports are used for the transportation of messages, on TCP/IP stack, where publisher is the client that sends data and the subscriber is the one that receives them, while broker is the application server which mostly controls the clients and their connectivity, messages delivery and storage, and the authentication. The surveillance camera is the publisher, while the inference engine is integrated into the publishers so based on the positive prediction of fire, only then a message is transmitted over the network. The server is the broker which sends the message to the web application, which is the subscriber. In case of both, negative or positive prediction of fire, the information is saved into the database, and only in the event of positive prediction it sends an email through the email alert system[28].

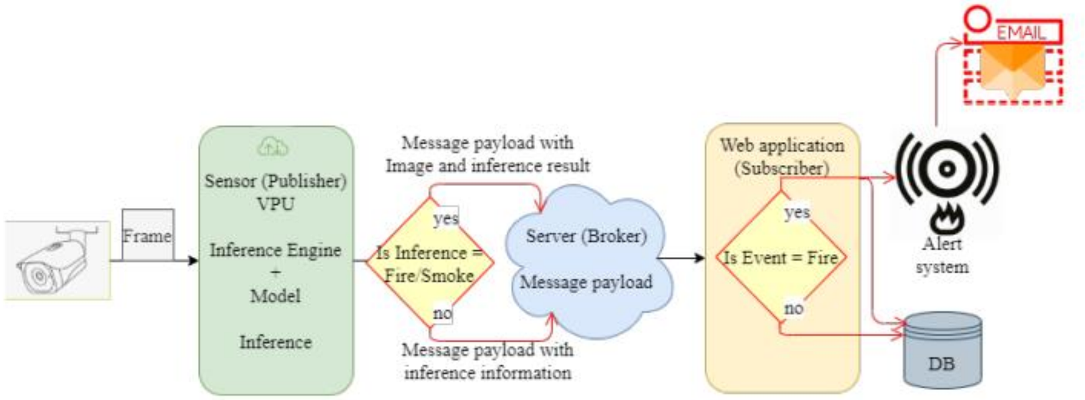


Fig. 5.5. *Conceptual representation of on-device Edge platform*

5.3 Deepstream pipeline

Another innovative tool against the problem of forest fires is the Deepstream pipeline and YOLOv3 along with edge computing approaches. This system based on multi sensors, that integrates edge computing with complex deep learning technologies, is designed to operate on specific NVIDIA Jetson edge devices, like Jetson NX Xavier. More specifically, Deepstream pipeline is using GStreamer framework and also uses CUDA GPU on the edge devices, allowing real time image and video analytics applications to be developed with maximum computation capabilities, while the user environment is using as operating system the Ubuntu 18.04 version. Deepstream SDK is an optimized toolkit that allows end to end AI applications suitable for the fire detection process. Overall, the use of Deepstream along with edge computing is a promising mechanism in the fire detection approach, since the high amount of data for the deep learning algorithms that need to be transmitted over the network are greatly reduced, which is significantly lowering the computational needs. However, there are limitations and challenges on that mechanism, as it still needs further optimization and validation in real time environments[29].

5.4 Fire alarm system for smart cities

This approach is designed and developed to provide a fire alarm system to a smart city, using the technologies of IoT and Edge computing. This system consists of a single central node, which is implemented with a Raspberry Pi and a SIM7600X 4G hat communication module added to it, and the smart context-aware edge nodes, a network of various sensing nodes. Each of the censoring nodes is connected to a NodeMCU-ESP8266 microcontroller and has a unique 32-bit chip ID. The microcontroller is linked to many sensors, mostly digital and few analog, of different type, such as flame, smoke, methane, temperature, humidity and carbon monoxide(CO), whose functionality is to sense the environment and do the fire detection. Each sensing node is aware of the network topology and the values of the parameters of the sensors nearby, and when it notices an anomaly to the value of a parameter in comparison to a predefined threshold, it triggers a message to a bridge node which then forwards the information to the central node via the established communication channel[31].

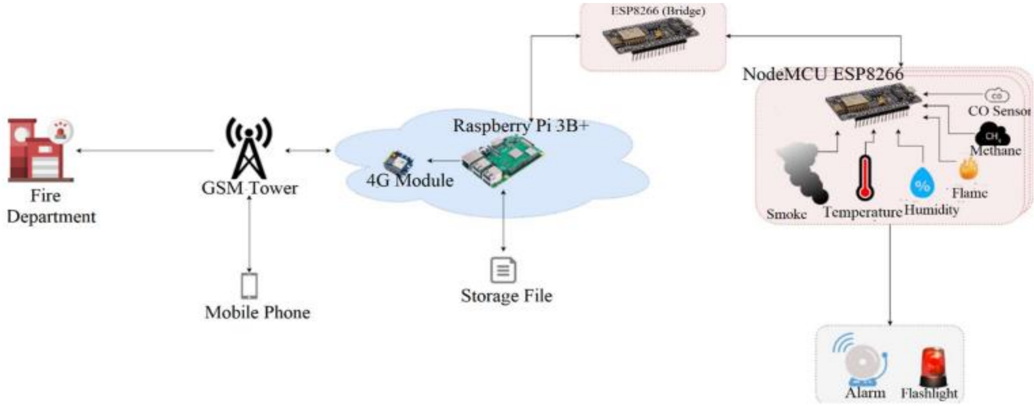


Fig. 5.6. *High level Architecture of Fire alarm system*

The sensing nodes form a mesh network, build by 'PainlessMesh' library, that has its own independent local Wi-Fi network and is connected to the bridge node acting as a gateway to the external network. The communication between the bridge node and the central node is done via the MQTT communication protocol, where the bridge node mostly operates as the publisher that sends the topics and the central node operates as the broker that receives those that is registered. The central node basically acts as the coordinator between the sensors, and the users or authorities that need to be informed of the fire alert, checks for failures in the maintenance, sends SMS, make calls, and has a webpage with the UI for users to communicate locally with the system. It will send SMS and call the phones of the users either when there is a fire alert triggered or when there is a request from a user, since data from the sensors are maintained and updated in the central node every 15 seconds. Requests from users can be done via SMS or the UI of the webpage hosted on a server on the central node. Additionally, it will send a SMS to the fire department with the location and the relevant data[31].

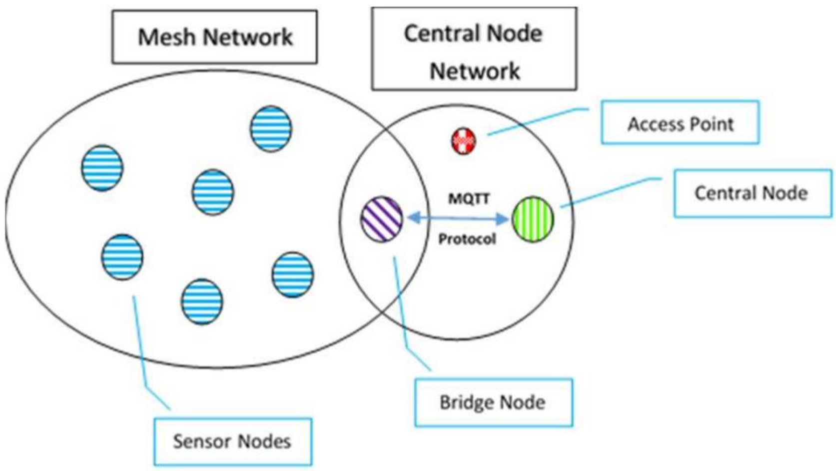


Fig. 5.7. *Network overview of proposed Fire alarm system*

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

Fire and smoke detection is vital for the insurance of the safety of the people and the ecosystem, and the reduction of the collateral damages to the infrastructures. Overall, this thesis is a contribution to the field of fire and smoke detection systems, including different types of detection methods and technologies, their advantages and disadvantages, challenges and opportunities related to their implementation. The use of case studies and real-world examples also adds depth and relevance to the analysis, illustrating how various technologies applied in practice to improve or create more sophisticated fire and smoke detection systems, especially machine learning and convolutional neural networks. Furthermore, due to the increasing demand for data processing, and to the fact that a central processing unit results in a single point of failure, it provides a comprehensive and insightful analysis of the potential of edge computing in fire and smoke detection systems and the technical specifications of the deployment of computing resources and data storage closer to the data generating devices.

6.2 Future Work

This thesis not only highlights the technical specifications, the implications and limitations, the recommendations and potentials for future development, and the importance of embracing emerging technologies, such as of edge computing and other related technologies, in enhancing public safety, but also identifies the need for a holistic approach that integrates technological innovation with organizational and social factors. More specifically, future research can explore the integration of edge computing technologies to devices equipped with sensors like robots, drones, and IoT devices, and to AI based algorithms, in order to create more robust and scalable architectures and to achieve detection in highly challenging environments. Moreover, a successful implementation requires a careful consideration of issues like data privacy, regulatory compliance, training programs for effective application management, and cybersecurity.

REFERENCES

[1] Shirley Selvan, David Anthony Durand, V. Gowtham(7 November, 2019). Survey of the Various Techniques Used for Smoke Detection Using Image Processing.
https://link.springer.com/chapter/10.1007/978-3-030-32150-5_76

[2] John Adedapo Ojo, Jamiu Alabi Oladosu(January, 2014). Video-based Smoke Detection Algorithms: A Chronological Survey.
https://core.ac.uk/download/pdf/234644843.pdf

[3] Yakhyokhuja Valikhujaev, Akmalbek Abdusalomov, Young Im Cho(18 November, 2020). Automatic Fire and Smoke Detection Method for Surveillance Systems Based on Dilated CNNs.
https://www.mdpi.com/2073-4433/11/11/1241/pdf

[4] Abdulaziz Namozov, Young Im Cho(November, 2018). An Efficient Deep Learning Algorithm for Fire and Smoke Detection with Limited Data.
https://www.researchgate.net/publication/329411277_An_Efficient_Deep_Learning_Algorithm_for_Fire_and_Smoke_Detection_with_Limited_Data

[5] Pu Li, Wangda Zhao(June, 2020). Image fire detection algorithms based on convolutional neural networks.
https://www.sciencedirect.com/science/article/pii/S2214157X2030085X

[6] Sergio Saponara, Abdussalam Elhanashi, Alessio Gagliardi(10 November, 2020). Real-time video fire/smoke detection based on CNN in antifire surveillance systems.
https://link.springer.com/content/pdf/10.1007/s11554-020-01044-0.pdf

[7] Qingjie Zhang, Jiaolong Xu, Liang Xu and Haifeng Guo(January, 2016). Deep Convolutional Neural Networks for Forest Fire Detection.
https://www.researchgate.net/publication/315562504_Deep_Convolutional_Neural_Networks_for_Forest_Fire_Detection#:~:text=Sharma%20et%20al.%20developed%20a%20fire%20detection%20scheme,region-based%20CNN%20to%20detect%20wild%20forest%20fire%20smoke

[8] Steven Verstockt, Peter Lambert, Rik Van de Walle, Bart Merci, Bart Merci(September , 2009). State of the art in vision-based fire and smoke detection.
https://www.semanticscholar.org/paper/State-of-the-art-in-vision-based-fire-and-smoke-Verstockt-Lambert/473966ec93310c6ffbabfbe43bc6de886c81583a

[9] Malik Mohamed Umar, Liyanage C De Silva, Muhammad Saifullah Abu Bakar(January, 2017). State of the art of smoke and fire detection using image processing.
https://www.researchgate.net/publication/317610608_State_of_the_art_of_smoke_and_fire_detection_using_image_processing

[10] Dae-Hyun Lee, Sang Hwa Lee, Taeuk Byun, Nam Ik Cho(30 August, 2017). Fire Detection using Color and Motion Models.
https://www.koreascience.or.kr/article/JAKO201725864506220.pdf

[11] Tony Lindeberg(2012). Scale Invariant Feature Transform, Scholarpedia.
http://www.scholarpedia.org/article/Scale_Invariant_Feature_Transform

[12] Bo Jiang, Yongyi Lu, Xiying Li, Liang Lin(2 February, 2015). Towards a Solid Solution of Real-time Fire and Flame Detection.

https://arxiv.org/pdf/1502.00416.pdf

[13] Ahmad A. A. Alkhatib(March, 2014). A Review on Forest Fire Detection Techniques https://journals.sagepub.com/doi/epub/10.1155/2014/597368

[14] Blesson Varghese, Nan Wang, Sakil Barbhuiya, Peter Kilpatrick, Dimitrios S. Nikolopoulos(26 December, 2016). Challenges and Opportunities in Edge Computing https://www.researchgate.net/publication/307888359_Challenges_and_Opportunities_in_Edge_Computing

[15] Keyan Cao, Yefan Liu, Gongjie Meng, Qimeng SunAn(01 May, 2020). Overview on Edge Computing Research
https://ieeexplore.ieee.org/abstract/document/9083958

[16] Mahesh K. Singha, M. Karthikb, P. Rameshc, G. Rama Naidud(January, 2023). Deep Neural Network Inference via Edge Computing: On-Demand Accelerating
https://www.researchgate.net/publication/367067702_Deep_Neural_Network_Inference_via_Edge_Computing_On-Demand_Accelerating

[17] D. Divya, T. R. Ganesh Babu(7 November, 2019). A Survey on Image Segmentation Techniques
https://link.springer.com/chapter/10.1007/978-3-030-32150-5_76

[18] Cheol Ho Hong, Blesson Varghese(September, 2019). Resource Management in Fog/Edge Computing: A Survey on Architectures, Infrastructure, and Algorithms
https://dl.acm.org/doi/pdf/10.1145/3326066

[19] Wonjae Lee, Seonghyun Kim, Yong-Tae Lee, Hyun-Woo Lee, Min Choi(30 March, 2017). Deep Neural Networks for Wildfire Detection with Unmanned Aerial Vehicle
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7889305

[20] Yanmin Luo, Liang Zhao, Peizhong Liu, Detian Huang(8 August, 2017). Fire smoke detection algorithm based on motion characteristic and convolutional neural networks
https://sci-hub.ru/10.1007/s11042-017-5090-2

[21] Shawn P. Urbanski, Wei Min Hao, Stephen Baker(23 September, 2008). Developments in Environmental Science - Chemical Composition of Wildland Fire Emissions
https://www.sciencedirect.com/science/article/abs/pii/S1474817708000041

[22] Chi Yuan, Zhixiang Liu, Youmin Zhang(28 March, 2018). Learning-Based Smoke Detection for Unmanned Aerial Vehicles Applied to Forest Fire Surveillance
https://www.researchgate.net/publication/324067014_Learning-Based_Smoke_Detection_for_Unmanned_Aerial_Vehicles_Applied_to_Forest_Fire_Surveillance

[23] Rui Dong, Changyang She, Wibowo Hardjawana, Yonghui Li, Branka Vucetic(10 October 2019). Deep Learning for Hybrid 5G Services in Mobile Edge Computing Systems: Learn from a Digital Twin
https://www.researchgate.net/publication/324067014_Learning-Based_Smoke_Detection_for_Unmanned_Aerial_Vehicles_Applied_to_Forest_Fire_Surveillance

[24] Tiago Koketsu Rodrigues, Katsuya Suto, Hiroki Nishiyama, Jiajia Liu, Nei Kato(24 September, 2019). Machine Learning meets Computation and Communication Control in Evolving Edge and Cloud: Challenges and Future Perspective

https://ieeexplore.ieee.org/abstract/document/8847416

[25] Garrett McGrath, Paul R. Brenner(17 July, 2017). Serverless Computing: Design, Implementation, and Performance
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7979855

[26] Sukhpal Singh Gill(18 January, 2021). Quantum and blockchain based Serverless edge computing: A vision, model, new trends and future directions
https://onlinelibrary.wiley.com/doi/epdf/10.1002/itl2.275?saml_referrer

[27] Shriya Kaur Chawla, Vinayak Malhotra (20 March, 2021). Jet fans as Stratified Tunnel Fire Suppressants
https://iopscience.iop.org/article/10.1088/1742-6596/2054/1/012001/pdf

[28] Sandeep Reddy Sabbella(March, 2021). Fire and Smoke Detection for Smart Cities Using Deep Neural Networks and Edge Computing on Embedded Sensors
https://www.researchgate.net/publication/353345042_Fire_and_Smoke_Detection_for_Smart_Cities_Using_Deep_Neural_Networks_and_Edge_Computing_on_Embedded_Sensors#fullTextFileContent

[29] Yi-Chun Chen, Halim Fathoni, Chao-Tung Yang(30 December, 2020). Implementation of Fire and Smoke Detection using DeepStream and Edge Computing Approaches
https://ieeexplore.ieee.org/document/9302759

[30] Marios Avgeris, Dimitrios Spatharakis, Dimitrios Dechouniotis, Nikos Kalatzis, Ioanna Roussaki, Symeon Papavassiliou(3 February, 2019). Where There Is Fire There Is SMOKE: A Scalable Edge Computing Framework for Early Fire Detection
https://www.mdpi.com/1424-8220/19/3/639/htm

[31] Asma Mahgoub, Nourhan Tarrad, Rana Elsherif; Loay Ismail, Abdulla Al-Ali(11 May, 2020). Fire Alarm System for Smart Cities Using Edge Computing
https://ieeexplore.ieee.org/document/9089653