Thesis submitted to the

UNIVERSITY OF MOHAMED BOUDIAF – MSILA

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE

In partial fulfillment of the requirements for the degree of

Master in **Computer science**

By

AMROUNE Karima

RAMDANI Manel

Entitled

Wildfire Detection Using Computer Vision

Composition of the jury

Hichem Debbi	University of Msila	Supervisor
Said Amri	University of Msila	President
Raouf LakehalAyat	University of Msila	Examiner

june, 2024

Dedications

باسم الله خالقي ووَليّي وميسّر أموري، الذي ما تفوّقتُ وما علَوتُ الا بإذنه، له كل الفضل والمنّة
الحمد لله على ما علّمني وهداني اِليهِ وسخّر

أهدي هذا النجاح لنفسي أوّلاً بعد مسيرة دامت سنوات من الجهد والتعب
والى كل من ساهم فيه وشجّعني عليه

اِلى من كلّله ربّي بالهيبة والوقار، الى الذي تعِب كي أرتاح وحُرِم كي أَنعَم
الى من مهّد لي طريق العلم وساندني عليه
-أبي حبيبي-

اِلى ملاكي في الدنيا، معنى الأمان والحب، الى رفيقتي في كل الخطوات
بسمة حياتي وداعمي الأول، الى من كان دعاؤها سرًّا لِنجاحي وحنانها دواءً لارهاقي
-أمي الرّوح-

الى من علّمني كتابة اسمي وأوّل الحروف، اللّاتي أجلَسنَني على مكاتِبهنّ أعبث بأور اقهنّ وكتبهنّ فسبِقت جيلي بفضلِهن
الى سندي في الحياة والكتف الذي أميل عليه،
ها أنا اليوم أختم المسيرة التي بدأتموها وأسير على نفس خطاكن
-أخو اتي المؤنسات-

الى من خفّفن عني مشقة الدنيا، من تقاسمت معهن أكواب القهوة قبل كل حصة، سبب ضحكاتي وتجاوزاتي للصعاب
الرفيقات اللاتي كلما توقّفت دفعوني، وكلّما نسِيت ذكّروني، من لهنّ أثر النجاح في حياتي

فلله الحمد على ما قسّم و أنعم، والحمد لله على ما لطف و أكرم

.

﴿ وَآخِرُ دَعْوَاهُمْ أَنِ الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ ﴾

-رمضاني منال-

من قال انا لها " نالها ".

لم تكن الرحلة قصيرة ولا ينبغي لها أن تكون، لم يكن الحلم قريبا ولا الطريق كان محفوفا بالتسهيلات،

لكني فعلتها ونلتها.

الحمدالله حبا وشكرا وامتنانا الذي بفضله ها أنا اليوم أنظر إلى حلم طال انتظاره وقد أصبح و اقعا أفتخر به.

إلى ملاكي الطاهر، وقوتي بعد الله، داعمتي الأولى والأبدية " أمي " أهديك هذا الإنجاز الذي لولا تضحياتك لما كان له وجود، ممتنة لأن الله قد اصطفاك لي من البشر أما يا خير سند وعوض.

إلى من كلل العرق جبينه إلى من دعمني بلا حدود وأعطاني بلا مقابل الذي لم يبخل على طيلة حياته إلى النور الذي أنار دربي والسراج الذي لا ينطفئ نوره بقلبي أبدا "والدي العزيز".

إلى من قيل فيهم: "سنشد عضدك بأخيك" إلى من مدو يدهم دون كلل وملل وقت ضعفي "إخوتي" أدامكم الله ضلعا ثابتا لي.

إلى من سندني وصبر معي هاتين السنتين: زوجي أهديك هذا البحث تعبيرا عن خالص شكري، لما قدمته لي طوال فترة دراستي من دعم معنوي ومادي فكنت نعم الزوج والصديق.

-عمرون كريمة –

Acknowledgments

We would like to express our profound appreciation to our supervisor **Pr. Debbi Hichem**, whose constant assistance and direction were invaluable on our trip. Your kindness and commitment to following us step by step along the way have been priceless to our academic and personal development.

We also wish to express our heartfelt appreciation to all our professors, whose knowledge, expertise, and passion have enriched our learning experiences over the years.

To our friends, whose encouragement, camaraderie, and laughter have brightened even the toughest of days, we offer our deepest thanks.

To everyone who has played a part in our education and development, we offer our heartfelt appreciation. Thank you for shaping us into the individuals we are today.

Contents

Dedications **3**
Acknowledgments **5**
Contents **6**
List of Figures **10**
List of Tables **12**
General Introduction **13**
Chapter 1 Wildfire Detection **15**
1 Introduction: **15**
2 Wildfire Causes: **15**
3 Wildfire types: **16**
3.1 Controlled Burns : 16
3.2 Surface Fires : 16
3.3 Crown fires: 16
3.4 Ground Fires : 16
3.5 Firestorms : 16
3.6 Urban Interface Fires : 17
3.7 Bushfires : 17
3.8 Spot Fires : 17
3.9 Peat Fires : 17
3.10 Rekindles : 17
4 Wildfire Consequences: **17**
4.1 Environmental Impact : 18
4.2 Economic Consequences: 18
4.3 Public Health Impact : 18
4.4 Social and Community Impacts : 19
4.5 Legal and Regulatory Implications : 19
5 WildFire Solutions: **19**
5.1 Prevention Measures : 20
5.2 Preparedness Measures : 20
5.3 Response Measures : 20
5.4 Mitigation of Climate Change: 21
5.5 Collaboration and Research : 21
6 The importance of early detection in wildfire: **21**
7 Advances in Wildfire Detection Technology: **22**
7.1 Satellite Technology: 22

7.2 Remote Sensing: 22
7.3 IoT and Sensor Networks: 23
7.4 Machine Learning and AI : 23
7.5 Smart Infrastructure: 23
8 The Urgency of Real-Time Detection: **23**
9 Challenges and Future Direction of Real Time Wildfire Detection: **24**
10 Conclusion: **25**
Chapter 2 Computer Vision **26**
1 Introduction: **26**
2 Computer vision : **26**
2.1 Definition : 26
2.2 Computer Vision Applications: 27
2.3 Computer vision algorithms: 28
2.3.1 Image classification: 28
2.3.2 Image segmentation: 28
2.3.3 Object detection: 28
2.3.4 facial recognition: 29
3 Machine Learning : 29
3.1 Definition : 29
3.2 Types of Machine Learning: 30
3.2.1 Supervised learning: 30
3.2.2 Unsupervised learning: 30
3.2.3 Semi-Supervised learning : 30
3.2.4 Reinforcement Learning: 30
3.3 Motivation to use ML: 31
4 Deep Learning : **31**
4.2 Definition : 31
4.2 How does Deep Learning work? 31
4.3 Deep learning models: 32
4.4 Motivation to use DL: 33
4.5 Deep Learning Requirements: 33
4.6 Machine learning vs Deep learning: 34
5 Convolutional neural network: **35**
5.1 Definition: 35
5.2 Convolutional neural network Architecture: 35
5.2.1 Convolutional layer: 35
5.2.2 Pooling layers: 36

5.2.3 Fully connected layers: 36
5.2.4 Dropout Layer: 36
5.2.5 Activation Function: 36
6 Transfer Learning: 36
6.1 Definition: 36
6.2 Transfer Learning Types: 37
6.3 Why using Transfer Learning? 38
6.4 Transfer Learning Applications: 39
6.5 Pre-trained Models: 39
7 Infrared Imaging: 41
7.1 Principle: 42
7.2 Applications: 42
7.3 Benefits: 42
7.4 Infrared Imaging Types: 42
7.5 Image Processing: 42
Chapter 3 Related Works 43
1. Introduction: 43
2 The most significant techniques previously employed: 43
2.1 Wildfire classification : 43
2.1.1 Binary classification: 43
2.1.2 Multiclass classification: 47
2.2 Wildfire Detection: 48
2.3 Wildfire Segmentation: 50
3 The most widely used databases on this topic: 52
4 General Comparaison : 56
5 Conclusion: 57
Chapter 4 Framework Implementation and Experimental Results 58
1 Introduction: 58
3 Our Proposed Models: 60
3.1 First framework (A Fine-tuned VGG16 model on the Wildfire Dataset) [23] : 60
3.2 Second framework (A Fine-tuned late fusion model on Flame 2) [24] : 64
4 Comparaison between our approaches and the original ones : 68
5 Why vgg16 ? 69
6 Why using drones for collecting dataset ? 70
7 The Used Programing Languages, Libraries and Tools: 71
7.1 Programming Languges : 71

7.2 Libraries: 71
7.3 Tools and IDEs: 72
- **Kaggle IDE:** 72
- **VS Code:** 73
- **Flask Framework:** 63
- **Figma Framework:** 63

8 Web and Mobile Applications Prototypes: 75
8.1 Web Application: 76
8.2 Mobile Application: 78
9 Conclusion: 79
General Conclusion 80
References 81
Abstruct 84

List of Figures

Figure 2. 1:Human Vision VS Computer Vision 27
Figure 2. 2: Types of Machine Learning 31
Figure 2. 3: Deep learning architecture 32
Figure 2. 4: CNN's structure 35
Figure 2. 5: Transfert Learning 37
Figure 2. 6: VGG-Model Architecture 40

Figure 3. 1: General Architecture of CNN Based Models 46
Figure 3. 2: Hyperparameter configuration for the used Method 48
Figure 3. 4: Resized sample of annotated images for MLC: (a) downloaded from the Web; (b) KTS dataset; (c) DNIM dataset; (d) dataset for error protection purposes; (e) dataset of a controlled fire captured by researchers. 53
Figure 3. 5: Images from the introduced dataset depicting visual features that may be misinterpreted by the model as a forest fire scene. 54
Figure 3. 6: Handpicked frame pairs from FLAME 2 Dataset [17]. (a) No flame with No smoke, (b) flame with no smoke, (c)(d) flame with smoke. 55
Figure 3. 7: examples from this data set. In the fire masks, red corresponds to fire, while grey corresponds to no fire. Black indicates uncertain labels (i.e., cloud coverage or other unprocessed data) 56

Figure 4. 1: Importing Necessary Libraries 60
Figure 4. 2: Preparing datasets to be processed 61
Figure 4. 3: Importing VGG16 without its top layers 61
Figure 4. 4: Including the additional layers for the first model 62
Figure 4. 5: Compiling and fitting the first model 62
Figure 4. 6: Accuracy and Loss plots of our model for “the wildfire dataset “ 63
Figure 4. 7: Early stopping function 63
Figure 4. 8: The prediction function 64
Figure 4. 9: Creating base Model 65
Figure 4. 10: Parallel Processing of Modalities 66
Figure 4. 11: Custom Data Generator 67
Figure 4. 12: Model Training and Evaluation 67
Figure 4. 13: loss and accuracy plots for fusion model 68
Figure 4. 14: Python 71
Figure 4. 15: Tensorflow 71

Figure 4. 16: Visual Studio Code 73
Figure 4. 17: Flask Logo 73
Figure 4. 18: Figma Logo 73
Figure 4. 19: Prototypes structure diagram 75
Figure 4. 20: Login page 76
Figure 4. 21: Dashboard Page for Forest Governorate 76
Figure 4. 22: Dashboard for Firefighters Unit 77
Figure 4. 23: Login Process Page 78
Figure 4. 24: Dashboard of Mobile app 78

List of Tables

Table 2. 1: Machine Learning Vs Deep Learning 34

Table 3. 1: Deep learning methods for UAV-based fire binary classification. 44
Table 3. 2: Performance evaluation of wildfire classification models on the FLAME dataset. 45
Table 3. 3: Hyperparameters used in the experiments 46
Table 3. 4: Experimental results obtained from the MALE 46
Table 3. 5: Performance metrics using augmented and 48
Table 3. 6: Fire detection using Deep learning methods for UAVs. 49
Table 3. 7: Fire segmentation using deep learning methods for UAVs. 51
Table 3. 8: Performance evaluation of wildfire segmentation models 51
Table 3. 9: General comparison between the most effective models 57

Table 4. 1: Data distribution across dataset's subclasses. 59
Table 4. 2: FLAME-2 dataset contents 60
Table 4. 3: Final Approaches Comparaison 68
Table 4. 4: Brief Summary about VGG16 69

General Introduction

In the context of facing the increase in wildfires and their widespread impact on ecosystems, human safety and infrastructure, the necessity of quickly detecting and mitigate their impact has become crucial. Wildfires have become increasingly frequent and severe in recent years, fueled by factors such as climate change, land use practices, and human activity. These catastrophic events can result in the loss of biodiversity, destruction of habitats, displacement of wildlife, and irreversible damage to natural landscapes. Moreover, wildfires pose significant risks to human lives, homes, and livelihoods, as they can rapidly spread and absorb large areas in flames, leading to evacuations, property damage, and even loss of life. Therefore, timely detection of wildfires is not only essential for initiating prompt response efforts but also represents a critical step in minimizing the destruction they can cause. Early detection enables emergency responders to mobilize resources quickly, deploy firefighting teams, and implement evacuation plans, thereby reducing the spread of wildfires and mitigating their impact on communities. Additionally, early warning systems can alert residents to potential dangers, allowing them to take proactive measures to protect themselves, their families, and their property.

In recent years, the integration of computer vision technology has emerged as a transformative approach to augment wildfire detection capabilities, offering innovative solutions to address the growing challenges posed by wildfire incidents. By harnessing advanced image processing algorithms and artificial intelligence techniques, computer vision systems can analyze visual data streams from various sources, such as satellite imagery, aerial surveillance, and ground-based cameras, to quickly identify potential wildfire occurrences. These systems can detect telltale signs of wildfires, such as smoke plumes, flames, and changes in vegetation patterns, with remarkable accuracy and speed. Moreover, computer vision algorithms can continuously monitor vast expanses of land in real-time, enabling early detection of wildfires before they escalate into catastrophic events. By providing timely alerts to emergency responders and enabling proactive measures, computer vision technology holds tremendous potential to enhance wildfire management strategies and mitigate the devastating impact of wildfires on ecosystems and communities.

Objective:

Our objective is to create a new predictive model for real-time wildfire detection using transfer learning and fusion techniques. By leveraging pre-trained deep learning models and adapting them to the task of wildfire detection, we aim to enhance accuracy and efficiency. Our models will be trained on diverse datasets of wildfire images, enabling robust detection capabilities that can provide timely alerts to aid in the swift containment and mitigation of wildfires.

Manuscript Organisation :

- Chapter 1: In this chapter, we will take a look inside the complex world of wildfires, examining their causes, types, consequences, and the urgent need for effective solutions and early detection.
- Chapter 2: In this chapter, we explore the field of computer vision, investigating its fundamental principles, including machine learning and deep learning techniques, with a particular focus on transfer learning.
- Chapter 3: The third chapter initiates a comprehensive review of existing literature and research in the field, analyzing previous approaches, methodologies, and case studies to learn more about their strengths and limitations.
- Chapter 4: In the last chapter, we propose a novel methodology for wildfire detection using Convolutional Neural Networks (CNNs), detailing the experimental framework, data collection procedures, and the development of our detection models and prototypes.

Chapter 1 Wildfire Detection

1 Introduction:

Wildfires are powerful and unpredictable forces of nature that can transform landscapes in a matter of hours. This chapter looks at what causes wildfires, how they can be different, and the problems they create for nature and people.

We'll also look at how wildfires damage the environment, harm animals, and affect people's lives, and explore the critical need for effective solutions and early detection systems to manage and mitigate the devastating impacts of these fires, highlighting the urgency of protecting both our natural world and our lives from their destructive path.

2 Wildfire Causes:

Although there are many different reasons why wildfires occur, understanding these causes is crucial for prevention and mitigation efforts aimed at reducing the frequency and severity of wildfires. Some of the main ones are as follows:

- **Human Activities:** These include abandoned cigarettes, unsupervised campfires, sparks from machinery or cars, pyrotechnics, arson, and electrical lines.
- **Natural Causes:** Wildfires are frequently caused by lightning strikes, among other natural sources. Lightning-caused fires can also occur during dry thunderstorms, in which the rain evaporates before it hits the ground.
- **Wind:** Wildfires can spread quickly due to strong winds, which makes containment more challenging. Long-distance wind can carry embers, starting new fires and accelerating the growth and spread of already-existing ones.
- **Topography:** Wildfire behavior can be influenced by an area's topography and geography. Not only can canyons and valleys funnel winds and intensify fires, but steep slopes can also accelerate the spread of fires uphill.

- **Climate Change:** The risk of wildfires can be increased by climate change through changes in weather patterns, extended droughts, and higher temperatures. In many areas, these changes may result in wildfire conditions that are more intense and frequent.

3 Wildfire types:

Wildfires can be categorized based on various factors including their size, behavior, and the type of vegetation they primarily affect, each type of wildfire presents its own set of challenges and considerations for firefighting, prevention, and management. Here are some common types of wildfires:

3.1 Controlled Burns:

These deliberately started fires, sometimes referred to as controlled or prescribed burns, are managed by land management organizations under strict guidelines with the goal of minimizing fuel accumulation, lowering the danger of more destructive wildfires, and enhancing the health of the ecosystem. Controlled fires are meticulously organized and closely watched to make sure they stay under allotted limits and do not endanger property or public safety. [1]

3.2 Surface Fires :

These fires typically burn along the forest floor or grasslands, consuming dry vegetation such as fallen leaves, grass, and shrubs. Les incendies forestiers de surface sont le type le plus courant et se propagent généralement assez lentement. [2]

3.3 Crown fires:

When flames travel across a tree's canopy, a crown fire happens. Due to the quantity of dry leaf in the tree crowns and high winds, these fires can be quite severe and swift. [2]

3.4 Ground Fires :

Peat and other organic-rich soil are examples of the organic material that is burned in ground fires. They can linger for a long time and are frequently challenging to find and put out. [2]

3.5 Firestorms:

Firestorms are extreme fire behavior phenomena characterized by intense heat, strong winds, and towering pyrocumulus clouds. These events can generate their own weather patterns, including lightning and powerful gusts, which can exacerbate fire spread and make firefighting efforts extremely challenging.

3.6 Urban Interface Fires :

These fires happen at the intersection of urban development and wilderness areas, resulting in a mixture of natural plants and built structures. Because of the close proximity of flammable materials and the possibility for rapid fire spread into populated areas, urban interface fires provide special concerns.

3.7 Bushfires :

Scrublands, woodlands, and other places with a lot of vegetation are common places for bushfires, often called scrub fires. These fires are common in areas where hot, dry weather is likely to occur and can spread quickly.

3.8 Spot Fires:

Small fires known as "spot fires" are started by sparks or embers that are transported by the wind ahead of the main fire front. They may transpire some distance from the primary fire and aid in its spread.

3.9 Peat Fires:

Where there are peatlands that are rich in organic matter, peat fires can develop. These subsurface flames have the capacity to burn for extended periods of time, producing copious volumes of smoke and greenhouse gases.

3.10 Rekindles :

Rekindles are fires that have already been put out; they are frequently brought on by embers that are still smoldering or by hotspots that have not completely burned out. Rekindles can happen days or even weeks after the original fire and can happen again when new fuel sources become available, or the weather is right.

4 Wildfire Consequences:

Comprehending the entire range of outcomes linked to wildfires is crucial in order to formulate comprehensive strategies for managing wildfires, enhancing resilience, and implementing sustainable land use practices. The well-being of human and environmental systems is promoted by tackling the complex issues posed by wildfires through stakeholder collaboration, interdisciplinary research, and community participation.

The following are some of the main effects:

4.1 Environmental Impact:

Significant environmental effects of wildfires include soil erosion, biodiversity loss, habitat damage, and air pollution. Natural environments including wetlands, woodlands, grasslands, and wildlife refuges can be completely destroyed by these fires, uprooting or eliminating local plant and animal populations. Long-term effects on biodiversity may result from the ensuing loss of vegetation and habitat diversity, which could jeopardize species populations and weaken ecosystem resilience. In addition, the removal of vegetation from streams by wildfires increases sedimentation and lowers the quality of the water, which exacerbates soil erosion. Moreover, smoke, particulate matter, and toxins released into the atmosphere by wildfires lead to poor air quality and respiratory health problems in both humans and wildlife. These consequences on the environment emphasize how urgently effective wildfire management measures are needed to lessen the destructive effects of these.

4.2 Economic Consequences:

Wildfires pose significant economic challenges, leading to property damage, disruptions in economic activities, and substantial firefighting costs. These fires can destroy homes, businesses, infrastructure, and agricultural assets, resulting in significant property damage and economic losses for affected communities. Furthermore, wildfires can disrupt various economic activities, including tourism, agriculture, forestry, and outdoor recreation, through road closures, evacuations, and damage to utilities. Additionally, the costs associated with wildfire suppression efforts, including personnel, equipment, aircraft, and logistical support, can be substantial, straining government budgets and resources. These economic impacts highlight the importance of proactive measures to prevent and mitigate the devastating effects of wildfires on local economies and communities.

4.3 Public Health Impact:

Wildfires have serious effects on public health, including issues with water contamination, mental health effects, and respiratory health. In addition to raising the risk of respiratory infections in those who are exposed, smoke from wildfires can worsen respiratory ailments including bronchitis and asthma. Wildfires can also result in psychological anguish, worry, and trauma for those who are immediately impacted by them as well as for the communities that are at risk of becoming threatened by them and having to deal with the devastation they leave behind. Moreover, ash, debris, and chemical pollutants from wildfires can contaminate water sources, endangering aquatic ecosystems and supplies of drinking water. These effects on public health highlight how crucial it is to use efficient wildfire management techniques

together with preventative steps to safeguard people's health and safety both during and after wildfire incidents.

4.4 Social and Community Impacts:

Significant social and cultural effects of wildfires can include trauma and displacement, disturbance of communities, and loss of cultural assets. People are frequently forced to from their homes and communities due to these fires, which can result in either temporary or permanent displacement and mental anguish for those impacted. Moreover, wildfires undermine social networks, local institutions, and community cohesion, making it difficult for impacted communities to recover socially and economically. Furthermore, wildfires have the potential to demolish traditional lands, artifacts, and cultural heritage sites that are significant to indigenous populations, causing them to lose their sense of cultural identity and connection to the land. These effects on society and culture emphasize how crucial it is to assist impacted people, protect cultural heritage, and build resilience against wildfire occurrences.

4.5 Legal and Regulatory Implications:

Wildfires have the potential to prompt regulatory and judicial actions. Individuals and organizations may be liable for damages caused by wildfires in terms of liability and legal procedures, particularly when negligence, arson, or disregard for fire safety standards are involved. Legal proceedings, insurance claims, and litigation resulting from wildfires can subject accountable parties to severe financial hardships and legal ramifications. In addition, policies and regulations aimed at enhancing fire prevention, mitigation, and disaster management are frequently prompted by wildfires. Land use planning, building codes, fire risk assessments, and wildfire control techniques may all need to be modified in response to these developments. Reducing potential hazards, building community resilience, and encouraging better wildfire management strategies are the objectives. These legislative and regulatory actions are essential for reducing the effects of wildfires and protecting property and life.

5 WildFire Solutions:

Addressing wildfires requires a multi-faceted approach involving prevention, preparedness, and response. Implementing a combination of these strategies allows communities to work towards reducing the frequency and severity of wildfires, protecting lives and property, and preserving natural ecosystems. The following are some solutions:

5.1 Prevention Measures :

- Forest management techniques like controlled burns, thinning forests, and clearing debris help to reduce fuel loads and prevent wildfires by eliminating excess vegetation and combustible materials. These practices promote healthier ecosystems and protect communities from the destructive impact of wildfires.
- The prevention of wildfires caused by humans can only be achieved by public education about fire safety, responsible land management, and fire prevention. Communities can reduce the chance of unintentional fires and save lives and property by spreading information and encouraging safe conduct.
- Enforcing building rules that mandate the use of fire-resistant materials and keeping defensible areas around buildings can help prevent wildfires from damaging homes and infrastructure by lowering the danger of ignition and erecting barriers that prevent the spread of fire.
- Effective wildfire risk mitigation can be achieved by closely enforcing fire restrictions during dry seasons and conducting routine monitoring of high-risk regions.

5.2 Preparedness Measures :

- Effective early warning systems, like sirens, mobile alerts, and community notifications, ensure residents receive timely information to evacuate safely during wildfires, minimizing risk of harm.
- By implementing standardized protocols for evacuation, communication, and resource distribution, community-level emergency planning guarantees a well-coordinated and effective response to wildfires, improving public safety and reducing damage.
- Investing in infrastructure, such as roads, water sources, and fire breaks, can help combat wildfires and shield nearby people from their effects
- Getting communities involved in neighborhood clean-up days and fire drills is a good way to increase overall preparation for wildfires.

5.3 Response Measures :

- Advanced technologies such as drones, satellite imagery, and firefighting aircraft are invaluable tools that enhance firefighting efforts by providing real-time information to incident commanders. Drones can offer aerial views of the fire, helping firefighters assess the situation and plan their response more effectively. Satellite imagery provides detailed information about the fire's spread and behavior, aiding in strategic decision-making. Firefighting aircraft, equipped with water or fire retardant, can quickly suppress flames in

hard-to-reach areas. By utilizing these technologies, firefighting teams can improve their efficiency and effectiveness in combating wildfires, ultimately reducing the impact on communities and ecosystems.

- Mutual aid agreements between fire departments and agencies ensure coordinated responses and resource-sharing during large wildfires, maximizing efficiency and effectiveness in firefighting efforts.
- Restoration of ecosystems and reduction of long-term effects of wildfires depend on the implementation of post-fire recovery strategies such as reforestation and erosion management.

5.4 Mitigation of Climate Change:

The conditions that lead to more frequent and severe wildfires can be lessened by addressing the underlying causes of climate change through the reduction of greenhouse gas emissions and the promotion of renewable energy sources.

5.5 Collaboration and Research :

To create efficient wildfire management methods, communities, researchers, NGOs, and government agencies must work together. Evidence-based wildfire policies and plans can be influenced by ongoing research into the behavior of wildfires, forecasting methods, and the efficacy of various control solutions.

6 The importance of early detection in wildfire:

For a number of reasons, early wildfire identification is essential because it enables prompt and efficient action to lessen the effects of wildfires on ecosystems, communities, and the environment. The following are some major arguments in favor of early wildfire detection:

6.1 Rapid Response:

In the early phases of a wildfire, early identification allows fire departments to react swiftly. In order to stop the fire from spreading and become harder to control, quick action is required.

6.2 Reducing Fire Spread:

By identifying wildfires early on, containment measures can be put in place before the fire has an opportunity to burn into wider regions. By doing this, it may be possible to save priceless natural resources, wildlife habitats, and property.

6.3 Protecting Lives and Property:

In order to preserve the lives of locals and emergency personnel, early discovery enables prompt evacuation of at-risk locations. More time is also given to put safety precautions for infrastructure and property into place.

6.4 Minimizing Environmental Impact:

The overall environmental impact of wildfires can be decreased with prompt discovery and action. It supports biodiversity, prevents soil erosion, and preserves ecosystems.

6.5 Cost Reduction:

The expense of firefighting can be greatly decreased by early detection. As flames get bigger and hotter, it gets harder and costs more to put them out. In the long term, resources can be saved by responding quickly to smaller fires.

6.6 Preserving Air Quality:

Large areas' air quality is impacted by wildfires because they send smoke and other pollutants into the atmosphere. Early detection aids in controlling and disseminating air quality warnings to save local residents' health.

7 Advances in Wildfire Detection Technology:

Recent years have seen notable advancements in wildfire detection technologies, which have improved early detection, monitoring, and response capabilities by utilizing advances in a variety of domains. Some notable advances include:

7.1 Satellite Technology:

- **High-Resolution Imagery:** High-resolution cameras on satellites allow for detailed image capture, which aids in the remote detection and observation of wildfires. This makes it possible to see more fire-prone areas.
- **Thermal Imaging:** Thermal sensors aboard satellites enable the detection of heat signatures, which makes it possible to identify active fires even when there isn't any smoke to see.

7.2 Remote Sensing:

- **Unmanned Aerial Vehicles (UAVs):** Drones with cameras and sensors can be used for data collection and real-time monitoring in areas susceptible to wildfires. They can get to places where it is difficult for people to go.

- **LiDAR Technology:** Light Detection and Ranging (LiDAR) technology can scan the terrain in great detail in 3D, which is useful for determining vulnerable locations and forecasting fire behavior.

7.3 IoT and Sensor Networks:

Sensor networks and Internet of Things (IoT) devices are used to monitor environmental factors in areas susceptible to wildfires. These sensors provide early warning indicators of possible fire outbreaks by detecting changes in temperature, humidity, wind speed, and air quality.

7.4 Machine Learning and AI :

- **Automated Image Analysis:** AI systems are able to examine aerial and satellite imagery to find indications of wildfires, such as smoke plumes or changes in the vegetation that suggest fire activity.
- **Predictive Modeling:** By examining a number of variables, such as past fire data, weather trends, and land cover, machine learning algorithms are able to forecast the danger of wildfires.

7.5 Smart Infrastructure:

- **Smart Grids:** These systems are able to identify and react to power line malfunctions or faults that have the potential to start wildfires in urban-wildland interface zones.
- **Building Sensors:** The placement of sensors on buildings can aid in the monitoring and early detection of wildfires in the vicinity of cities.

8 The Urgency of Real-Time Detection:

In general, the desire to save people, property, and the environment from the devastation caused by natural disasters is what makes real-time wildfire detection so urgent.

There are various reasons why it is essential to identify wildfires in real time, include:

- **Rapid response:** When a fire is detected in real time, resources for combating it can be quickly deployed to contain and put out the flames before they spread and cause further damage.
- **Public safety:** Early wildfire identification lowers the risk to people's lives and property by enabling the timely issuance of evacuation orders.
- **Environmental impact:** The impact of wildfires on ecosystems, wildlife, and air quality can be disastrous. Because real-time detection allows for faster containment and response, it can help limit these effects.

- **Economic impact:** The destruction of property, infrastructure, and natural resources caused by wildfires can lead to large financial losses. The quicker response that real-time detection enables can help lower these losses.

9 Challenges and Future Direction of Real Time Wildfire Detection:

There are various obstacles to overcome in real-time wildfire detection, and doing so is essential to enhancing early detection and response systems. In addition, using advanced technologies and creative methods is where wildfire detection is headed in the future.

Addressing these challenges and utilizing cutting-edge technologies, real-time wildfire detection can be greatly enhanced, allowing for quicker and more efficient responses to lessen the impact of wildfires on populations and ecosystems.

Here are some challenges and potential future directions:

9.1 Challenges :

- **Data Quality and Availability:** limited availability of high-quality, real-time data in inaccessible or distant areas. Accurate detection may be impeded by incomplete or antiquated weather data and satellite imaging.
- **Sensor Integration:** combining different sensor technologies—drones, satellites, and ground-based sensors—to create a unified, effective system. Guaranteeing data sharing and interoperability between various sensor networks.
- **False Alarms:** reducing false positives and negatives to prevent missed detections or needless emergency reactions. Developing algorithms to differentiate between wildfire incidents and natural phenomena.
- **Scalability:** creating scalable solutions that can manage large regions with a variety of diverse. Ensuring the system can adapt to changes in the environment and respond to multiple concurrent incidents.
- **Communication Infrastructure:** building strong communication networks in remote locations to provide response teams with real-time data. Reducing dependency on traditional communication infrastructure that may be vulnerable during wildfires.

9.2 Future Directions :

- **Artificial Intelligence and Machine Learning:** implementing advanced machine learning algorithms to detect wildfires more quickly and accurately. Pattern recognition, fire behavior prediction, and resource allocation are all made possible by integrating AI with data analysis.

- **IoT and Sensor Networks:** increasing the amount of real-time data collecting done by using sensor networks and Internet of Things (IoT) devices. Integrating newly developed sensor technologies to improve detection performance.
- **Satellite Technology:** making use of satellite imagery with high quality and increasing the frequency of data updates. Exploring the possibility of continuous surveillance using small satellite constellations.
- **Drones and Unmanned Aerial Vehicles (UAVs):** use drones more frequently for data collecting and real-time aerial surveillance. Creating autonomous unmanned aerial vehicles (UAVs) for quick reaction and surveillance in difficult terrain.
- **Community Engagement:** utilizing community-based sensor networks and smartphone applications to engage nearby communities in reporting and monitoring. Enhancing public awareness and education about wildfire prevention and response.
- **Climate Modeling:** combining weather and climate models to forecast areas that are likely to catch fire. Enhancing the understanding of the impact of climate change on wildfire frequency and intensity.

10 Conclusion:

In conclusion, wildfires represent a significant and recurring threat to communities, ecosystems, and economies worldwide. The devastation they bring underscores the urgent need for comprehensive strategies that prioritize prevention, early detection, and swift response. By investing in advanced technologies, enhancing public awareness, implementing proactive land management practices, and fostering international cooperation, we can mitigate the risks posed by wildfires and build greater resilience to their impacts. It's crucial that policymakers, communities, and individuals alike recognize the gravity of the wildfire threat and work together to implement effective solutions, ensuring the safety and sustainability of our planet for generations to come.

Chapter 2 Computer Vision

1 Introduction:

In general, researchers and practitioners in Computer Vision have been able to construct reliable, accurate, and effective vision systems for a variety of applications, from autonomous driving and medical imaging to surveillance and augmented reality, thanks to different architectures such as Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs). Through the utilization of pre-trained models and their subsequent refinement on certain tasks, developers can expedite the creation of state-of-the-art Computer Vision solutions. When combined with transfer learning, CNNs can be largely adapted in very efficient way in many computer vision tasks such as: Image Classification, Object Detection, Semantic Segmentation, Feature Extraction and others. All these tasks will be discussed in this chapter.

2 Computer vision :

2.1 Definition :

Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs and take actions or make recommendations based on that information. If AI enables computers to think, computer vision enables them to see, observe and understand.

Human vision is superior to computer vision, despite the latter's attempts to function in the same way. By using their sense of sight, humans are able to see objects clearly, determine their distance from one another, detect motion, and determine whether a picture is distorted. We train the machine to perform these functions, but it has to do it in much less time with cameras, data and algorithms rather than retinas, optic nerves and a visual cortex. Therefore, a vision system, which is trained to recognize thousands of objects and analyze hundreds of pixels in real-time, would be able quickly to surpass human capabilities. [3]

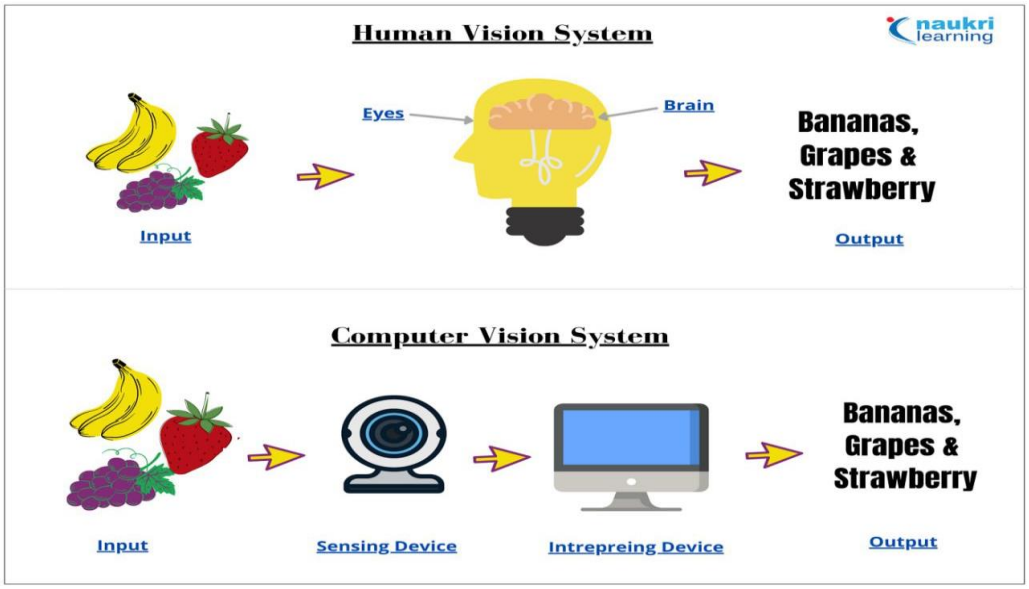

Figure 2. 1:Human Vision VS Computer Vision

2.2 Computer Vision Applications:

Computer vision is one of the most advanced inventions of artificial intelligence and machine learning. Real-world applications demonstrate how important computer vision is to endeavors in business, entertainment, transportation, healthcare and everyday life. The following are some of the most common computer vision applications:

- One common application of computer vision is in autonomous vehicles, where cameras and other sensors are used to interpret the surrounding environment and make decisions about driving. Computer vision algorithms can help the vehicle detect lane markings, traffic signs, pedestrians, and other vehicles, enabling it to navigate safely and efficiently.
- Another application is in healthcare, where computer vision can be used for medical image analysis. For example, it can assist radiologists in interpreting X-rays, MRIs, and CT scans by automatically detecting anomalies or abnormalities. This can help improve diagnostic accuracy and speed up the process of diagnosing and treating patients.
- In retail, computer vision can be used for various purposes, such as tracking inventory, analyzing customer behavior, and enhancing the shopping experience. Retailers can use computer vision to monitor shelves and automatically restock products, or to analyze customer demographics and preferences based on their interactions with products in-store or online.

These are just a few examples of the many applications of computer vision across different industries. Technology continues to advance rapidly, opening up new possibilities for innovation and improving various aspects of our daily lives.

2.3 Computer vision algorithms:

Computer vision algorithms are a set of techniques and methods used to enable computers to interpret and understand visual information from the real world. These algorithms can be used for tasks such as :

2.3.1 Image classification:

Image classification is a fundamental task in computer vision that involves categorizing an image into a predefined set of classes or categories. The goal is to assign a label to an image based on its visual content. This can include identifying objects, scenes, or patterns within the image. To perform image classification, computer vision algorithms use machine learning techniques, such as deep learning models like Convolutional Neural Networks (CNNs). These models are trained on large datasets of labeled images to learn patterns and features that are characteristic of different classes.

During the training phase, the model learns to map input images to their corresponding labels through a process of optimization and adjustment of model parameters. Once the model is trained, it can be used to predict the class of new unseen images by analyzing their visual features and comparing them to the learned patterns.

Image classification has a wide range of applications, including object recognition, face detection, medical image analysis, autonomous driving, and more. It plays a crucial role in enabling machines to understand and interpret visual information, making it a key component of many computer vision systems. [4]

2.3.2 Image segmentation:

It is another computer vision task that involves partitioning an image into multiple segments or regions based on certain criteria, such as colors, textures, or shapes. The goal of image segmentation is to divide an image into meaningful and semantically coherent parts to simplify its analysis and interpretation.

There are several approaches to image segmentation, including: Thresholding, Clustering, Edge-based segmentation, Region-based segmentation, Semantic segmentation.[5]

2.3.3 Object detection:

Object detection is a fundamental task in computer vision with numerous applications. It is the task of identifying and locating objects of interest within an image or a video sequence.

The goal of object detection is not only to recognize the presence of objects in an image but also to accurately localize and classify them, this involves detecting multiple objects of different classes within an image and drawing bounding boxes around them to indicate their locations. [6]

Object detection typically involves the following steps:

- **Input Image:** An input image or video frame is provided for object detection.
- **Feature Extraction:** Features are extracted from the input image using techniques like convolutional neural networks (CNNs) to capture relevant patterns and characteristics of objects.
- **Object Localization:** Object detection algorithms predict bounding boxes that tightly enclose the detected objects in the image.
- **Object Classification:** The detected objects are classified into different categories or classes based on the features extracted from the image.
- **Post-processing:** post-processing steps may include non-maximum suppression (NMS) to remove redundant detections and improve the accuracy of object localization.

2.3.4 facial recognition:

Facial recognition in computer vision is the process of identifying or verifying a person's identity by analyzing and comparing their facial features. It involves detecting, extracting, and analyzing facial characteristics such as the distance between the eyes, the shape of the nose, and the contours of the face to create a unique facial signature for each individual. Facial recognition technology has gained significant popularity and adoption in various applications, including security systems, access control, surveillance, and personalized user experiences.

3 Machine Learning :

3.1 Definition :

Machine Learning is a scientific field that falls under the broader umbrella of artificial intelligence, focuses on developing algorithms and statistical models that enable computers to learn and make predictions or decisions from data with little human intervention and without being explicitly programmed, the algorithms process and analyze this data to uncover underlying patterns and relationships to make predictions based on.

Ultimately, machine learning has numerous applications, from predictive analytics to computer vision and natural language processing, revolutionizing industries and driving innovation. [7]

3.2 Types of Machine Learning:

Machine learning can be broadly categorized into several types, based on the presence or absence of human influence and the nature of the data:

3.2.1 Supervised learning:

In supervised learning, the algorithm is trained on a labeled dataset, The goal is to learn a mapping from input features to output labels or target values, enabling the algorithm to make predictions on new, unseen data.

Supervised learning can be separated into two types of problems when data mining:

- **Classification**: where the algorithm predicts discrete class labels.
- **Regression:** where the algorithm predicts continuous numerical values.

3.2.2 Unsupervised learning:

In supervised learning, the machine tries to find patterns in the unlabeled data and provides a response on its own, we have the input data but no output data.

Unsupervised learning models are used for three main tasks:

- **Clustering:** where the algorithm groups similar data points into clusters.
- **Association rules**: discover interesting relationships or associations between variables in large datasets.
- **Dimensionality reduction:** where the algorithm reduces the number of input features while preserving as much information as possible.

3.2.3 Semi-Supervised learning :

Semi-supervised learning is a type of machine learning that combines elements of both supervised and unsupervised learning, the algorithm learns from a dataset that contains both labeled and unlabeled examples, this combination allows machine learning algorithms to learn to label unlabeled data.

3.2.4 Reinforcement Learning:

Reinforcement learning enables an agent to acquire a behavior that has never been previously defined by humans, by interacting with its environment and learning from the consequences of its actions, receiving feedback in the form of rewards or punishments.

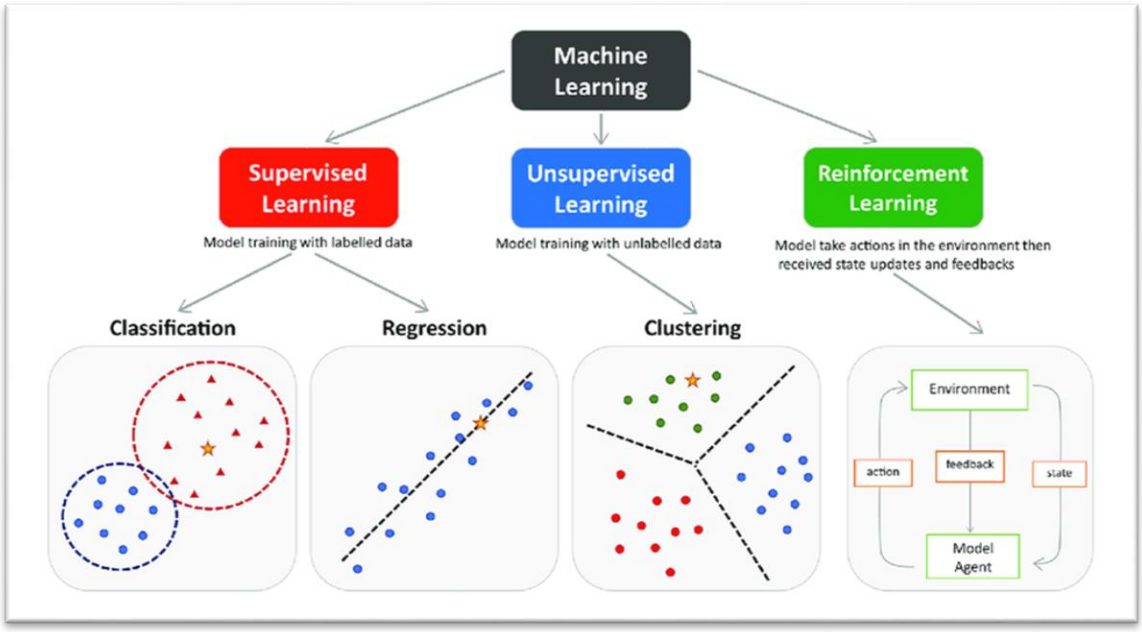

Figure 2. 2: Types of Machine Learning

3.3 Motivation to use ML:

By leveraging the power of ML, businesses can gain information from large and complex datasets, automate repetitive tasks, and make accurate predictions and decisions. the importance of ML lies in its ability to automate, personalize, innovate, secure, and solve complex problems, leading to increased efficiency, improved experiences, and advancements in various fields.

4 Deep Learning :

4.2 Definition :

Deep Learning is a branch of ML that enables a machine to learn from data using an artificialneural network, this last is inspired by the biological neural networks found in the human brain, it is able to simulate certain functions of the brain, such as pattern recognition and decision making.

Artificial neural networks consist of interconnected layers of neurons, which are capable of capturing complex relationships between data, allowing them to effectively handle tasks such as image and speech recognition, natural language processing, and many others. [8]

4.2 How does Deep Learning work?

The core model of deep learning consists of artificial neural networks (ANNs), which are organized into interconnected layers of neurons that are trained on large amounts of data to

learn patterns efficiently. This data is typically represented as numerical values, such as pixels in an image or features in text, and split into a training set and a validation set.

The training process involves iteratively presenting input data to the neural network. At each layer, the input is transformed according to the weights and biases of that layer and then passed to the next layer. This process is called forward propagation. Afterwards, the difference between the network's predictions and the actual output is calculated using the loss function. Based on this difference, the network's parameters (weights and biases) are then adjusted through an optimization algorithm called backpropagation, improving accuracy by performing this process over multiple epochs while minimizing the loss function. This process continues in subsequent hidden layers until the correct face is identified and sent to the output layer. Adding more hidden layers to a neural network can solve more complex problems.

Once the model is trained, it can be evaluated on a separate validation or test dataset to assess its performance. After evaluation, the trained model can be used to make predictions on new and unseen data.

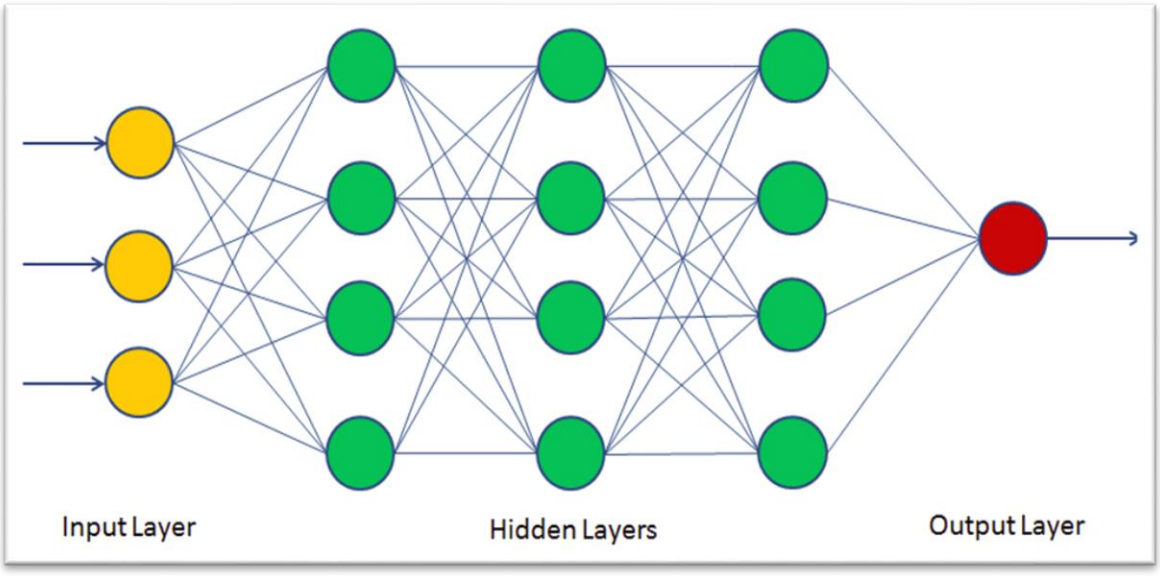

Figure 2. 3: Deep learning architecture

4.3 Deep learning models:

There are different deep learning models, each with their own strengths and weaknesses, characterized by their depth – focusing on CNNs –.

Common deep learning models include:

- Feedforward Neural Networks (FNN)
- Convolutional Neural Networks (CNN)
- Recurrent Neural Networks (RNN)
- Long Short-Term Memory Networks (LSTM)
- Generative Adversarial Networks (GAN)
- Capsule Networks (CapsNets)
- Spatial Transformer Networks (STN)
- Autoencoders
- Radial Basis Function Networks (RBFNs)
- Multilayer Perceptrons (MLPs)

4.4 Motivation to use DL:

Deep learning's ability to automatically identify complex patterns from data eliminates the need for human feature engineering, which is the driving force behind its application.

Deep learning models are excellent at handling high-dimensional data because of their flexibility and scalability, and they may be expanded to take on challenging tasks, it is a potent tool for solving problems in the real world and advancing various industries because of its versatility with massive datasets and ongoing improvement.

4.5 Deep Learning Requirements:

In order to have the best performance and outcomes, deep learning needs to meet key requirements. The following are some essential conditions:

- Data: large volumes of labeled data are required to learn patterns effectively.

High-quality data is essential for training accurate models across different domains, diversity is also important to ensure that the model can generalize well to unseen examples.

- Processing Capacity: Strong hardware accelerators like Graphics Processing Units (GPUs), such as the Tegra X1 mobile superchip with a 256-core GPU and an 8-core 64-bit CPU, or Tensor Processing Units (TPUs), such as Google Cloud TPU, are frequently needed for deep learning models due to their high computational demands, which make them ideal for training enormous datasets with intricate structures.
- Frameworks: Effective model implementation, training, and deployment are made possible by deep learning algorithms and frameworks, which offer the required infrastructure and tools. TensorFlow, PyTorch, Keras, and MXNet are well-known deep learning frameworks that support a variety of topologies and provide a large range of functionality.

4.6 Machine learning vs Deep learning:

Properties	Machine Learning (ML)	Deep Learning (DL)
Approach	Feature engineering, which involves manually extracting pertinent characteristics from the data for the algorithm's input by human experts, is a common method used by machine learning algorithms. These algorithms use statistical techniques to identify patterns and connections between features and labels	Uses multi-layered deep neural networks to automatically create hierarchical representations of features from raw data. Intricate patterns and representations can be found by deep learning algorithms straight from data, doing away with the requirement for human feature engineering
Data requirements	Although ML algorithms can handle high-dimensional data and frequently need hand-crafted features, they can also work with organized, semi-structured, and unstructured data	High-dimensional data, like text, music, and photos, works well with DL algorithms. Their ability to acquire hierarchical representations from unstructured data makes them especially suitable for unstructured data kinds
Hardware	Able to function on CPU	Demands a large amount of processing power
Model complexity	Often contain less parameters and simpler structures	Often surpass machine learning models in the number of parameters and complexity of structure.
Training time	Quick to train	High computational cost

Table 2. 1: Machine Learning Vs Deep Learning

High-dimensional data, like text, music, and photos, works well with DL algorithms. Their ability to acquire hierarchical representations from unstructured data makes them especially suitable for unstructured data kinds

5 Convolutional neural network:

5.1 Definition:

A Convolutional Neural Network (CNN) is a type of deep learning neural network specifically designed for processing structured grid-like data, such as images. CNNs are widely used in computer vision tasks due to their ability to automatically learn hierarchical representations of features from raw input data.

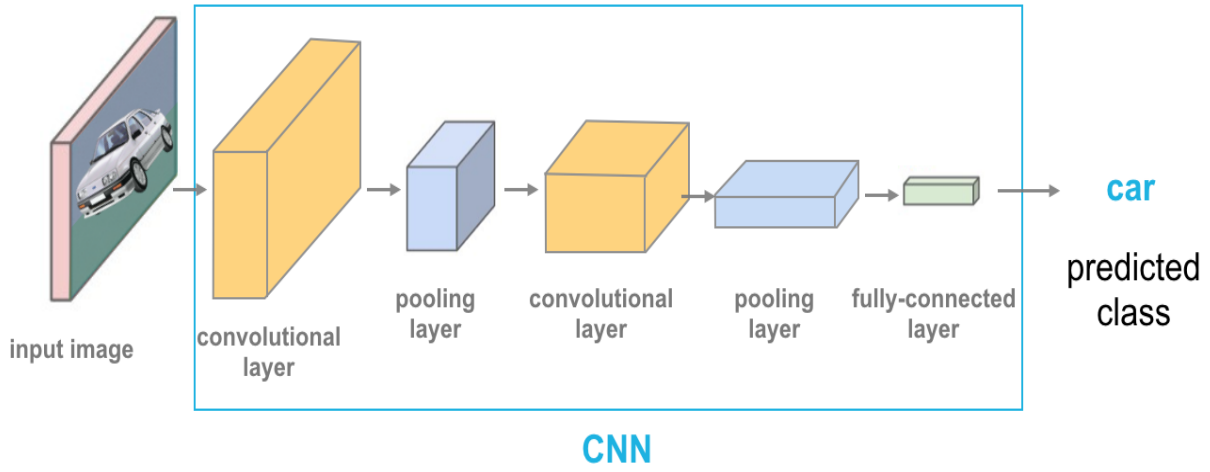

Figure 2. 4: CNN's structure

5.2 Convolutional neural network Architecture:

The architecture of a Convolutional Neural Network (CNN) typically consists of a series of layers that are designed to extract, and process features from input data, such as images. There are three types of layers that make up the CNN which are the convolutional layers, pooling layers, and fully connected (FC) layers. When these layers are stacked, a CNN architecture will be formed.

In addition to these three layers, there are two more important parameters which are the dropout layer and the activation function. [9]

5.2.1 Convolutional layer:

This layer is the first layer that is used to extract the various features from the input images. In this layer, the mathematical operation of convolution is performed between the input image and a filter of a particular size (M x M). By sliding the filter over the input image, the dot product is taken between the filter and the parts of the input image with respect to the size of the filter.

The output is termed as the Feature map which gives us information about the image such as the corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image. The convolution layer in CNN passes the result to the next layer once applying the convolution operation on the input.

5.2.2 Pooling layers:

Convolutional networks may include local and/or global pooling layers along with traditional convolutional layers. Pooling layers reduce the dimensions of data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. Local pooling combines small clusters, tiling sizes such as 2 × 2 are commonly used. Global pooling acts on all the neurons of the feature map. There are two common types of pooling in popular use: max and average. *Max pooling* uses the maximum value of each local cluster of neurons in the feature map, while average pooling takes the average value.

5.2.3 Fully connected layers:

Fully connected layers connect every neuron in one layer to every neuron in another layer. It is the same as a traditional multilayer perceptron neural network (MLP). The flattened matrix goes through a fully connected layer to classify the images. In this stage, the classification process begins to take place. The reason two layers are connected is that two fully connected layers will perform better than a single connected layer. These layers in CNN reduce human supervision.

5.2.4 Dropout Layer:

The Dropout layer is a mask that nullifies the contribution of some neurons towards the next layer and leaves all others unmodified - used to prevent overfitting-.

5.2.5 Activation Function:

- **SoftMax:** is often used as the activation function in the output layer of a CNN model for classification tasks. It is used to convert the raw output scores of the model into probabilities.
- **ReLU:** Rectified Linear Unit is one of the most widely used activation functions in CNNs. It sets all negative values to zero and keeps positive values unchanged. Mathematically, it is defined as f(x) = max (0, x).

6 Transfer Learning:

6.1 Definition:

Transfer learning is a machine learning technique where a pre-trained model that was trained on a large dataset for a specific task is used to train a new model on a different but similar task.

The knowledge encoded in the pre-trained model is transferred to the new task, either by using the pre-trained model as a feature extractor or by fine-tuning its parameters on the new dataset, rather than training the new model from scratch.

Transfer learning in the context of deep learning is applying the knowledge that a pre-trained model has learnt on a big dataset to solve a different task that may require less data.[10]

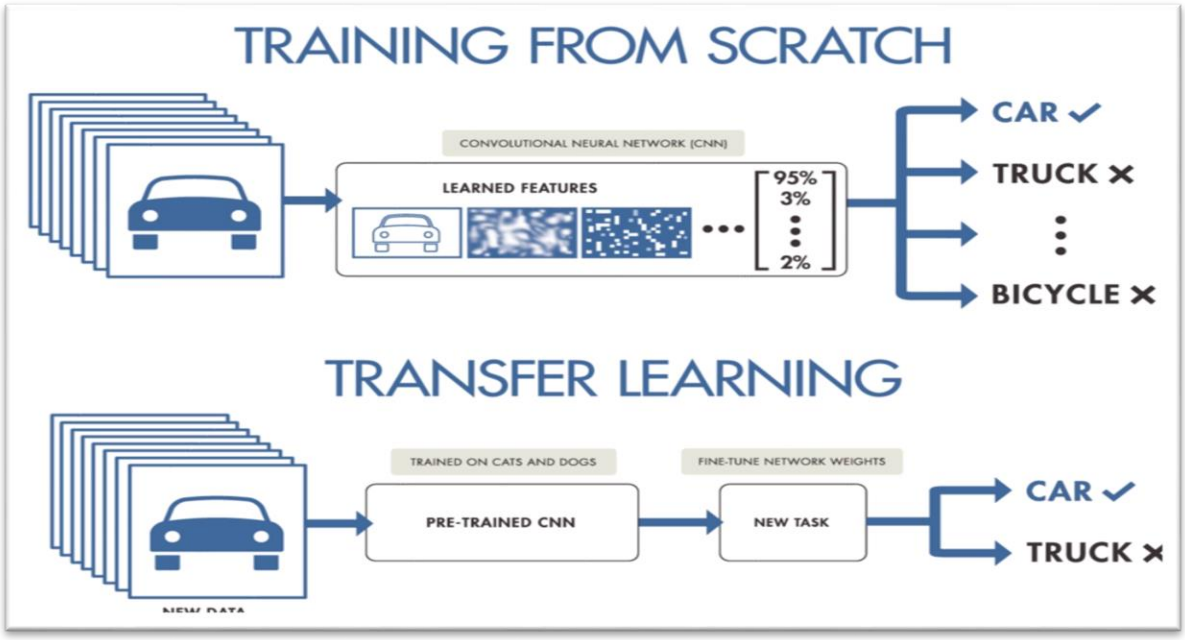

Figure 2. 5: Transfert Learning

6.2 Transfer Learning Types:

The way that knowledge is transmitted from the pre-trained model to the target task can be used to classify distinct types of transfer learning.

The primary forms of transfer learning are as follows:

- **Inductive Transfer Learning:** Applying a pre-trained model information to a new task with a distinct input feature space and output space is known as inductive transfer learning. The input-output mapping of the new task is fitted to the learned representations of the pre-trained model by adaptation or fine adjustments.
- **Transductive Transfer Learning:** Transferring information from a pre-trained model to a new task where the input feature space is the same, but the output space is distinct, is the focus of transductive transfer learning. The learnt representations of the pre-trained model are applied to the new task without any additional modification.
- **Unsupervised Transfer Learning:** Transferring information from a trained model to a new task in which labeled data is insufficient or unavailable is known as unsupervised transfer

learning. To acquire generic representations of the data, the pre-trained model is trained on an unsupervised task (autoencoder, clustering, etc.) and then applied to the new task.

- **Semi-Supervised Transfer Learning:** By using both labeled and unlabeled data from the domain of the pre-trained model, semi-supervised transfer learning combines elements of supervised and unsupervised learning to enhance performance on the new task. When there is a lack of labeled data for the target task, this strategy is useful.
- **Domain Adaptation:** The goal of domain adaptation is to transfer information from a trained model to a new task in which the data distributions are different, but the input-output space remains the same. The learnt representations of the pre-trained model are adjusted, usually via adversarial training or domain alignment, to match the data distribution of the target domain.
- **Multi-Task Learning:** In multi-task learning, a single model is trained to do several related tasks at once. As the model has the ability to combine between knowledge of different tasks and make use of the collaborative training process, transfer learning is inadvertently implicated in multitask learning.

The choice of transfer learning approach depends on factors such as the availability of labeled data, similarity between the source and target tasks, and domain shift between datasets.

6.3 Why using Transfer Learning?

Transfer learning is a useful technique in machine learning and deep learning since it provides several benefits:

- Transfer learning enables the use of expertise from sizable and labeled datasets in one domain to enhance performance on tasks in another domain with very less labeled data. Through the application of pre-trained model knowledge transfer, the time and resources required for data collection and annotation can be reduced for practitioners by achieving good performance on fewer datasets.
- Transfer learning speeds up training and uses less computational power than starting from scratch when training models by reusing learnt representations. Applications requiring predictions in real-time or almost real-time would particularly benefit from this.
- Transfer learning helps models identify stronger and more significant patterns in the data by utilizing information from similar tasks or domains, which improves performance on unobserved samples.

- Transfer learning helps models adapt to new datasets or settings by optimizing the pre-trained model's learnt representations to match the target domain's distribution.
- Pre-trained models serve as effective initializations for the model's parameters, helping to avoid the need for random initialization and enabling faster convergence during training, this is especially useful in scenarios where limited computational resources or time constraints prevent training large-scale models from scratch.

6.4 Transfer Learning Applications:

Transfer learning is a very effective method for utilizing current knowledge and skills to successfully take on new and difficult tasks due to its adaptability**.** Some notable uses include:

- **Computer Vision:** Pre-trained convolutional neural networks (CNNs), like those trained on ImageNet, are commonly used. These models are fine-tuned for specific object recognition tasks, it also aids in constructing efficient image classifiers by adapting pre-trained models to new classes or domains with minimal training data. Furthermore, transfer learning finds application in image segmentation tasks.
- **Natural Language Processing (NLP):** Pre-trained language models like BERT or GPT are fine-tuned on domain-specific text data, enhancing performance in tasks such as sentiment analysis, spam detection, or topic classification. Transfer learning also makes Named Entity Recognition (NER) easier. These models are efficient at identifying names of entities like people, organizations, and locations in text documents. Moreover, machine translation systems are greatly improved.
- **Healthcare:** Transfer learning significantly improves medical image analysis by enhancing tasks like tumor detection and classification in radiology. Furthermore, transfer learning helps to develop precise systems in the diagnosis of diseases and enabling predictions of diseases based on patient symptoms, genetic markers, or medical records.
- **Financial Forecasting**: Transfer learning improves stock market prediction by refining pre-trained models using past financial data. By increasing the accuracy of forecasting models, trading signals, market trends, and stock prices may all be forecasted more successfully.

6.5 Pre-trained Models:

Models like VGG, ResNet, Inception, and MobileNet have been pre-trained on the ImageNet dataset, which contains millions of labeled images across thousands of categories. These models are widely used for image classification tasks.

Some of these models are defined as follow:

A. VGG-Net:

The VGG (Visual Geometry Group) model, also known as VGG-Net, is a deep convolutional neural network architecture and a popular choice in the computer vision community for its simplicity, performance, and versatility. Developers and researchers often use pre-trained VGG models as a baseline or starting point for building more complex deep learning architectures or custom models tailored to specific tasks.

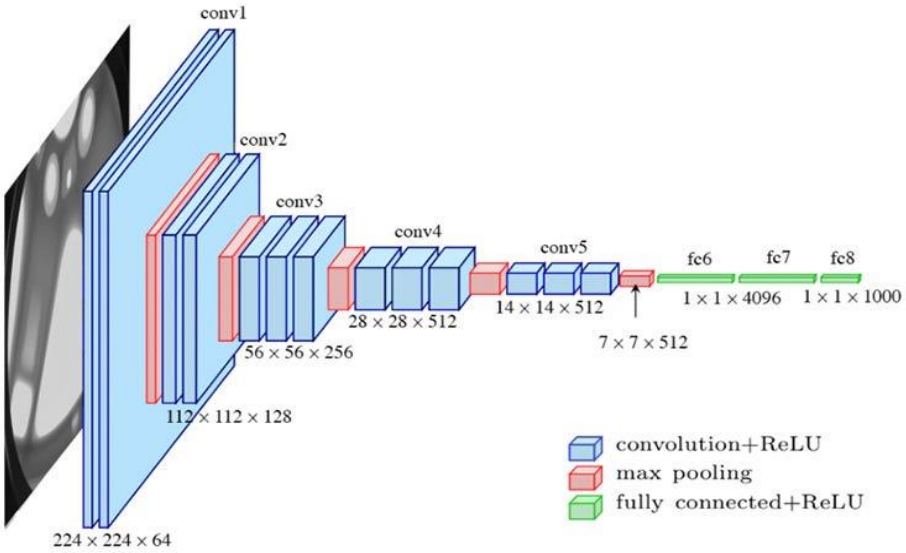

Figure 2. 6: VGG-Model Architecture

- **Architecture:** The VGG model consists of multiple convolutional layers followed by max-pooling layers. The network architecture is characterized by its deep stack of convolutional layers with small 3x3 filters and 2x2 max-pooling layers.
- **Variants:** The original VGG model comes in different variants, such as VGG16 and VGG19, which refer to the number of convolutional and fully connected layers in the network. VGG16 has 16 layers (13 convolutional and 3 fully connected), while VGG19 has 19 layers (16 convolutional and 3 fully connected).
- **Pre-training:** The VGG model was pre-trained on the ImageNet dataset, a large-scale dataset with millions of labeled images across thousands of categories. Pre-trained VGG models are widely used for image classification tasks due to their high accuracy and generalization capabilities.

- **Transfer Learning:** Due to its effectiveness in feature extraction, the VGG model is often used for transfer learning, where the pre-trained model is fine-tuned on a smaller dataset or a specific task to achieve better performance.
- **Applications:** The VGG model has been used in various computer vision tasks, including image classification, object detection, and image segmentation. It has also been employed in research and industry applications for tasks like medical image analysis, autonomous driving, and visual recognition.

B. Res-NET:

The ResNet model consists of a series of residual blocks, each containing multiple layers, typically convolutional layers, batch normalization, and activation functions such as ReLU. The key innovation of ResNet is the introduction of skip connections, also known as shortcut connections, which allow the output from one layer to bypass one or more layers and be added to the output of a later layer. This creates a residual learning framework where the model learns the residual mapping (the difference between the input and output) rather than the complete mapping.

By incorporating these skip connections, ResNet can effectively train very deep neural networks (hundreds or even thousands of layers) without suffering from issues like degradation in performance that typically arise when networks become too deep.

C. YOLO (You Only Look Once):

YOLO is a real-time object detection system that can detect multiple objects in an image with high accuracy. Pre-trained YOLO models are available for various versions, such as YOLOv3 and YOLOv4.

7 Infrared Imaging:

With the use of technology, we can detect, record, and display the infrared radiation that objects emit. This technique is called infrared imaging, or thermal imaging. It is a powerful technology that enables us to see beyond what is visible to the naked eye and provides valuable insights into temperature variations and thermal characteristics of objects and environments.

The underlying theory behind this kind of imaging is that everything which has a temperature higher than zero radiates infrared light. Using the temperature variations between the objects in the picture as a guide, infrared cameras use this radiation to form images.

Here are some key points about infrared imaging:

7.1 Principle:

The principle of infrared imaging is the detection of heat radiated by objects in the form of infrared light. Temperature fluctuations in a scene are visible because different things release varying amounts of infrared radiation depending on their temperature.

7.2 Applications:

There are many different industries where infrared imaging is used. Night vision systems, medical imaging, industrial inspections, firefighting, security, and environmental monitoring are among the many fields in which it finds frequent application.

7.3 Benefits:

The capacity of infrared imaging to take pictures in low- or no-light situations is one of its key benefits. In addition, it can recognize overheating parts of machinery, locate hidden objects, and offer useful data regarding temperature distributions.

7.4 Infrared Imaging Types:

Thermal infrared and near-infrared (NIR) imaging are the two primary forms of infrared imaging. While thermal infrared imaging is used to record temperature differences, near-infrared imaging is employed for tasks like night vision and surveillance.

7.5 Image Processing:

To improve the visibility of temperature variations and details in the scene, infrared photographs are usually processed using specialist software. Through the application of image processing techniques, infrared data may be converted into meaningful information and image quality can be enhanced.

8 Conclusion:

In conclusion, the area of computer vision has been considered a revolutionary transformation with the incorporation of machine learning techniques, specifically Convolutional Neural Networks (CNNs), and transfer learning. Deep learning models, such as CNNs, have shown remarkable performance in different tasks, and through transfer learning, performance on new tasks with sparse data can be improved by utilizing pre-trained models on large datasets. The convergence of deep learning, transfer learning, CNNs, computer vision, and machine learning has created new and intriguing opportunities across a range of fields, with the potential to revolutionize industries and improve our interaction with visual data in previously unheard-of ways.

Chapter 3 Related Works

1. Introduction:

In order to better understand the field of wildfire research, this chapter looks into techniques for segmentation, classification and detection, which are crucial for managing wildfires efficiently. We explore the available datasets that support these efforts, making it easier to create and test algorithms and models. Our goal is to provide light on the current status of wildfire research and provide directions for future investigation and innovation by combining findings from various studies.

2 The most significant techniques previously employed:

2.1 Wildfire classification:

2.1.1 Binary classification:

In the last few years, driven by their great success, researchers published a number of CNN-based advances in the literature for aerial image-based fire classification, these are listed in Table 1:

Ref.	architecture	Dataset	Accuracy (%)
Lee & Kim [11]	AlexNet GoogLeNet Modified GoogLeNet VGG13 ModifiedVGG13	Private dataset: 23,053 images	94.80 99.00 96.90 86.20 96.20
Shamsoshoara, & Afghah [12]	Xception	FLAME dataset: 48,010 images	76.23

Zhao & Ma [13]	Fire_Net AlexNet	UAV_Fire dataset: 1540 images	98.00 97.10
Treneska & Stojkoska [14]	VGG16 VGG19 ResNet50 InceptionV3 Xception	FLAME dataset: 8617 images	80.76 83.43 88.01 87.21 81.30
Srinivas & Dua [15]	Fog computing and simple CNN	Private dataset: 2964 images	95.07

Table 3. 1: Deep learning methods for UAV-based fire binary classification.

Several studies have proposed the use of deep convolutional neural networks (CNNs) to detect wildfires in aerial images. Lee et kim.[11] utilized five deep CNN models, including GoogLeNet and VGG13, to detect forest fires, with data augmentation techniques improving accuracy. Shamsoshoara et Afghah [12] developed a method based on the Xception model for wildfire classification, achieving an accuracy of 76.23% using the FLAME dataset. Treneska & Stojkoska [29] employed four deep CNNs, such as InceptionV3 and ResNet50, for wildfire detection in aerial images. ResNet50 achieved the highest accuracy of 88.01% in detecting forest fires using transfer learning with the FLAME dataset. Srinivas et al. proposed a CNN model integrated with Fog computing for early-stage forest fire detection, achieving an accuracy of 95.07%.

In another work, the researcher Rafik Ghali,and Moulay A. Akhloufi ,and Wided Souidene Mseddi trained wildfire classification methods on aerial images collected using the Matrice 200 drone with a Zenmuse X4S camera. Testing data are collected using the Phantom drone with a Phantom camera. [28]

With an accuracy of 85.12% and an F1-score of 84.77%, Table 2 demonstrates that the suggested techniques outperform deep CNN models. It achieved higher results for object classification than other recent models by using the EfficientNet-B5 and DenseNet-201 models to extract scaled and diversified feature maps. Even for minor fire areas, the suggested method showed good skill in identifying and categorizing forest fires in aerial pictures, despite a high inference time.

Overall, the proposed method strikes a balance between performance and inference time, highlighting excellent capabilities in classifying forest fires in aerial images.

architecture	Accuracy (%)	F1-Score (%)	Inference Time (s)
Xception	78.41	78.12	0.002
Xception [12]	76.23	—	—
EfficientNet-B5	75.82	73.90	0.010
EfficientNet-B4	69.93	65.51	0.008
EfficientNet-B3	65.81	64.02	0.004
EfficientNet-B2	66.04	60.71	0.002
InceptionV3	80.88	79.53	0.002
DenseNet169	80.62	79.40	0.003
MobileNetV3-Small	51.64	44.97	0.001
MobileNetV3-Large	65.10	60.91	0.001
Proposed ensemble model	85.12	84.77	0.018

Table 3. 2: Performance evaluation of wildfire classification models on the FLAME dataset.

The architecture shown in Figure 1 was experimented with, using well-known deep CNN architectures as feature extractors for wildfire classification on infrared black hot images. The hyperparameters chosen in the experiments are demonstrated in Table 3

Due to the imbalanced nature of the infrared black hot dataset, class weight was imposed with a specified formula during training. Strategies determined for forest fire classification with heat-sensitive infrared images led to superior performance in the methods experimented with. Deeper networks were observed to provide better results among the models used as feature extractors in experiments with Deep Neural Networks. The comparative test performance of the experimented models is shown in Table 4.[29]

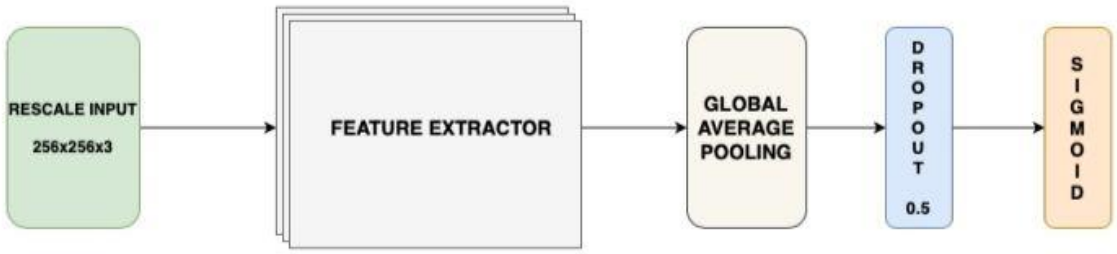

Figure 3. 1: General Architecture of CNN Based Models

Hyperparameters	Metric *Hyperparameter Space*
Input Shape	256×256×3
Optimizer	Adam
Learning Rate	0.0001
Loss	Binary Crossentropy
Last Layer Activation Function	Sigmoid

Table 3. 3: Hyperparameters used in the experiments

The test scores obtained by the trained models are close. In the experiments, The ResNet101 model attained the highest performance with 99.20% test accuracy and 98.77% F1 score, in this work Aerospace Industries AKSUNGUR UAV EO/IR cameras were used as the dataset.

Architecture	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
ResNet101	99.20	99.11	99.63	98.77
ResNet50	98.04	98.05	98.04	98.04
VGG16	98.56	98.57	98.57	98.56
EffiientNetB1	98.54	98.53	98.54	98.53
Densnet121	97.24	97.35	97.24	97.27

Table 3. 4: Experimental results obtained from the MALE (medieum altitude long endurance) class Turkish.

2.1.2 Multiclass classification:

A recent study presents a novel approach to classify images related to wildfires into five different subclasses according to recognized confounding factors. The goal of these subclasses is to provide an even distribution of photos showing occurrences related to fire or smoke and those that don't. This classification framework makes it easier to comprehend and classify wildfire-related images by offering an organized method of analysis. The following subclasses are included :

- **Smoke from fires (subclass 1):** Images in this class depict smoke emissions from fires without visible flames.
- **Both smoke and fire (subclass 2):** This class includes images showing both flames and smoke emissions from fires.
- **Forested areas without confounding elements (subclass 3):** Images without any identified confounding elements, representing typical forested areas, belong to this class.
- **Fire confounding elements (subclass 4):** Images containing elements that could be mistaken for fire are categorized here.
- **Smoke confounding elements (subclass 5):** Images featuring elements that may be misinterpreted as smoke are placed in this class.

Upon applying the MobileNetV3 model to the wildfire dataset, the authors observed significant potential for enhancements in the model's capacity to detect forest fires. This included the skilled handling of complex variables known as confounding elements within the images. This improvement was realized through a novel two-step multi-class classification strategy. The application of this approach strengthened the model's robustness, reduced false alarms, and increased its ability to adapt to different and often challenging situations specific to RGB forest fire detection.[16]

Hyperparameter	Value
Learning Rate	1×10^{-2}
Optimizer	Adam
Batch Size	32
Epochs	30
Fine tune at the layer number	1

Figure 3. 2: Hyperparameter configuration for the used Method

The following results are based on five runs for each model, with bootstrapping used to calculate 95% confidence intervals for the differences in performance metrics between the two models:

Metrics	**Augmented Wildfire Dataset**	**Original Wildfire Dataset**
Primary Accuracy	Mean = 0.8766 Std = 0.0147	Mean = 0.9073 Std = 0.0077
Precision	Mean = 0.7974 Std = 0.0206	Mean = 0.8832 Std = 0.0263
Recall	Mean = 0.9171 Std = 0.0273	Mean = 0.8855 Std = 0.0247
F1-Score	Mean = 0.8526 Std = 0.0146	Mean = 0.8839 Std = 0.0201
ROC-AUC	Mean = 0.8842 Std = 0.0127	Mean = 0.9053 Std = 0.0164

Table 3. 5: Performance metrics using augmented and original wildfire datasets

2.2 Wildfire Detection:

Region-based CNNs are utilized for object detection in images by determining objects' locations through bounding boxes. They are divided into one-stage detectors, which directly

detect objects, and two-stage detectors, which first generate region proposals and then classify them. Region-based CNNs have shown high accuracy in object detection tasks and have been successful in detecting fires in aerial images.

Table 6 summarizes deep learning methods for UAV-based fire detection. Jiao et al. utilized the one-stage detector YOLOv3 to detect forest fires. YOLOv3 improves detection performance by making detections at three scales and using the Darknet-53 model with 53 convolutional layers. Testing showed great performance and high speed. Jiao et al. also introduced the UAV-FFD platform, which uses YOLOv3 to detect smoke and fire in UAV-acquired images. YOLOv3 demonstrated high performance with reduced computational time, achieving an F1-score of 81% at 30 frames per second, proving its effectiveness in real-time UAV applications.

Alexandrov et al. utilized SSD, YOLOv2, and Faster R-CNN detectors to detect wildfires, with YOLOv2 demonstrating superior performance in detecting smoke early on. Tang et al. developed a novel approach using 4K UAS images, implementing a coarse-to-fine strategy involving an Adaptive sub-Region Select Block to select subregions of interest, followed by YOLOv3 for object detection. This method achieved a mean average precision of 67% with an average speed of 7.44 frames per second using 1400 4K aerial images.

Ref.	Architecture	Smoke/Flame	Dataset	Results (%)
Jiao & Zhang [30]	YOLOv3	Flame	Private dataset: 3,840,000 images	F1-score = 81.0 Precision = 0.82 Recall = 0.79 Accuracy= —
Alexandrov & Pertseva [31]	YOLOv2 Faster R-CNN SSD	Smoke	Private dataset: 12,000 images	Accuracy = 98.3 Accuracy = 95.9 Accuracy = 81.1
Jiao &Zhang [32]	YOLOv3	Flame/Smoke	Private dataset: 3,684,000 images	F1-score = 81.0 Precision =0.84 Recall= 0.78 Accuracy= —
Tang & Liu [33]	YOLOv3 and ARSB method	Flame	Private dataset: 1400 K images	mAP = 67.0 Accyracy= ---

Table 3. 6: Fire detection using Deep learning methods for UAVs.

They tested the performance of their flame detection system equipped with the MSER-NMS operation in several scenarios and evaluated its accuracy and robustness. Thus, their proposed framework is simple, stable, computationally efficient, and labor-free. A hierarchical method can be implemented by first classifying frames into fire versus no-fire,

and then running the detection algorithm only on fire frames, to be more computationally efficient. In order to verify the suitability of the proposed detection method in real-time scenarios, the processing speed of the proposed method with real wildfire images was examined. It was observed that the system achieves an average of 83 FPS on 39,700 flame images with 256p × 256p resolution, a frame much higher than the usual 30 FPS rate of most UAV cameras. The program has been tested on a PC with Intel i7-11700 CPU and 16GB memory. Since the computation time depends mainly on the performance of the CPU and RAM, they believe that it has the potential of being deployed on edge devices such as UAVs and mobile sites, toward making real-time aerial monitoring ubiquitous. [17]

2.3 Wildfire Segmentation:

Image segmentation is crucial in computer vision as it accurately identifies the shapes of objects in images. Deep learning models have made significant advancements in solving various problems related to image segmentation, including detecting fire pixels and accurately outlining smoke or flames in aerial images.

Table 7 shows deep learning methods for UAV-based fire segmentation. Barmpoutis and Stathaki [34] developed a 360-degree remote sensing system to segment fire and smoke using RGB images captured by a UAV. They utilized DeepLab V3+ models with ASPP to detect smoke and flame regions and implemented an adaptive post-validation scheme to filter out false-positive regions. Their method achieved an F1-score of 94.6% on urban and forest images, surpassing DeepLabV3+. In a similar vein, Shamsoshoara et al. used U-Net for wildfire segmentation, achieving an F1-score of 87.75%. Frizzi et al. proposed a VGG16-based method for smoke and fire segmentation with an accuracy of 93.4% and a segmentation time of 21.1 seconds per image, outperforming previous models through data augmentation techniques.

Ref.	architecture	Smoke/Flame	Dataset	Results (%)
Barmpoutis & Stathaki [34]	DeepLabV3+ DeepLabV3+ + validation approach	Flame/Smoke	Fire detection 360-degree dataset: 150 360 images	Accuracy = — F1-score = 81.4 Precision=68.9% Recall= 99.3% F1-score = 94.6 Precision=90.3% Recall=99.3%
Ronneberger & Fischer [27]	U-Net	Flame	FLAME dataset: 5137 images	Accuracy = — F1-score = 87.7

Frizzi & Bouchouicha [28]	U-Net CNN based on VGG16	Flame/Smoke	Private dataset: 366 images	Accuracy = 90.2 Accuracy = 93.4 F1-score = — Precision= — Recall= —

Table 3. 7: Fire segmentation using deep learning methods for UAVs.

Vision Transformers, specifically TransUNet and TransFire, outperformed deep CNN models like EfficientSeg and U-Net due to their ability to capture long-range interactions in input features and extract fine details from images. TransUNet-R50-ViT achieved the highest performance with 99.9% accuracy and F1-score by leveraging local and global features from a hybrid backbone comprising a CNN, R-50, and pretrained ViT Transformer. Both TransUNet-R50-ViT and TransUNet-ViT demonstrated accurate segmentation of wildfires, especially in detecting small fire areas. TransFire also showed high accuracy (99.83%) and F1-score (99.82%) by extracting high-level information and finer features to segment wildfire pixels effectively. EfficientSeg performed well with 99.63% accuracy and 99.66% F1-score but had a higher inference time compared to vision transformers. Overall, TransUNet, TransFire, and EfficientSeg showcased excellent performance in segmenting wildfires and identifying precise fire shapes, with TransFire standing out for its performance close to vision transformers and superior detection of small fire areas under challenging conditions like smoke presence

architecture	**Accuracy (%)**	**F1-Score (%)**	**Inference Time (s)**
TransUNet-R50-ViT	99.90	99.90	0.51
TransUNet-ViT	99.86	99.86	0.40
TransFire	99.83	99.82	1.00
EfficientSeg	99.63	99.66	1.38
U-Net	99.00	99.00	0.29

Table 3. 8: Performance evaluation of wildfire segmentation models on the FLAME dataset.

Table 8 illustrates the quantitative results of fire segmentation using the FLAME dataset. We can see that TransUNet, TransFire, and EfficientSeg obtained excellent results and outperformed U-Net used as a baseline model. [28]

3 The most widely used databases on this topic:

1 Short and Cortez datasets :

Following a review of the literature, two databases that are publicly accessible were found to contain over 1.04 million geo-referenced wildfires that burned over 359,000 km2 (88.7 million acres) in the United States (Short, 2017) and Portugal (Cortez and Morais, 2007a) between 1992 and 2015. These databases include information on well-recorded wildfires with a range of features. These databases are going to be utilized in different machine learning investigations [18].

The first database covers major wildfires in the US from 1992 to 2015, comprising 1.04 million records over 88.7 million acres. It includes 50 parameters, such as location, cause, and size, and can be accessed freely. The second database, curated by Cortez and Morais (2007b, 2008), focuses on wildfires in Portugal's Montesinho natural park from 2000 to 2003. It comprises 517 incidents and includes attributes like geographic features, weather variables, and fire danger indices. The main predictor is the size of burned areas, ranging from 0 to 11 km2 (0–2695.5 acres).

2 A Kaggle Dataset:

Another database includes 2165 web-based annotated photos, 1000 images from the KTS dataset, 101 images from the DNIM dataset, 254 images for error correction, and 280 cropped images taken by researchers. Classes like "Wildfire," "Non-Fire," "Flame," "Smoke," "Building," and "Pedestrian," are marked on each instance. Every annotated image was subjected to two rounds of verification by two distinct writers in order to guarantee annotation correctness.

The researchers used cropped images of a controlled fire in the forest taken by a drone, together with daytime and nighttime wildfire images taken by security cameras or drones that they got from the Internet, to train and test the deep learning model. The Korean tourist spot (KTS) datasets created for deep learning research, which comprise images linked by forest labels containing significant wooden cultural properties in the forest, and the day-night image matching (DNIM) dataset, which was used to lessen the effects of day and night lighting changes, were also included in this study. Furthermore, photographs resembling wildfires, together with 154 clouds and 100 sun images, were also incorporated into the datasets due to their shared characteristics with the smoke and flames of early wildfires, such as their hue or shape. [19]

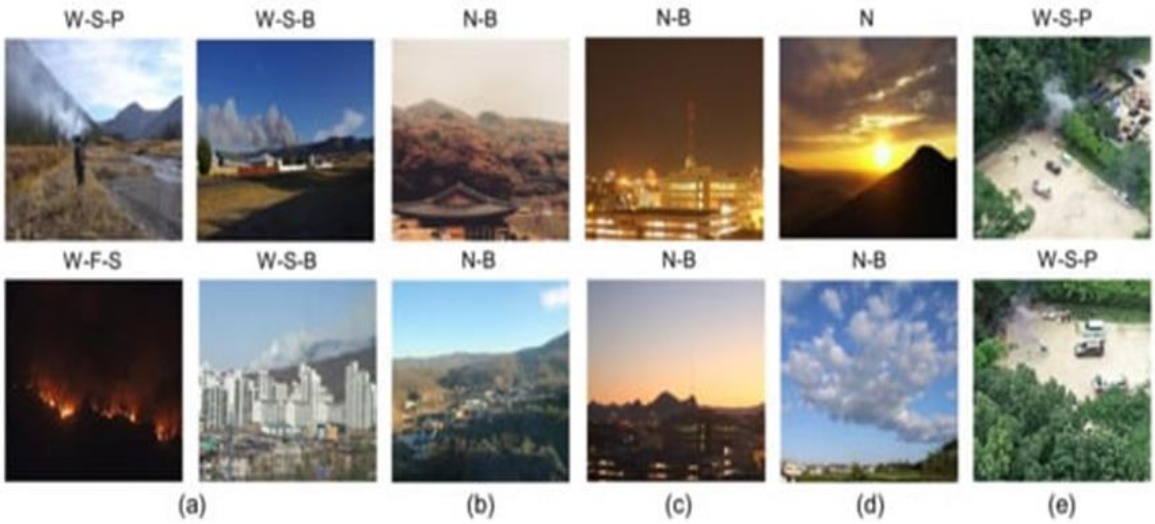

Figure 3. 3: Resized sample of annotated images for MLC: (a) downloaded from the Web; (b) KTS dataset; (c) DNIM dataset; (d) dataset for error protection purposes; (e) dataset of a controlled fire captured by researchers.

3 The Wildfire Dataset:

The wildfire dataset is an example of a serious attempt to investigate the potential of RGB images in the field of machine learning techniques for wildfire detection. It comes with 2,700 aerial and ground photos that have been selected from a wide range of internet sources, including government databases, Flickr, and Unsplash. The dataset is a comprehensive baseline for research since it captures a wide range of environmental circumstances, forest types, geographical regions, and the complex dynamics of forest ecosystems and fire occurrences [16].

The dataset collection process was meticulously designed to encompass a broad spectrum of environmental and caption-related factors to enhance representativeness and generalizability. Environmental parameters included varied terrain features, canopy density, forest types, ground cover, natural and human-made elements, weather conditions, foliage changes, sunlight exposure, fire characteristics, and smoke dispersion patterns. Caption-related factors encompassed lighting conditions, image resolution, altitude and distance, camera angle and orientation, perspective, platform type, obstructions and reflections, image compression, and camera motion blur. This comprehensive approach ensures that the dataset captures diverse scenarios, facilitating robust analysis and modeling in wildfire research. A careful process guaranteed the quality of the dataset. Perceptual hashing and image comparison were used to address duplication and eliminate duplicates, improving the integrity of the dataset.

The wildfire dataset images were sourced from various online platforms, including government agencies, Flickr, and Unsplash. Prior to inclusion, rigorous verification of licensing and permissions was conducted to ensure adherence to intellectual property rights. All images in the dataset are dedicated to the public domain, guaranteeing legal compliance and enabling unrestricted sharing within the research community. Moreover, a Supplementary File will accompany the dataset, providing download URLs for each image, enhancing transparency and facilitating traceability to their original sources.

Figure 3. 4: Images from the introduced dataset depicting visual features that may be misinterpreted by the model as a forest fire scene.

4 Flame 2:

A collection of dual RGB/IR drone videos and images captured during a prescribed burn in northern Arizona in November 2021 is intended to be provided by the FLAME2 project. Another significant feature of this dataset is that it might be used as potential input for high-resolution burn intensity/severity modeling and fire spread. [17]

Although there are a lot of publicly accessible datasets on fire detection, very few provide aerial photos of wildfires or controlled burns. In addition, most UAV-collected fire image datasets currently available only contain thermal or color photos, not both. To fill this void, the FLAME2 dataset was developed with the goal of offering an extensive compilation of dual RGB/IR drone footage and pictures taken during a prescribed burn.

The FLAME2 supplemental dataset is distinct from existing datasets in its comprehensive inclusion of various data types beyond just image samples. It encompasses geo-referenced point clouds of the prescribed burn area, alongside weather and burn conditions information, and pre-burn aerial RGB videos. These additions significantly broaden the dataset's potential

applications beyond fire detection and masking. For instance, researchers can leverage this dataset to fuse thermal and color images, generating high-resolution, accurate 3-dimensional point clouds. Such data can be utilized to create burn intensity models or georeferenced temperature maps before and after fires occur. This comprehensive dataset offers valuable resources for various research endeavors related to fire management and analysis.

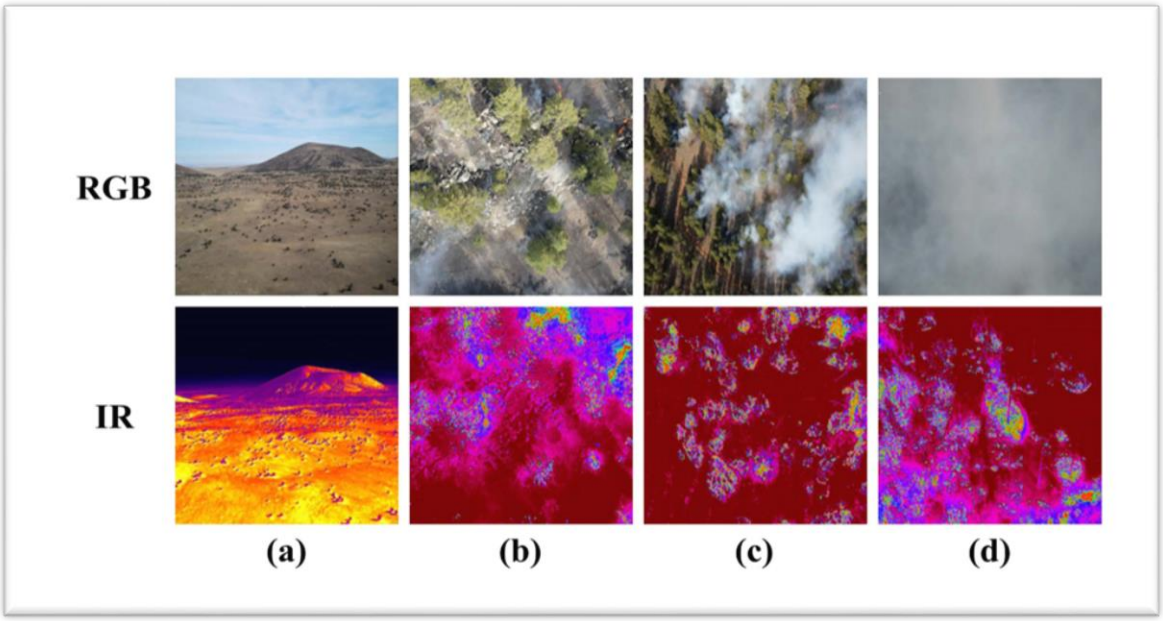

Figure 3. 5: Handpicked frame pairs from FLAME 2 Dataset [17]. (a) No flame with No smoke, (b) flame with no smoke, (c)(d) flame with smoke.

5 Next Day Wildfire Spread:

A carefully selected, extensive, multivariate data set of past wildfires compiles almost ten years' worth of remote sensing information across the US. This data set provides a feature-rich data set for machine learning applications by combining 2D fire data with numerous explanatory variables (such as topography, vegetation, weather, drought index, and population density) aligned over 2D regions, in contrast to other fire data sets based on Earth observation satellites. With a one-day lead time, this data collection can serve as a standard for creating wildfire propagation models based on remote sensing data [20].

This dataset, which includes 18,445 samples and covers the continental United States from 2012 to 2020, shows zones of 64 km x 64 km where fires occurred at a resolution of 1 km. To record patterns of fire spreading over time, each sample contains fire information encoded as a fire mask that indicates 'fire', 'no fire', or unsure labels in addition to 'prior fire mask' and 'fire mask'. Data from multiple sources is combined and superimposed with location- and

time-based fire data using Google Earth Engine (GEE). The dataset contains information on elevation, wind direction and speed, minimum and maximum temperatures, humidity, precipitation, drought index, normalized difference vegetation index (NDVI), energy release component (ERC), and population density in addition to fire data. This rich set of features makes it easier to create comprehensive wildfire prediction models.

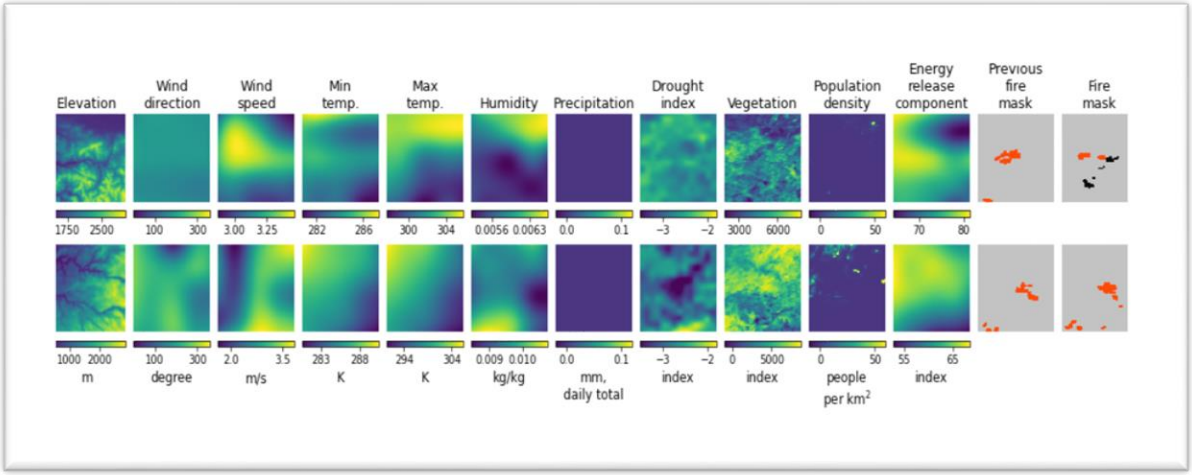

Figure 3. 6: examples from this data set. In the fire masks, red corresponds to fire, while grey corresponds to no fire. Black indicates uncertain labels (i.e., cloud coverage or other unprocessed data).

4 General Comparaison :

Tasks	Most effective models	Dataset	Data Type	Accuracy
Binary classification	GoogLeNet	Private dataset: 23,053 images	RGB	99%
	Fire_Net	UAV_Fire dataset: 1540 images	RGB	98%
	InceptionV3	aerial images collected using the Matrice 200 drone with a Zenmuse X4S camera	RGB	80.88%

	ResNet101	infrared black hot Dataset	Thermal	99.20%
Multiclass classification	MobileNetV3	The Wildfire Dataset	RGB	90%
Detection	YOLOv2 Faster R-CNN SSD	Private dataset: 12,000 images	RGB	98.3%
Segmentaion	TransUNet-R50-ViT	Fire detection 360-degree dataset: 150 360 images	RGB	99.9%
	TransUNet-ViT		RGB	99.86%

Table 3. 9: General comparison between the most effective models for different tasks on wildfire topic.

5 Conclusion:

Overall, this chapter aims to provide a comprehensive overview of the current state of wildfire detection research, highlighting key findings, trends, and future directions for advancing the field. Through a critical analysis of existing literature and methodologies, we hope to contribute to the ongoing efforts to develop more effective and efficient wildfire detection systems for better disaster prevention and management.

Chapter 4 Framework Implementation and Experimental Results

1 Introduction:

Our approach to wildfire detection involves utilizing transfer learning with the VGG16 convolutional neural network architecture and applied on two different datasets for binary classification. By fine-tuning the pre-trained VGG16 model on our datasets, we adapted its learned features to the task of distinguishing between wildfire flames and non-flame elements with remarkable precision. In addition, we have developed an innovative approach incorporating a fusion model for RGB and thermal images. By integrating information from both modalities, we were able to enhance the robustness and accuracy of our model, which can be also useful across various tasks and domains. The obtained results show the effectiveness of our models across the two datasets, which are very challenging for wildfire classification, where the Wildfire Dataset includes many confounding effects which could affect the accuracy of any model, and the Flame-2 requires careful analysis of both RGB and thermal images in a complementary way.

Subsequently, we developed two prototypes, for web and mobile applications, that integrate these trained models to facilitate real-time wildfire detection. These applications offer users an intuitive interface for monitoring, providing timely alerts and visualizations to enable swift response and mitigation efforts, thereby enhancing community safety and environmental conservation.

2 Datasets Overview:

In our methodology, we leverage three high-quality datasets to enhance model generalization, enabling strong performance across various scenarios and environments. Through exposure to diverse data sources, the model becomes able at identifying meaningful patterns and features applicable across different contexts, bolstering its resilience and adaptability. Furthermore, training with a range of datasets serves to mitigate biases inherent in individual datasets, leading to a more impartial and fairer model. Ultimately, this approach reinforces robustness, and fairness in the model, amplifying its efficacy and utility in real-world applications.

- **“The Wildfire Dataset” [21]**: A big collection spanning 2,700 aerial and ground-based images. To improve representativeness and generalizability, the process of collecting the dataset was carefully planned to cover a wide range of environmental and caption-related

● parameters. The environmental parameters encompass a range of features such as different terrain, forest kinds, canopy density, ground cover, natural and man-made elements, weather, changes in foliage, exposure to sunlight, fire characteristics, and smoke dispersion patterns. It is divided as shown in the figure:

Set	Class	Subclass	Instances
Training	Nofire	Forested areas without confounding elements	591
	Nofire	Fire confounding elements	254
	Nofire	Smoke confounding elements	330
	Fire	Smoke from fires	463
	Fire	Both smoke and fire	286
Validation	Nofire	Forested areas without confounding elements	127
	Nofire	Fire confounding elements	50
	Nofire	Smoke confounding elements	69
	Fire	Smoke from fires	99
	Fire	Both smoke and fire	57
Test	Nofire	Forested areas without confounding elements	128
	Nofire	Fire confounding elements	52
	Nofire	Smoke confounding elements	71
	Fire	Smoke from fires	100
	Fire	Both smoke and fire	59

Table 4. 1: Data distribution across dataset's subclasses.

● **"The flame-2 RGB images" [22]:** A collection that contains more than 53K frames captured from videos. The FLAME2 supplemental dataset is different from other datasets since it includes a wide range of data kinds, not simply image samples. It is a significant and useful database that contains geo-referenced point clouds of the particular burn area, weather data, burn conditions, and pre-burn atmospheric RGB movies, making it an important and valuable database.

● **"The flame-2 Thermal images" [22]:** 53 451 frames in a collection which have the same content as flame-2 RGB images, thermal images are essential in wildfire detection due to their ability to capture heat signatures emitted by fires, even in conditions of low visibility such as heavy smoke or darkness. This capability enables early detection and intervention, helping to prevent the rapid spread of wildfires and making this dataset more important and influential.

Data Name	Data Type	Camera	File Type	Resolution (p x p)	Duration	FPS	File Size	Labeled
1) Video Pair 1	RGB Video IR Video	M2EA RGB Cam M2EA IR Cam	.mp4 .mp4	3840 x 2160 640 x 512	291 seconds 291 seconds	30 30	3.50 GB 322 MB	N N
2) Video Pair 2	RGB Video IR Video	M2EA RGB Cam M2EA IR Cam	.mp4 .mp4	3840 x 2160 640 x 512	183 seconds 183 seconds	30 30	2.20 GB 199 MB	N N
3) Video Pair 3	RGB Video IR Video	M2EA RGB Cam M2EA IR Cam	.mp4 .mp4	1920 x 1080 640 x 512	404 seconds 404 seconds	30 30	1.69 GB 446 MB	N N
4) Video Pair 4	RGB Video IR Video	M2EA RGB Cam M2EA IR Cam	.mp4 .mp4	1920 x 1080 640 x 512	301 seconds 301 seconds	30 30	1.26 GB 333 MB	N N
5) Video Pair 5	RGB Video IR Video	M2EA RGB Cam M2EA IR Cam	.mp4 .mp4	1920 x 1080 640 x 512	267 seconds 268 seconds	30 30	1.12 GB 297 MB	N N
6) Video Pair 6	RGB Video IR Video	M2EA RGB Cam M2EA IR Cam	.mp4 .mp4	1920 x 1080 640 x 512	185 seconds 185 seconds	30 30	800 MB 206 MB	N N
7) Video Pair 7	RGB Video IR Video	M2EA RGB Cam M2EA IR Cam	.mp4 .mp4	1920 x 1080 640 x 512	239 seconds 240 seconds	30 30	1.00 GB 266 MB	N N
8) Original Resolution Frame Pairs	RGB Frames RGB Frames IR Frames	M2EA RGB Cam M2EA RGB Cam M2EA IR Cam	.jpg .jpg .jpg	3840 x 2160 1920 x 1080 640 x 512	13,700 frames 39,751 frames 53,451 frames	- - -	58.3 GB 46.6 GB 21.6 GB	Y Y Y
9) Scaled Resolution Frame Pairs	RGB Frames IR Frames	M2EA RGB Cam M2EA IR Cam	.jpg .jpg	254 x 254 254 x 254	53,451 frames 53,451 frames	- -	3.04 GB 5.30 GB	Y Y
10) Frame Pair Labels	text file	-	.txt	-	-	-	543 B	-

Table 4. 2: FLAME-2 dataset contents

3 Our Proposed Models:

3.1 First framework (A Fine-tuned VGG16 model on the Wildfire Dataset) [23] :

We developed one general model with two separate fine-tuning procedures, and we had to modify the parameters in order to match each dataset.

A VGG16 model requires several important stages to be fine-tuned. First, we imported libraries, and the VGG16 model that had already been trained, using its weights which was trained on the large “ImageNet” dataset.

```
import tensorflow as tf
import numpy as np
from tensorflow import keras
import os
import cv2
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten
from keras.applications import VGG16
```

Figure 4. 1: Importing Necessary Libraries

Next, we get ready to work with the dataset by creating separate datasets for training, testing, and validation.

```
train = ImageDataGenerator( rescale=1./255)
test = ImageDataGenerator(rescale=1/255)
valid=ImageDataGenerator(rescale=1/255)

train_dataset = train.flow_from_directory(r'C:\Users\Pc Maker\Desktop\memoire code\archive\the_wildfire_dataset\the_wildfire_dataset\train',
                                          target_size=(224,224),
                                          batch_size = 32,
                                          class_mode = 'binary')

test_dataset = test.flow_from_directory(r'C:\Users\Pc Maker\Desktop\memoire code\archive\the_wildfire_dataset\the_wildfire_dataset\test',
                                          target_size=(224,224),
                                          batch_size =32,
                                          class_mode = 'binary')

valid_dataset = valid.flow_from_directory(r'C:\Users\Pc Maker\Desktop\memoire code\archive\the_wildfire_dataset\the_wildfire_dataset\val',
                                          target_size=(224,224),
                                          batch_size =32,
                                          class_mode = 'binary')
```

Figure 4. 2: Preparing datasets to be processed

After that, we modified the model's top layers to correspond with the number of our two classes (Fire/No-Fire).

```
base_model = VGG16(weights='imagenet', include_top=False, input_shape=(224, 224, 3))
```

Figure 4. 3: Importing VGG16 without its top layers

To avoid overfitting, we then froze a few pre-trained layers. This is a crucial step when working with small amounts of data. Next, we include other layers, including a Dropout layer for 50% regularization and a Flatten layer, which are followed by 256 dense layers with suitable activations ("ReLU"). Then we add a dense layer with a sigmoid activation function to output the final classification probabilities with respect to the two classes (Fire and No-Fire).

```
for layer in base_model.layers:
    layer.trainable = False

model = Sequential()
model.add(base_model)
model.add(Flatten())
model.add(Dense(256, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))
```

Figure 4. 4: Including the additional layers for the first model

We then compile the model using the "Adam" optimizer and "binary_crossentropy" loss function at the end. After using our dataset to train the model for 20 epochs, we were able to achieve an 86% validation accuracy, and obtain 88% accuracy while assessing its generalization capability on a test set.

```
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

r = model.fit(train_dataset,
         epochs = 20,
         validation_data = valid_dataset
         callbacks=[early_stopping])
```

Figure 4. 5: Compiling and fitting the first model

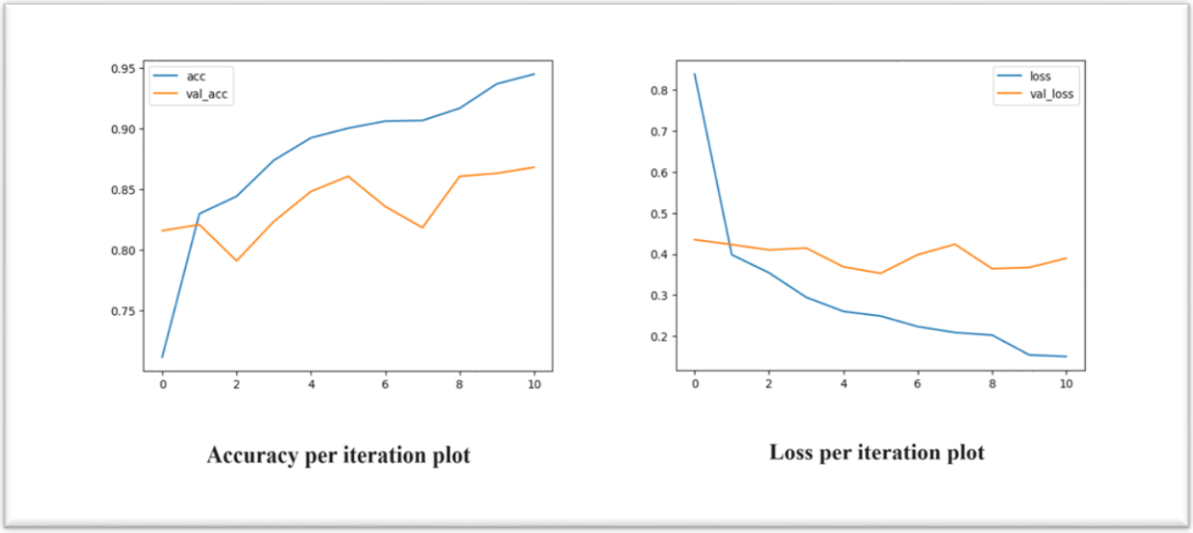

Figure 4. 6: Accuracy and Loss plots of our model for "the wildfire dataset "

It is worth to mention that tour results outperform the results reported by the researchers who gathered created the dataset itself. They achieved results by training the Mobile-Net model on this dataset yielded an accuracy of just 84%. As a result, we were able improve validation and test accuracy over their results and achieve **4%** improvement.

- In order to avoid overfitting, we also included an early stopping function, which is a regularization method used during machine learning model training. It stops the training process when the model's performance starts to decline, preventing the model from fitting noise in the data by keeping an eye on its performance on our validation dataset. When compared to other regularization techniques, this is relatively easy to implement, enhances the model's capacity for generalization, and saves time and computational resources by stopping training once further progress is doubtful.

```
from tensorflow.keras.callbacks import EarlyStopping

early_stopping = EarlyStopping(monitor='val_loss', patience=5,
                               verbose=1, restore_best_weights=True)
```

Figure 4. 7: Early stopping function

- Additionally, we included a prediction function, which is necessary to use the model in practical situations. With the use of this function, the model can employ its newly acquired knowledge to forecast fresh, unseen data. The model's interpretability, scalability, and modularity are all improved by breaking apart the prediction logic.

```
def predictImage(filename):
    img1 = image.load_img(filename,target_size=(224,224))
    plt.imshow(img1)
    Y = image.img_to_array(img1)
    X = np.expand_dims(Y,axis=0)
    val = model.predict(X)
    print(val)
    if val == 1:
        plt.xlabel("No Fire",fontsize=30)
    elif val == 0:
        plt.xlabel("Fire",fontsize=30)
```

Figure 4. 8: The prediction function

3.2 Second framework (A Fine-tuned late fusion model on Flame 2) [24] :

In the realm of computer vision, leveraging multiple modalities of images often leads to more robust and accurate models. Our approach demonstrates a method to fuse thermal and RGB images for wildfire detection. Thermal images capture temperature information, while RGB images contain color information. By combining both, we can potentially enhance the understanding and performance of models, particularly in scenarios like object detection or classification in low-light conditions.

We use this technique in our provided method, precisely, the late fusion technique. It involves combining the features extracted from different modalities at a late stage of the model architecture, so that the model would be able to learn and integrate information from both modalities at the end of the network architecture.

This approach allows the model to jointly learn representations that capture complementary aspects of the input data, potentially leading to more effective and robust performance.

In our architecture we followed these steps:

a. Features from RGB and thermal images are extracted separately by processing them through distinct sets of convolutional layers.
b. These features are then concatenated after being passed to the subsequent layers of the model.
c. The concatenated features are used to make predictions through a dense layer with a sigmoid activation function.

Our code utilizes nearly the same fine-tuned VGG16 model proposed for the wildfire dataset with only a few modifications for both RGB and thermal images. By leveraging pre-trained models, the base layers have already learned rich features from large-scale datasets like ImageNet. However, the last few layers of the base models are fine-tuned during training to adapt them to the specific task at hand.

Two functions « **create_base_RGB_layers** » and « **create_base_Thermal_layers** » are defined to create base layers. These functions freeze the weights of all layers except the last five, allowing for fine-tuning during training:

```python
# Function to create base layers for RGB images
def create_base_RGB_layers(input_img):
    vgg_RGB_model = VGG16(weights="imagenet", include_top=False, input_tensor=input_img)
    for layer in vgg_RGB_model.layers[:-5]:
        layer.trainable = False
    return vgg_RGB_model

# Function to create base layers for thermal images
def create_base_Thermal_layers(input_img):
    vgg_model_thermal = VGG16(weights="imagenet", include_top=False, input_tensor=input_img)
    for layer in vgg_model_thermal.layers[:-5]:
        layer.trainable = False
    return vgg_model_thermal
```

Figure 4. 9: Creating base Model

The RGB and thermal images are processed in parallel paths within the model architecture. We defined Separate sets of convolutional layers for each modality, allowing the model to learn modality-specific features independently. After processing, the feature maps from both paths are then concatenated creating a fused representation that combines information from both modalities. This fusion step facilitates the integration of complementary information, enhancing the model's overall understanding of the input data.
We followed these steps:

- We defined Input layers for both RGB and thermal images.
- Convolutional layers are defined separately for RGB and thermal images, followed by max-pooling and flattening operations.
- We then concatenated the outputs of these layers to merge the features learned from both modalities.
- A dense layer with a sigmoid activation function serves as the output layer, predicting a binary class.

```
# Define input layers for thermal and RGB images
thermal_input = Input(shape=(224, 224, 3), name='thermal_input')
rgb_input = Input(shape=(224, 224, 3), name='rgb_input')

# Create base layers for RGB and thermal images
RGB_base_layers = create_base_RGB_layers(rgb_input)
thermal_base_layers = create_base_Thermal_layers(thermal_input)

# Define the convolutional layers for RGB images
conv_rgb = Conv2D(32, (3, 3), activation='relu')(rgb_input)
pool_rgb = MaxPooling2D((2, 2))(conv_rgb)
conv_rgb = Conv2D(64, (3, 3), activation='relu')(pool_rgb)
pool_rgb = MaxPooling2D((2, 2))(conv_rgb)
flatten_rgb = Flatten()(pool_rgb)

# Define the convolutional layers for thermal images
conv_thermal = Conv2D(32, (3, 3), activation='relu')(thermal_input)
pool_thermal = MaxPooling2D((2, 2))(conv_thermal)
conv_thermal = Conv2D(64, (3, 3), activation='relu')(pool_thermal)
pool_thermal = MaxPooling2D((2, 2))(conv_thermal)
flatten_thermal = Flatten()(pool_thermal)

# Concatenate the flattened outputs
concatenated = Concatenate()([flatten_rgb, flatten_thermal])

# Output layer
output = Dense(1, activation='sigmoid')(concatenated)

# Create the model
model = Model(inputs=[rgb_input, thermal_input], outputs=output)
```

Figure 4. 10: Parallel Processing of Modalities

Next, we implemented a custom data generator to feed batches of data containing both RGB and thermal images into the model during training. This generator iterates over the data from both modalities simultaneously, ensuring that each batch contains paired samples from both sources along with their corresponding labels. This proposed approach enables the model to learn to correlate information between the two modalities effectively.

```python
# Define a custom data generator to yield data from both thermal and RGB generators
def combined_generator(rgb_generator, thermal_generator):
    while True:
        rgb_data, rgb_labels = next(rgb_generator)
        thermal_data, thermal_labels = next(thermal_generator)
        yield ([rgb_data, thermal_data], rgb_labels)  # Assuming both RGB and thermal labels are the same
```

Figure 4. 11: Custom Data Generator

Finally, during training, the model is optimized using the Adam optimizer and trained to minimize binary cross-entropy loss. The accuracy metric is used to evaluate the model's performance on both training and validation datasets. By training on a combined dataset containing samples from both modalities, the model learns to generalize well to unseen data and effectively utilize the complementary information provided by each modality.

```python
# Training the model using the combined generator
history = model.fit(
    combined_generator(train_rgb_generator, train_thermal_generator),
    epochs=10,
    steps_per_epoch=min(len(train_rgb_generator), len(train_thermal_generator)),
    validation_data=combined_generator(valid_rgb_generator, valid_thermal_generator),
    validation_steps=min(len(valid_rgb_generator), len(valid_thermal_generator)),
    verbose=1
)
```

Figure 4. 12: Model Training and Evaluation

- Upon training our model for wildfire detection, we achieved a remarkable validation accuracy of 99.99%. This achievement represents an improvement of **1%** compared to previous original findings of Flame 2 creators. Obtaining perfect validation accuracy underscores the efficacy of our approach according to the nature of this dataset, showcasing the model's ability to accurately detect wildfires from both RGB and thermal imagery.

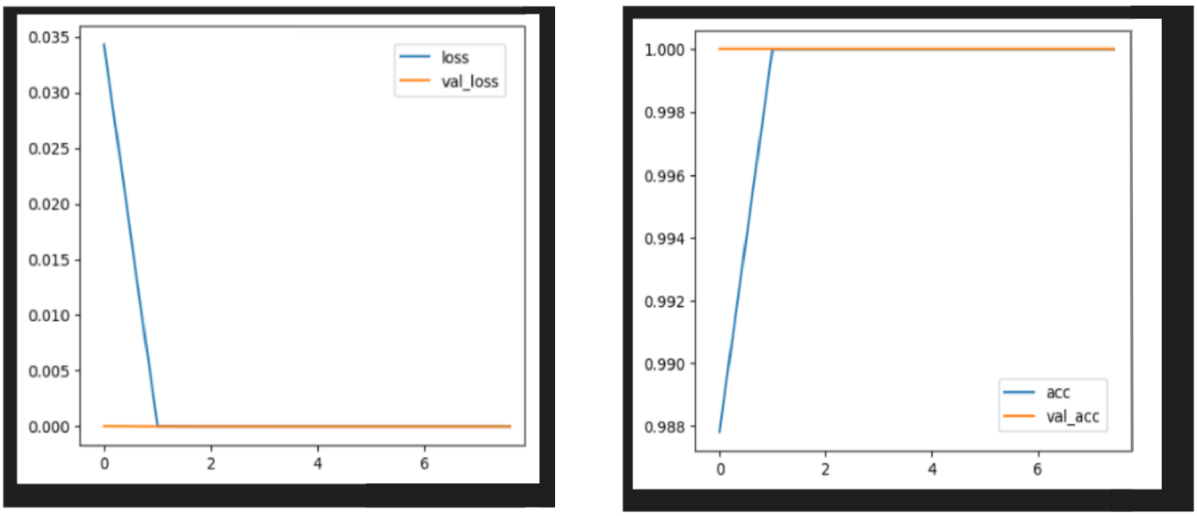

Figure 4. 13: loss and accuracy plots for fusion model

4 Comparaison between our approaches and the original ones:

Dataset	Source of Data	Fusion	Approach	Model	Result
The wildfire dataset	Kaggle	None	Original approach	Mobile-Net	**Accuracy=84%**
			Our Approach	VGG16	**Accuracy=88%**
Flame-2	IEEE	Thermal & RGB	Original approach	MobileNetV2	**Accuracy=99.82%**
				RasNet18	**Accuracy=99.5%**
			Our Approach	VGG16	**Accuracy=99.9%**

Table 4. 3: Final Approaches Comparaison

5 Why vgg16?

In the landscape of deep learning, few models have achieved the iconic status of VGG16. Developed by the Visual Geometry Group (VGG) at the University of Oxford, VGG16 has left an indelible mark on the field of computer vision since its introduction in 2014. Its simplicity, elegance, and remarkable performance have made it a cornerstone in various applications, ranging from image classification to feature extraction.

As suggested by the name, VGG16 is a convolutional neural network (CNN) with 16 weight layers. It is made up of three fully connected layers at the end, followed by a stack of convolutional layers and max-pooling layers. One of the unique features of VGG16 is its uniform architecture, in which the max-pooling layers have a 2x2 window with a stride of 2, while the convolutional layers all have a 3x3 filter size with a stride of 1. Its ease of comprehension and application is partly due to its simplicity.

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	224 x 224 x 3	-	-	-
1	2 X Convolution	64	224 x 224 x 64	3x3	1	relu
	Max Pooling	64	112 x 112 x 64	3x3	2	relu
3	2 X Convolution	128	112 x 112 x 128	3x3	1	relu
	Max Pooling	128	56 x 56 x 128	3x3	2	relu
5	2 X Convolution	256	56 x 56 x 256	3x3	1	relu
	Max Pooling	256	28 x 28 x 256	3x3	2	relu
7	3 X Convolution	512	28 x 28 x 512	3x3	1	relu
	Max Pooling	512	14 x 14 x 512	3x3	2	relu
10	3 X Convolution	512	14 x 14 x 512	3x3	1	relu
	Max Pooling	512	7 x 7 x 512	3x3	2	relu
13	FC	-	25088	-	-	relu
14	FC	-	4096	-	-	relu
15	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax

Table 4. 4: Brief Summary about VGG16

Moreover, since VGG16 was pre-trained on the ImageNet dataset, a sizable collection of more than 1.4 million of labeled images covering 1k categories, we selected VGG16 as the best option for transfer learning. This gives it access to a wide range of hierarchical features useful for various visual identification tasks. Its standard architecture and simplicity make it

simple to comprehend, apply, and modify, and the availability of pre-trained models in widely used frameworks makes it easier to integrate into new projects.

Likewise, the accuracy of VGG16 can differ based on a number of variables, including the particular dataset and task it is used for, as well as any adjustments or changes made to the model. VGG16, however, usually achieves good accuracy rates when applied as a pre-trained model for transfer learning on tasks like image categorization. With accuracy ranging from 92 to 94% on the validation set, VGG16 has proven to have excellent performance on benchmark datasets like ImageNet. VGG16 works well, though, more sophisticated and modern architectures have since been created that can accomplish even better accuracy on specific tasks. However, VGG16's robust performance, together with its ease of use and adaptability, keeps it a popular option in many real-world deep learning applications.

6 Why using drones for collecting dataset?

- Drones can be quickly deployed to survey large areas of land, especially in remote or inaccessible terrain where traditional methods may struggle to reach promptly. This rapid response capability allows for early detection of wildfires, enabling faster initiation of firefighting efforts and reduces the risk to human pilots who would otherwise need to fly low-altitude missions over dangerous terrain or near active fires.
- Drones using infrared sensors or even high-resolution cameras can take precise pictures of the terrain. They can now identify little flames or hotspots that satellites or people on the ground might overlook. In order to help with firefighting tactics, high-resolution photography also makes it possible to map fire perimeters and advance more precisely.
- Firefighters and emergency personnel can watch the situation develop by using real-time video feeds or data streams that drones can provide. Real-time monitoring facilitates prompt decision-making and resource allocation, enhancing the efficacy of firefighting operations and perhaps lowering casualties and property damage.
- Depending on the scope and severity of the wildfire, drones may be used singly or in groups. They may be adjusted to different operating requirements because of their scalability and flexibility, whether it is monitoring a small, localized fire or carrying out extensive aerial surveys over a large area.
- Drone systems' unique features of 3D mobility, low flight altitude, and fast and easy deployment make them a valuable tool for early detection and assessment of wildland fires, especially in remote forests that are not easily accessible by ground vehicles. [17]

7 The Used Programing Languages, Libraries and Tools:

7.1 Programming Languges :

We chose to utilize the Python programming language because of its widespread use in the artificial intelligence space, the vast array of libraries it supports, and its vibrant community.

- **Python** is a high-level, general-purpose, interpreted programming language. Python's design philosophy, which makes extensive use of indentation, places a strong emphasis on code readability.

Figure 4. 14: Python

7.2 Libraries:

- **TensorFlow**: is an open-source machine learning framework developed by Google. It provides a comprehensive ecosystem of tools, libraries, and resources for building and deploying machine learning models, particularly neural networks.

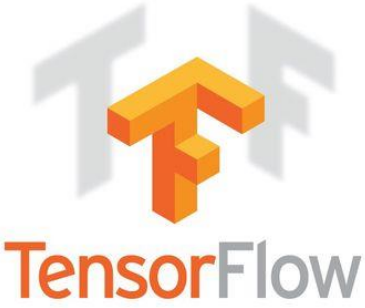

Figure 4. 15: Tensorflow

- **Keras** : is a high-level neural networks API written in Python, capable of running on top of TensorFlow, Theano, or Microsoft Cognitive Toolkit (CNTK). It provides a user-friendly interface for building and training deep learning models with minimal code, making it accessible to beginners and useful for rapid prototyping.

- **NumPy**: is a fundamental library for numerical computing in Python. It provides support for multidimensional arrays, along with a collection of mathematical functions to operate on these arrays efficiently.
- **cv2 (OpenCV)**: Open Source Computer Vision Library is a popular library for computer vision tasks. It offers a wide range of functions and algorithms for image processing, object detection, feature extraction, and more.
- **Matplotlib**: is a plotting library for Python that enables the creation of static, interactive, and animated visualizations. It provides a high-level interface for creating a variety of plots, including line plots, scatter plots, histograms, and more.

7.3 Tools and IDEs:

❖ **Kaggle IDE:**

Kaggle IDE is a web-based platform designed for data science and machine learning tasks, provides an interactive environment where users can write, execute, and share code in popular languages like Python and R. Additionally, it offers access to powerful computing resources such as GPUs and TPUs for running resource-intensive machine learning algorithms.
Kaggle provides a lot of benefits such as :

- **Rich Datasets:** Kaggle provides access to a wide range of high-quality datasets covering diverse domains, allowing users to explore real-world data and gain insights for research, analysis, and model building.
- **Accessible Platform:** Kaggle IDE provides a user-friendly web-based interface that allows users to access data science tools and resources from any device with internet connectivity, eliminating the need for local installations.
- **Powerful Computing Resources:** Kaggle offers access to high-performance computing resources, including GPUs and TPUs, enabling users to train complex machine learning models efficiently and at scale.
- **Collaborative Environment:** Kaggle facilitates collaboration among data scientists through features such as discussion forums, shared notebooks, and team projects, fostering knowledge sharing and peer learning.
- **Experimentation and Prototyping:** With its integrated code editor and execution environment, Kaggle IDE enables users to write, execute, and iterate on data analysis and machine learning models in a seamless workflow.

- **Visualization Tools:** Kaggle offers data visualization tools that allow users to create interactive charts, plots, and dashboards to explore and communicate insights from their data effectively.

❖ **VS Code:**

Visual Studio Code is a versatile and powerful code editor that supports various programming languages and frameworks, it is considered as a good choice for developing artificial intelligence (AI) models due to several key factors:

- **Support for AI Libraries and Frameworks:** VS Code provides support for various programming languages commonly used in AI development, such as Python, R, and Julia. It also offers extensions and plugins for popular AI libraries and frameworks, including TensorFlow, PyTorch, scikit-learn, and Keras.
- **Integration with Jupyter Notebooks:** VS Code supports Jupyter notebooks, which are widely used in AI development for interactive computing and prototyping.
- **Code Editing and Productivity Features:** VS Code offers a range of features to enhance productivity and code editing, including IntelliSense for code autocompletion, code navigation, refactoring tools, and built-in Git integration for version control.
- **Community and Ecosystem:** VS Code benefits from a vibrant community of developers and AI practitioners who contribute extensions, plugins, and resources to enhance its capabilities for AI development. This rich ecosystem provides access to tools, tutorials, and libraries to support various aspects of AI model development, from data preprocessing to model training and evaluation.

Figure 4. 16: Visual Studio Code

❖ **Flask Framework :**

Flask is a web framework, it's a Python module that lets you develop web applications easily. It's has a small and easy-to-extend core ; a microframework that doesn't include an ORM (Object Relational Manager) or such features.

Flask represents a collection of libraries and modules that enable web application developers to write applications without worrying about low-level details such as protocol, thread management, containing many cool features like url routing, template engine. [25]

Figure 4. 17 : Flask Logo

❖ **Figma Framework :**

Figma is a collaborative platform for editing vector graphics and prototyping. It allows us to design systems to facilitate the creation of websites and mobile applications. [26]

It is a solution for UI and UX designers and developers that offers many features such as:

- **Design:** with web design tools, automatic layout functions, plugins to reduce repetitive tasks.
- **Prototyping:** to test concepts very early in the design process.
- **Design System:** to create coherent designs with constantly updated libraries.
- **Collaborative:** to work with several people at the same time on a project, go back to a previous version if necessary or even display the work of a single collaborator for example.

Figure 4. 18 : Figma Logo

8 Web and Mobile Applications Prototypes:

Wildfires pose significant threats to both human lives and natural ecosystems, necessitating effective early detection and prevention strategies. In response to this pressing need, we proposed « Wildfire Detector », an innovative application for detecting wildfires using our cutting-edge deep learning algorithms.

We made two prototypes, a web application and a mobile one, each designed to harness the power of technology and community engagement in combating wildfires. By integrating real-time monitoring capabilities and proactive alert systems, "Wildfire Eye" aims to revolutionize wildfire management and mitigate the catastrophic consequences of these natural disasters.

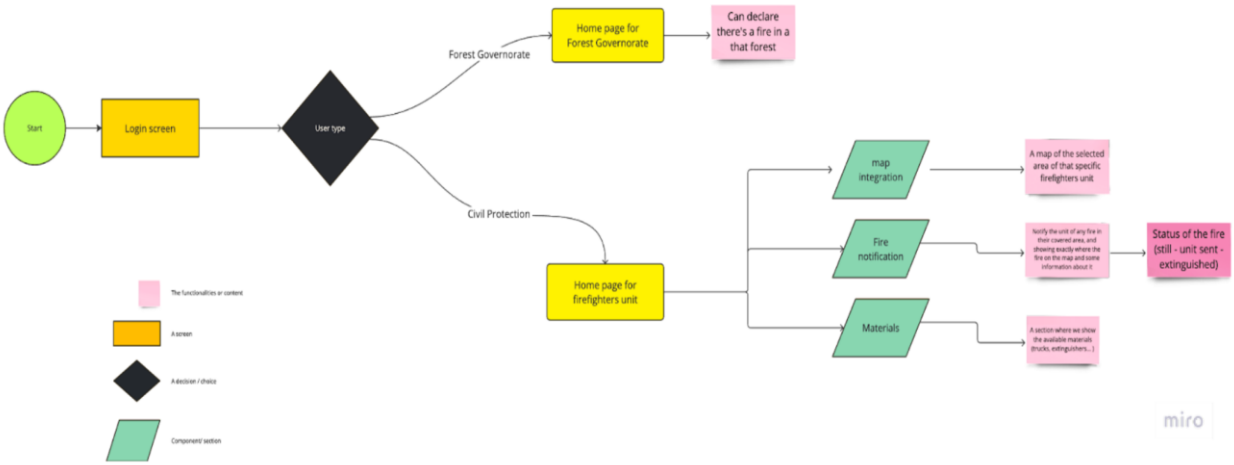

Figure 4.19 : Prototypes structure diagram

8.1 Web Application:

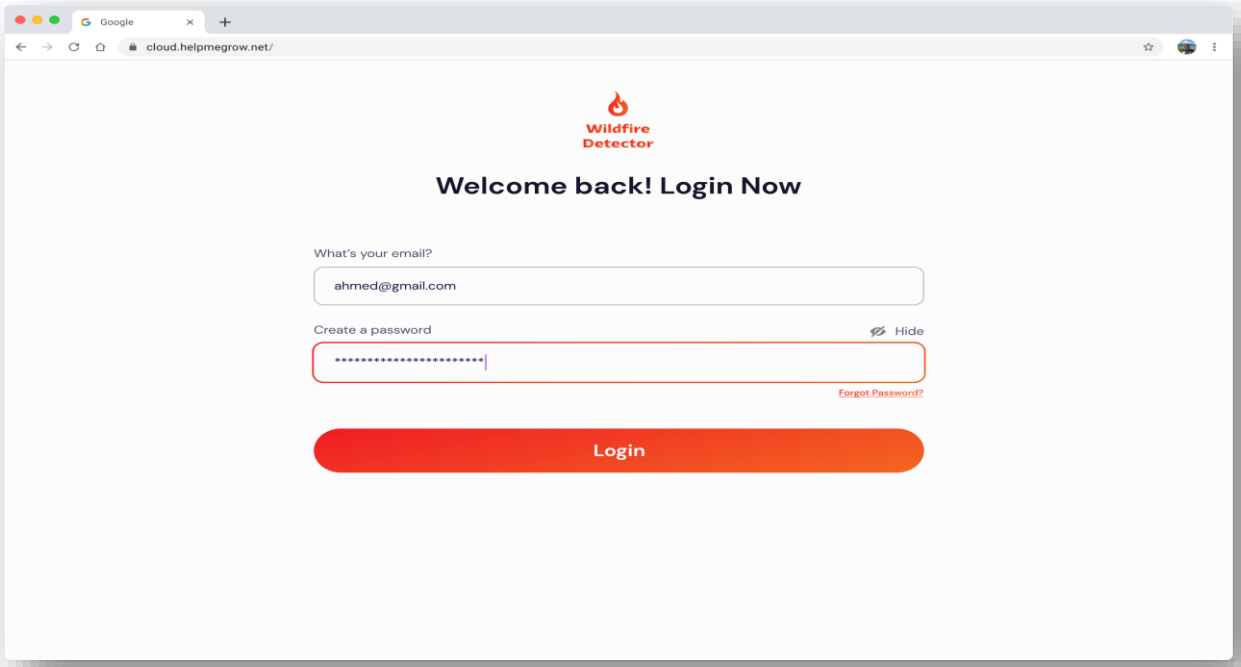

Figure 4. 20 : Login page

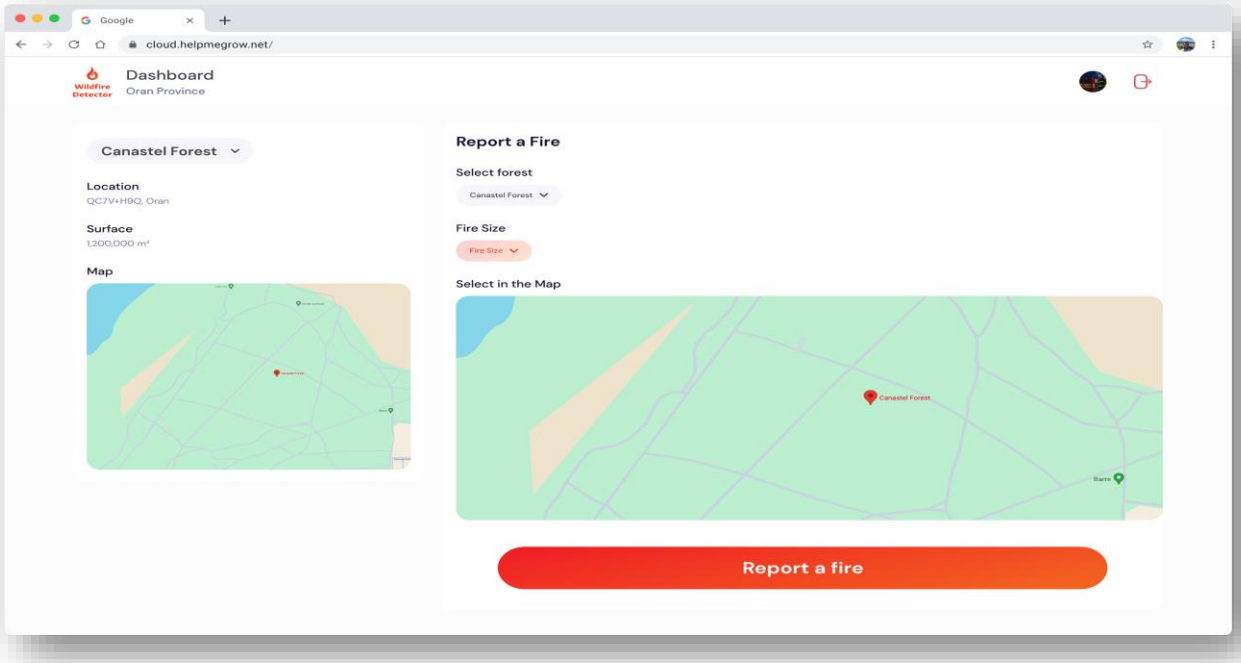

Figure 4. 21: Dashboard Page for Forest Governorate

In this page, the forest governorate can report a wildfire and select its location and details to send a notification to the firefighters units.

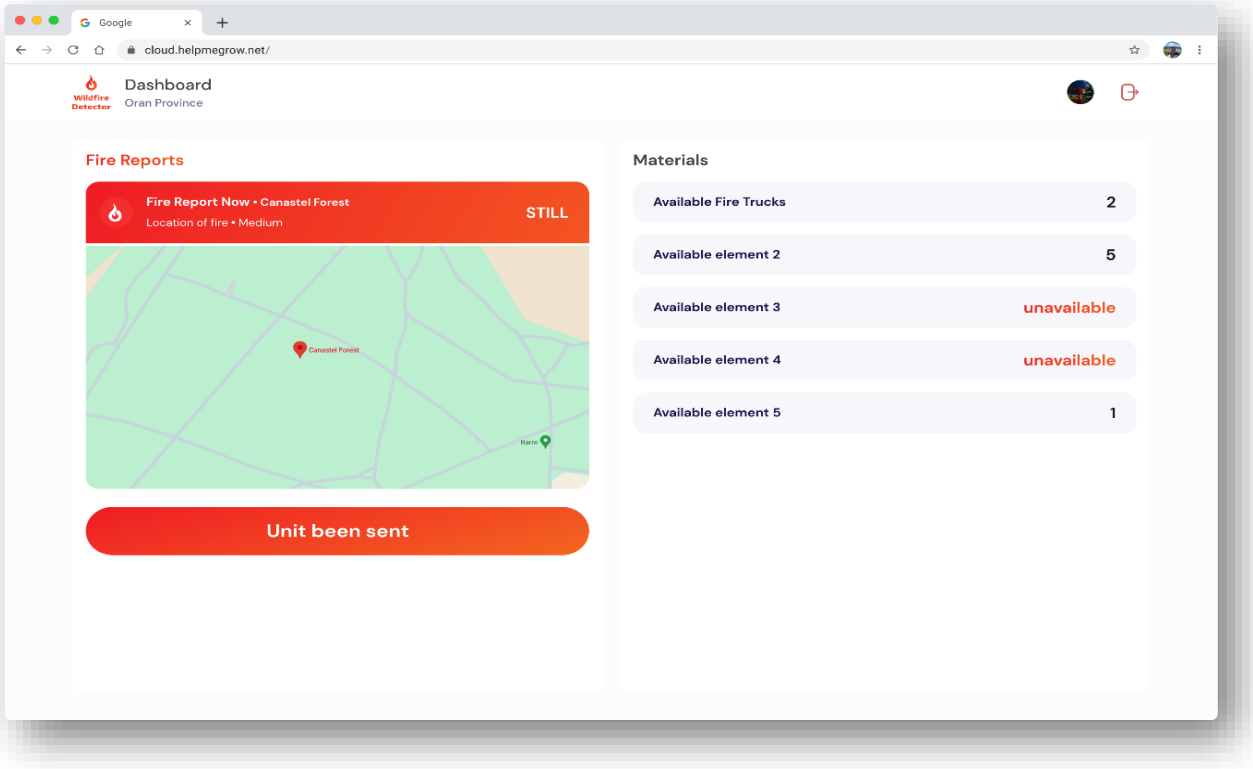

Figure 4. 22: Dashboard for Firefighters Unit

In this page the firefighters can be alerted, dispatch a unit, and oversee and manage the resources that are on hand.

8.2 Mobile Application:

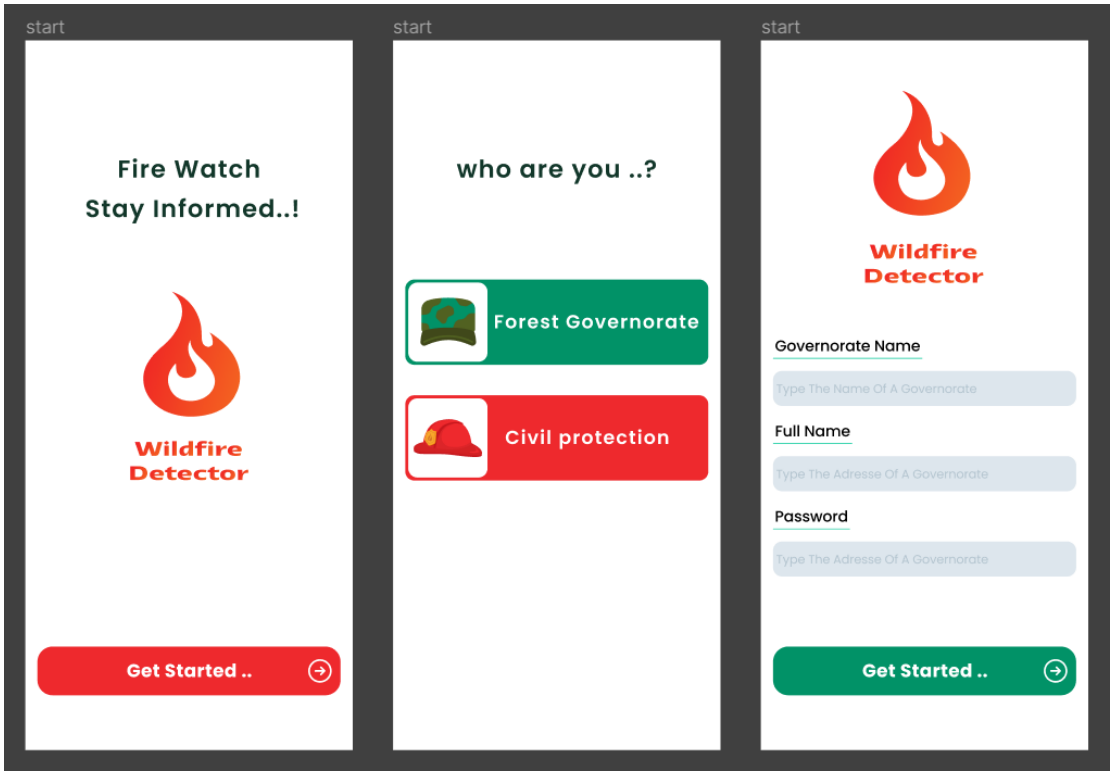

Figure 4. 23: Login Process Page

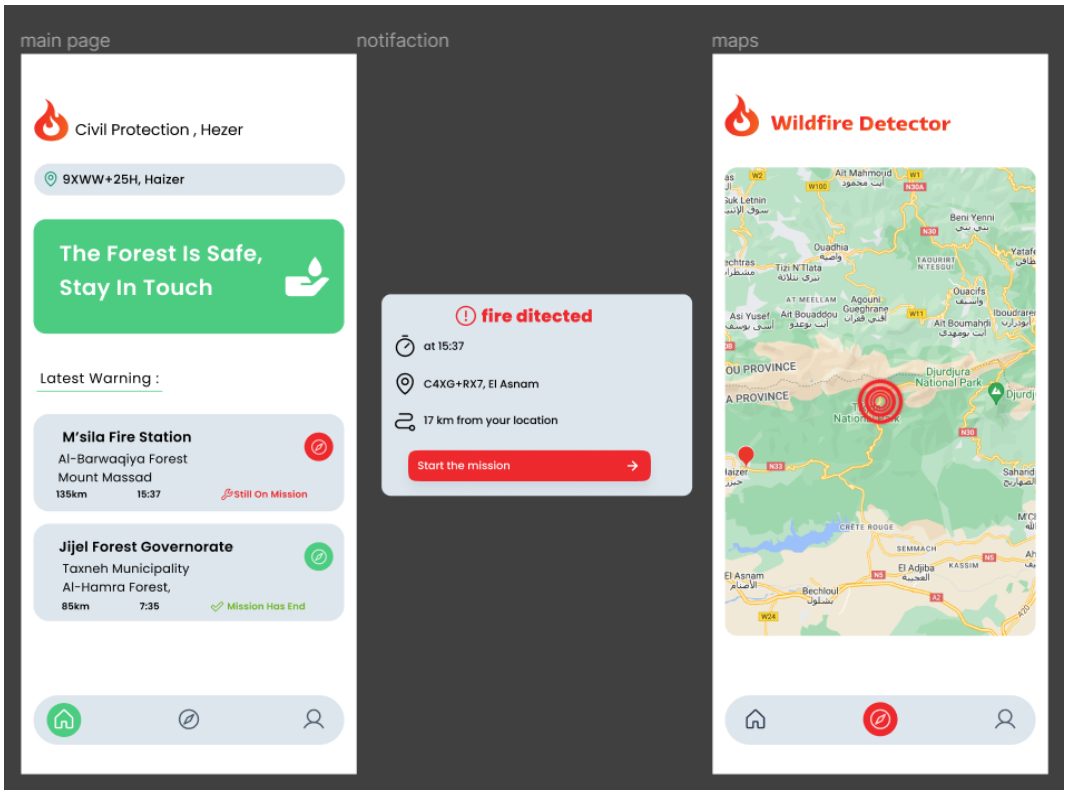

Figure 4. 24: Dashboard of Mobile app

In this dashboard, the firefighters can be notified about the wildfire and its details.

9 Conclusion:

In conclusion, our approaches in wildfire detection mark a significant advancement in leveraging cutting-edge technology to address critical environmental challenges. We've demonstrated the effectiveness of utilizing pre-trained VGG16 model, originally trained on the ImageNet dataset, to adapt to the task of wildfire detection through the fine-tuning technique. The later, has been efficiently applied on two different datasets, with nearly the same robust model. While the first model was applied on dataset with confounding elements, the other model, with late fusion, has been applied on the first existing dataset with both RGB and Thermal modalities.

Looking ahead, our work sets the stage for further advancements in wildfire detection technology, with potential applications ranging from automated monitoring systems to predictive analytics for wildfire risk assessment. By continuing to innovate and refine our methodologies, we can make significant strides towards mitigating the impact of wildfires on communities and ecosystems, ultimately contributing to a more sustainable and resilient future.

General Conclusion

In conclusion, the integration of computer vision technology in wildfire detection represents a significant advancement in mitigating the devastating impact of wildfires on ecosystems, communities, and infrastructure. Through the development of innovative predictive models and early warning systems, computer vision offers a powerful tool for enhancing the accuracy, efficiency, and timeliness of wildfire detection efforts.

Our approach, centered on the development of a novel predictive model using transfer learning techniques, has demonstrated the efficacy of adapting pre-trained deep learning architectures to the task of real-time wildfire detection. Through rigorous experimentation and validation, we have validated the effectiveness of our approach in detecting wildfires with high accuracy. Our contribution lies in the advancement of wildfire detection methodologies, specifically in the realm of computer vision, by providing stakeholders with timely and accurate information to facilitate prompt response efforts.

As we continue to refine and optimize these technologies, there is tremendous potential to further improve wildfire detection capabilities and enhance our ability to mitigate the impact of wildfires on ecosystems and communities. Through interdisciplinary collaboration, research, and innovation, we can harness the power of computer vision to safeguard lives, property, and the environment from the destructive forces of wildfires.

References

[1] National Geographic Society, "Controlled Burning | National Geographic Society," *education.nationalgeographic.org*, May 20, 2022. https://education.nationalgeographic.org/resource/controlled-burning/

[2] Bennett, M., S.A. Fitzgerald, B. Parker, M. Main, A. Perleberg, C.C. Schnepf, and R. Mahoney. 2010. Reducing Fire Risk on Your Forest Property. PNW 618 : 40 p. | https://www.nwfirescience.org/sites/default/files/publications/Types%20of%20Fire.pdf

[3] IBM, "Computer Vision," *IBM*, 2019. https://www.ibm.com/topics/computer-vision

[4] Analytics Vidhya, "Build an Image Classification Model in just 10 Minutes!," *Analytics Vidhya*, Jan. 10, 2019. https://www.analyticsvidhya.com/blog/2019/01/build-image-classification-model-10-minutes/

[5] P. Sharma, "A Step-by-Step Introduction to the Basic Object Detection Algorithms (Part 1)," *Analytics Vidhya*, Oct. 11, 2018. https://www.analyticsvidhya.com/blog/2018/10/a-step-by-step-introduction-to-the-basic-object-detection-algorithms-part-1/

[6] Analytics Vidhya, "Step-by-Step Tutorial on Image Segmentation Techniques in Python," *Analytics Vidhya*, May 06, 2019. https://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques-python/

[7] Wikipedia Contributors, "Machine learning," *Wikipedia*, Apr. 29, 2019. https://en.wikipedia.org/wiki/Machine_learning

[8] Wikipedia Contributors, "Deep learning," *Wikipedia*, May 10, 2019. https://en.wikipedia.org/wiki/Deep_learning

[9] *Chapter 5 Convolutional neural networks and their applications in NLP | Modern Approaches in Natural Language Processing*. Available: https://slds-lmu.github.io/seminar_nlp_ss20/convolutional-neural-networks-and-their-applications-in-nlp.html

[10] J. Robert, "Transfer Learning : Qu'est-ce que c'est ?" *Formation Data Science | DataScientest.com*, Jul. 21, 2020. https://datascientest.com/transfer-learning (accessed Jun. 03, 2024).

[11] Lee, W.; Kim, S.; Lee, Y.T.; Lee, H.W.; Choi, M. Deep neural networks for wildfire detection with unmanned aerial vehicle. In Proceedings of the IEEE International Conference on Consumer Electronics (ICCE), Taipei, Taiwan, 12–14 June 2017; pp. 252–253.

[12] Shamsoshoara, A.; Afghah, F.; Razi, A.; Zheng, L.; Fulé, P.Z.; Blasch, E. Aerial imagery pile burn detection using deep learning: The FLAME dataset. Comput. Netw. 2021, 193, 108001. [CrossRef]

[13] Zhao, Y.; Ma, J.; Li, X.; Zhang, J. Saliency Detection and Deep Learning-Based Wildfire Identification in UAV Imagery. Sensors2018, 18, 712. [CrossRef]

[14] Treneska, S.; Stojkoska, B.R. Wildfire detection from UAV collected images using transfer learning. In Proceedings of the 18th International Conference on Informatics and Information Technologies, Skopje, North Macedonia, 6–7 May 2021.

[15] Srinivas, K.; Dua, M. Fog Computing and Deep CNN Based Efficient Approach to Early Forest Fire Detection with Unmanned Aerial Vehicles. In Proceedings of the International Conference on Inventive Computation Technologies, Coimbatore, India, 26–28 February 2020; pp. 646–652.

[16] Forests 2023, 14(9), 1697 | https://doi.org/10.3390/f14091697 | Universitat Politècnica de Catalunya, spain.

[17] Article of Wildland Fire Detection and Monitoring Using a Drone-Collected RGB/IR Image Dataset IEEE | USA.

[18] Natural Hazards Research journal | Volume 2, Issue 3, September 2022, Pages 154-165

[19] *Remote Sens.* 2021, *13*(19), 3985| https://doi.org/10.3390/rs13193985 | Sungkyunkwan University, Suwon 16419, Korea

[20] Article of A Machine Learning Data Set to Predict Wildfire Spreading from Remote-Sensing Data, arXiv:2112.02447v2 [cs.CV] 2 Mar 2022 | submitted to IEEE.

[21] "The wildfire dataset," www.kaggle.com. https://www.kaggle.com/datasets/elmadafri/the wildfire-dataset/data (accessed Jun. 03, 2024).

[22] " FLAME 2: FIRE DETECTION AND MODELING : AERIAL MULTI-SPECTRAL IMAGE DATASET " | Bryce Hopkins (Clemson University) | 16 february 2023

[23]" WILDFIRE Detection Using VGG16," kaggle.com. https://www.kaggle.com/code/manelramdani/wildfire-detection-using-transfert-learning-vg (accessed Jun. 03, 2024).

[24] "Fusion model (thermal&RGB) Using VGG16," *kaggle.com*. https://www.kaggle.com/code/manelramdani/fusion-model-thermal-rgb-using-vgg16 (accessed Jun. 03, 2024).

[25] "What is Flask Python - Python Tutorial," *pythonbasics.org*, 2021. https://pythonbasics.org/what-is-flask-python/

[26] "Figma : un outil de prototypage et de design collaboratif," *BDM | Tools*. https://www.blogdumoderateur.com/tools/figma/

[27] Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention | Munich, Germany | 5–9 October 2015 | Springer International Publishing: Cham, Switzerland, 2015; pp. 234–241.

[28] Article Deep Learning and Transformer Approaches for UAV-Based Wildfire Detection and Segmentation, Sensors 2022, 22, 1977 | https://doi.org/10.3390/s22051977.

[29] Wildfire Classification Using Infrared Unmanned Aerial Vehicle Data with Convolutional Neural Networks" 2023 International Conference on Smart Applications, Communications and Networking (SmartNets) | 979-8-3503-0252-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/SmartNets58706.2023.10215891.

[30] Jiao, Z.; Zhang, Y.; Xin, J.; Mu, L.; Yi, Y.; Liu, H.; Liu, D. A Deep Learning Based Forest Fire Detection Approach Using UAV and YOLOv3. In Proceedings of the 1st International Conference on Industrial Artificial Intelligence (IAI), Shenyang, China,23–27 July 2019; pp. 1–5.

[31] Alexandrov, D.; Pertseva, E.; Berman, I.; Pantiukhin, I.; Kapitonov, A. Analysis of Machine Learning Methods for Wildfire Security Monitoring with Unmanned Aerial Vehicles. In Proceedings of the 24th Conference of Open Innovations Association (FRUCT), Moscow, Russia, 8–12 April 2019; pp. 3–9.

[32] Jiao, Z.; Zhang, Y.; Mu, L.; Xin, J.; Jiao, S.; Liu, H.; Liu, D. A YOLOv3-based Learning Strategy for Real-time UAV-based Forest Fire Detection. In Proceedings of the Chinese Control and Decision Conference (CCDC), Hefei, China, 22–24 August 2020; pp. 4963–4967.

[33] Tang, Z.; Liu, X.; Chen, H.; Hupy, J.; Yang, B. Deep Learning Based Wildfire Event Object Detection from 4K Aerial Images Acquired by UAS. AI 2020, 1, 10. [CrossRef]

[34] Barmpoutis, P.; Stathaki, T.; Dimitropoulos, K.; Grammalidis, N. Early Fire Detection Based on Aerial 360-Degree Sensors, Deep Convolution Neural Networks and Exploitation of Fire Dynamic Textures. Remote Sens. 2020, 12, 3177. [CrossRef]

Abstract

This work aims to enhance early and real-time wildfire detection utilizing computer vision and transfer learning techniques, specifically employing the VGG16 model. We developed two models, the first using only RGB images, achieving an accuracy of 88%, representing a 4% improvement over previously existing models. The second model utilize fusion technique, integrates both RGB and thermal images, attaining a remarkable 99% accuracy. Additionally, prototypes for future web and mobile applications have been created to facilitate real-time wildfire detection and response.

Key words: Wildfire Detection, Transfert Learning, Fusion technique, VGG16 Model.

الملخص

يهدف هذا العمل إلى تعزيز الكشف المبكر وفي الوقت الفعلي عن حرائق الغابات باستخدام رؤية الحاسوب وتقنيات تعلم النقل، وتحديدًا باستخدام نموذج VGG16. قمنا بتطوير نموذجين، الأول يستخدم صور RGB فقط، حقق دقة 88%، وهو ما يمثل تحسنًا بنسبة 4% عن النماذج الموجودة سابقا. يستخدم النموذج الثاني تقنية الدمج، يدمج كلاً من صور RGB والصور الحرارية، محققا دقة ملحوظة تبلغ 99%. بالإضافة إلى ذلك، تم إنشاء نماذج أولية لتطبيقات الويب والهواتف المحمولة المستقبلية لتسهيل اكتشاف حرائق الغابات والاستجابة لها في الوقت الفعلي.

الكلمات المفتاحية: اكتشاف حرائق الغابات، تقنيات تعلم النقل، تقنية الدمج، نموذج VGG16 .

Résumé

Ce travail vise à améliorer la détection précoce et en temps réel des feux de forêt en utilisant des techniques de vision par ordinateur et d'apprentissage par transfert, en particulier le modèle VGG16. Nous avons développé deux modèles, le premier utilisant uniquement des images RVB, atteignant une précision de 88%, soit une amélioration de 4% par rapport aux modèles déjà existants. Le deuxième modèle utilise une technique de fusion, intègre des images RVB et thermiques, atteignant une précision remarquable de 99%. De plus, des prototypes pour des applications web et mobiles futures ont été créés pour faciliter la détection et la réponse en temps réel aux feux de forêt.

Mots Clés : détection des incendies de forêt, Apprentissage par transfert, Technique de fusion, Modèle VGG16.