Software Implementation of Fire and Smoke Detection using Convolutional Neural Networks(CNN) and ResNet50

21BCE179 Kaushal Parmar, 21BCE186 Arya Patel, 21BCE206 Man Patel

***Abstract*—Fire incidents represent a significant global challenge due to their destructive impact on human lives, and economies. This paper introduces an innovative approach to address the challenges posed by this fire incidents, focusing the detection of fire and smoke in an closed environment. Fire has vast impact on the The increasing frequency and intensity of fire incidents, driven by factors such as human error and vantilation problems, emphasizes the need for efficient early detection and containment strategies. We study existing fire incident images data and apply required image preprocessing steps on the dataset. Our study leverages software engineering remote sensing technologies along with deep learning (DL) methodologies, focusing on Convolutional Neural Networks (CNNs) and ResNet50, which are used for automatic feature extraction and classification capabilities. Our research aims to significantly enhance the effectiveness and accuracy of fire incidents and smoke detection by leveraging these advanced methods. The results of our study demonstrate the exceptional performance of the CNN model, with an impressive classification accuracy of 96.37 percent. We do comparative analysis of CNN and ResNet50 model using different metrics. This research contributes to the advancement of early fire incidents detection, with the potential to mitigate the extensive ecological, economic, and societal impacts of fire incidents.**

***Index Terms*—Fire incidents, Remote Sensing, Deep Learning, CNN, ResNet50, QGIS**

I. Introduction

Fire incidents are a serious problem because they affect ecosystems, risk human life, and cause huge financial losses. These catastrophes have significant ecological and environmental ramifications, including habitat destruction, elevated carbon emissions, and the possibility of biodiversity loss, regardless of whether they are caused by human activity or natural causes [1]. Because smoke and fire are hard to reach and there are a lot of things that can catch fire in wooded areas, it can be hard to put them out quickly and effectively. In addition, effective early discovery and containment methods are needed because they are happening more often and with more force because of climate change and human activities [2]. The effects of fire incidents on our environment, economy, and society are extensive and severe. They put people's lives in danger in the impacted areas and cause huge losses in terms of the economy and environment.

Millions of hectares of forests, large number of indoor fire accidents cause vast impact on the environment by these fires every year, which increases carbon dioxide levels and oxygen loss, both of which can worsen climate change [3]. Furthermore, fire incidents severely harm wildlife by destroying their habitats and reducing biodiversity. Fire incidents degrade the air quality, causing respiratory disorders and other health problems in both people and animals. These emissions also include smoke and other pollutants. Additionally, they degrade land and cause erosion over time due to their long-term effects on soil quality. In addition to causing significant emissions of carbon dioxide into the atmosphere, these fires exacerbate climate change [4]. Their effects go beyond financial losses to include devastation of infrastructure and timber resources [5].

Given that fire incidents have the potential to cause high damage to large area of land and result in fatalities, it is necessary that they are detected promptly in order to minimise the consequences. A number of factors, including prolonged droughts, high temperatures, and human negligence, are linked to the rise in forest fire incidents [6]. Forest fires on a large scale can do a lot of damage, which shows how important it is to plan and handle them well to avoid disaster. The adoption of DL and traditional machine learning techniques has been prompted by the shortcomings of traditional forest fire detection systems, which are frequently insufficient in wide-open outdoor environments [7].

CNN is a potent model for automatically extracting features and classifying them. It has also done better than traditional machine learning methods at finding trends and outliers. Finding forest fires and smoke in new ways is the focus of this study, which stresses accuracy and early warning systems. Deep learning, hybrid techniques, and advanced object recognition are all combined in the suggested model. It uses the power of the ResNet50 and CNN models to rate the performance of the system. CNN achieves an impressive 96.32 percent accuracy rate in classification jobs. The system is great at more than just classifying; it's also great at analyzing trends and finding outliers, which makes it much more accurate at finding forest fire incidents. This study is a big step toward making fire warning systems that are more accurate, timely, and effective. This could have effects on protecting the environment and keeping people safe. [8] [9].

We investigate a hybrid system that blends deep learning and remote sensing technology in order to accomplish these objectives [10]. In addition, we present a collaborative region detection and grading framework that makes use of a lightweight CNN and weakly supervised fine segmentation to tackle the problem of fire incidents detection and grading. Our goal is to increase the accuracy and effectiveness of the crucial job of detecting forest fires and smoke by utilising these methods.

II. Related Works

Various techniques have been used in the field of fire and smoke detection research, ranging from manual techniques that focus on the motion and color characteristics of flames to deep learning algorithms that can automatically extract features. Past research investigates the chromatic and dynamic characteristics of smoke and fire, sometimes relying on inconsistent color spaces and flimsy logic to discern flames from comparable objects [11]. FireNet is a lightweight neural network designed to cater to real-time Internet of Things applications, and it can efficiently run on inexpensive embedded devices such as the Raspberry Pi. Its purpose is to overcome the limitations of deep learning models like VGG16, ResNet50, and AlexNet [11]. FireNet is built from the ground up to address the limitations of the existing infrastructure, and it has proven to be successful on a range of datasets.

Conventional techniques for detecting fire and smoke depend on manually designed characteristics and rule-based algorithms. It is clear that deep learning methods, and convolutional neural networks, in particular, have great potential for solving visual recognition difficulties. Within this field, researchers have looked in a variety of directions. For the purpose of detecting fire in forest photos, Zhang et al suggested a technique that combines global and location-specific classified into two linked deep CNNs. Simultaneously, Tao and colleagues suggested a specific deep CNN model for smoke identification [12].

Bugarovich et al. presented a computer vision-based technique in their research paper that measures flame dimensions using augmented reality based on geographic information systems. The deep CNN model referenced in the study shows the capacity to concurrently classify images of both fire and smoke [13].

Previous studies on fire detection have made use of convolutional neural networks(CNNs).The research is reviewed critically in this article, which also cautions against the use of balanced datasets for performance assessments, which may not accurately reflect real-world settings and lead to biases [14]. The usage of Blender simulations and the AlexNet architecture to generate a custom fire dataset. Additionally, a modified VGG16 with four completely linked layers is introduced in this study, exhibiting enhanced classification accuracy. The recognition of deep CNN architectures, superior performance on the ImageNet dataset over AlexNet—specifically, VGG16 and Resnet50—further bolsters the case for their use in the suggested research [14].

The research under consideration introduces a novel method for smoke and fire detection with a 99.05fire and smoke by combining deep learning methodologies with HSV color characteristics. [15]. The paper presents a detailed comparative analysis of the proposed Vision Transformer model and other popular networks, including ResNet, InceptionV3, MobileNet, and Faster R-CNN [15].

The study conducted in [16] utilizes Support Vector Machines (SVM) and Artificial Neural Networks (ANN) to measure the level of fire susceptibility in Indian forests. This study is among the first in list to predict wildfire susceptibility in India using an integrated approach, utilising MODIS fire hotspot data from 2001 to 2020. The study determines the best ML techniques for creating a fire prediction model and emphasizes the importance of basic characteristics pertaining to topography, climate, and forests. It also emphasises how intricately forests, climatic, and topographic factors interact to increase the frequency and intensity of fires. The study recommends effectiveness models that assume dynamic relationships between parameters to accurately predict fire susceptibility with accuracy.

[17] concerns the application of CNN as the basis for in the formulation of fire detection algorithms designed to identify forest fires in satellite imagery. Many classification architectures have been explored, as fully connected and softmax layers, convolution and max pooling layers, and ROI methods. Due to the limitations of traditional fire control systems in monitoring large areas and open spaces, algorithms such as AdaBoost and Local Binary Pattern are used. While methods like the Restricted Boltzman Machine and Deep Belief Network have been used for simultaneous fire and non-fire region extraction, deep learning architectures like GoogleNet and SqueezeNet have been used for early flame detection. For real-time fire incidents detection, faster R-CNN, YOLO, and SSD architectures have been compared.

III. System model and problem formulation

In this section we define our problem statement, formulate it so that a model can be implemented on it to get desired output. In this research project, our main goal is to apply state-of-the-art deep learning techniques to create a reliable and effective system for identifying fire incidents in photos from data provided. Fire incidents are dangerous for people and their property in addition to being bad for the environment. In recent times various hardware sensors are used to detect the fire at homes, malls, etc. These sensor detect the temperature and gives the signal if there is fire occured or not. Our proposed system architecture uses surveillance cameras to detect the fire which is not only cost effective but it has few more advantages over hardware mechanisms like the camera will used for the security reasons as well as for fire detection. Because it is necessary for efficient reaction and mitigation activities, the importance of prompt detection cannot be emphasized. Creating a system that can automatically detect and map fire and smoke from visual data is the main goal of our research in order to provide precise and timely response.

We use a rich and diversified dataset from DataCluster Labs to accomplish this challenging task. This dataset mainly focuses on Fire and Smoke images captured and crowdsourced from over 400+ urban and rural areas, where each image is manually reviewed and verified by computer vision professionals at Datacluster Labs. It contains a plethora of cartographic information that has been gathered from many sources, such as aerial photos, field surveys and historical records. We do a thorough preprocessing phase on this dataset before using it for model training. The photos will improve in quality and utility throughout this stage. Resizing, normalizing, and augmenting images are all examples of preprocessing tasks. These metrics are critical because they help to normalize the input data and make it suitable for the subsequent deep learning models, both of which improve the models' overall performance.

Fig. 1: Samples from the data

In the context of fire identification in video images, Convolutional Neural Networks (CNNs) are utilized as the primary approach, with a specific focus on the ResNet-50 architecture. Convolutional Neural Networks (CNNs) have received recognition for their remarkable ability to excel in tasks related to picture classification. This success can be linked to their inherent capability to independently acquire hierarchical features from image input. Within the realm of CNN architectures, ResNet-50 emerges as a prominent and well regarded model, renowned for its remarkable depth and the integration of skip connections. The aforementioned characteristics confer upon it the capacity to effectively capture delicate details inside images while simultaneously preserving robust gradient flows during the training procedure. ResNet-50 is a very suitable option for our research context because to its ability to accurately identify minor image characteristics, which is crucial for effective fire diagnosis.
The CNN and ResNet based fire detection process can be mathematically represented as:

$$P(FireDetected|I) = f_{CNN}(I) \tag{1}$$

$$P(FireDetected|I) = f_{Resnet}(I) \tag{2}$$

Where P(Fire Detected—I) signifies the probability of detecting a fire in input image I, and f represents the DL based detection function.

Utilizing image analysis methods, our main objective is to build a fire early warning system. Equations and functions of mathematics play a critical role in real-time picture data and sensor networks, which this system will leverage. We must create an algorithm that, in a mathematical setting, applies picture categorization using mathematical operations, continuously checks incoming image streams, and raises alarms when necessary.This process can be represented mathematically as follows:

Let S(t) represent an incoming image stream at time t, and A(S(t)) denote the early warning system's alert function:

$$A(S(t)) = \begin{cases} 1 & \text{, if fire or smoke is detected in the stream S(t)} \\ 0 & \text{, otherwise} \end{cases}$$

The objective of this study is to exploit the feature-extraction capabilities of Convolutional Neural Networks (CNNs) in order to identify patterns related to fires, including flames, smoke plumes, and other pertinent indicators that can be observed in satellite data. One crucial aspect of our study involves doing a comparative analysis of traditional Convolutional Neural Network (CNN) models and the ResNet-50 architecture. This comparison investigation will provide vital insights into the performance of both methods in the unique setting of fire and smoke detection. Our research project will go into the finer points of the model architectures, explain the subtleties of our training techniques, and clarify the evaluation metrics that support our estimation of the models' efficacy in the next parts. In order to evaluate the overall resilience and dependability of our fire detection system, we will also make use of advanced visualizations and perform a thorough analysis of the data. After that, we will have a conversation about the various uses for our system, from timely notifications to alert the owners using the software and further guidance to tackle the situation. The importance and urgency of this research are highlighted by the socioeconomic and environmental significance of our work.

IV. The Proposed Scheme

In this section of paper we describe the software engineering procedures needed to finish our study. We concentrate on the steps involved in acquiring data and putting the model into practice. Our research includes preprocessing imagery data, picture analysis, region characterization, application of DL techniques CNN and ResNet50. We also discuss about early detection of fire incidents and mitigation strategy development.

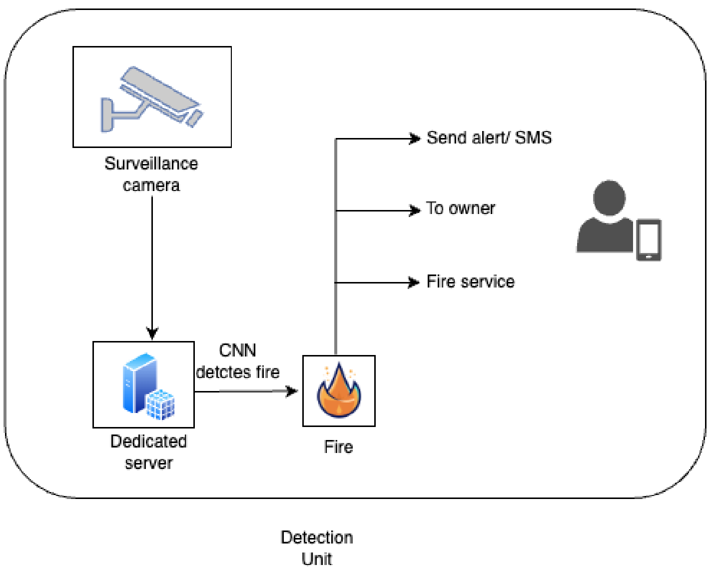


Fig. 2: Working of fire detection unit

The system's architecture comprises both hardware and software components, allowing users to configure it based on the number of units they possess. To enhance fire detection capabilities, a surveillance camera unit is integrated into the hardware system. This unit continuously monitors the premises and transmits the video feed to a centralized server for fire incident detection and alert notifications. In the event of fire detection, the property owner receives an alert, and simultaneously, a notification is dispatched to the fire service. The notification to the fire and rescue crew includes regional mapping and navigational instructions, facilitated by the integration of the Google Map API. The overall depiction of the system architecture is illustrated in Fig. 2. The detailed working of the Fig. 2 is depicted with the help of flow diagram Fig. 3 for fire detection software. Fig. 3 describes three potential outputs namely fires,potential fires and no fire associated with probabilities 95 precent, 75-95 and less than 75 respectively.

The system architecture comprises of an mobile application. Given the convenience and portability of mobile phones, having a mobile application to supplement traditional sirens becomes crucial. This approach ensures that in the event of a fire incident, individuals, including both occupants and property owners, can

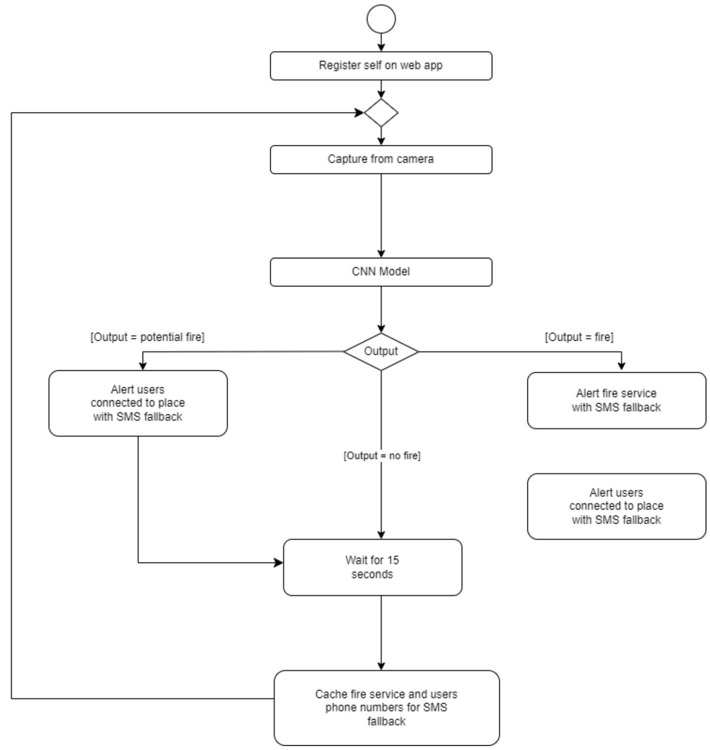


Fig. 3: Flow diagram of working of detection system

receive timely notifications regardless of their location. The user needs to add their surveillance camera to the mobile application so that it can send alerts if any fire is noticed. Fig. 4 shows the use case diagram for the mobile application that will be for the user and Fig. 5 shows for the manufacturers.

As a first step in improving the model's capacity to generalize, the algorithm preprocesses and augments the data. By performing different picture transformations, the ImageDataGenerator is used to enhance the dataset. Rescaling pixel values to a range of 0 to 1 is one of these transformations, along with shifts, shear, zooming, and flipping horizontally and vertically. With the use of data augmentation, training data can become more diverse and the model is more equipped to manage appearance variations in images, including adjustments to lighting, perspective, and orientation. For reliable picture classification, this is a crucial step. Preprocessed picture batches are produced by the train and validation data generators. From the dataset directory, these generators read and prepare the photographs. Images undergo resizing to a common 256 x 256 pixel size and RGB color mode conversion throughout this process. When training and validating the model, the consistent format of the input data guarantees that the model receives photos of the same size.

Fig. 7 shows working principle of CNN model on an image dataset. Tensorflow's Keras is used to organize the CNN model into a sequential model. Layers can be added consecutively using the sequential paradigm, beginning with the input and

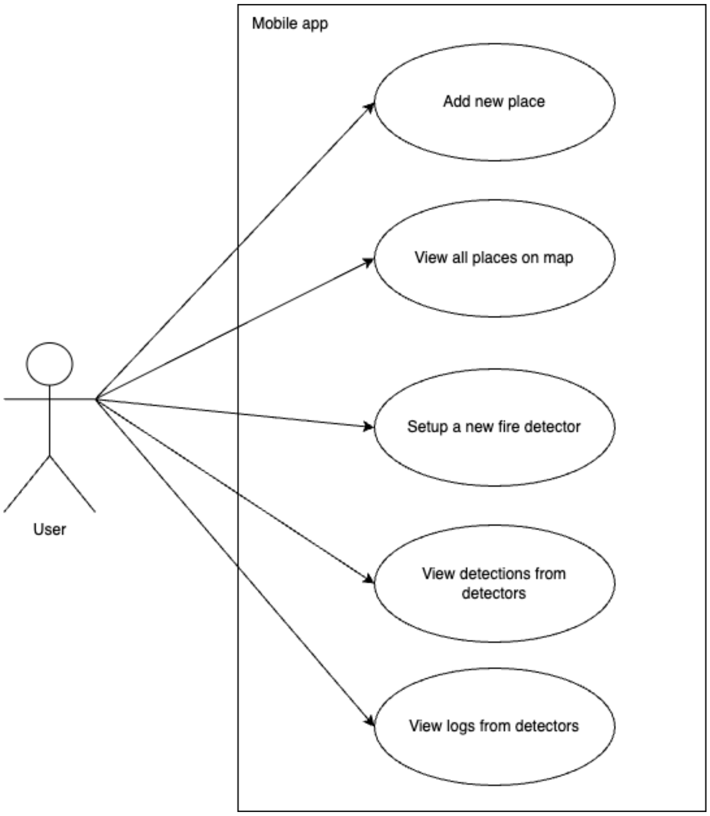


Fig. 4: Use case diagram for user

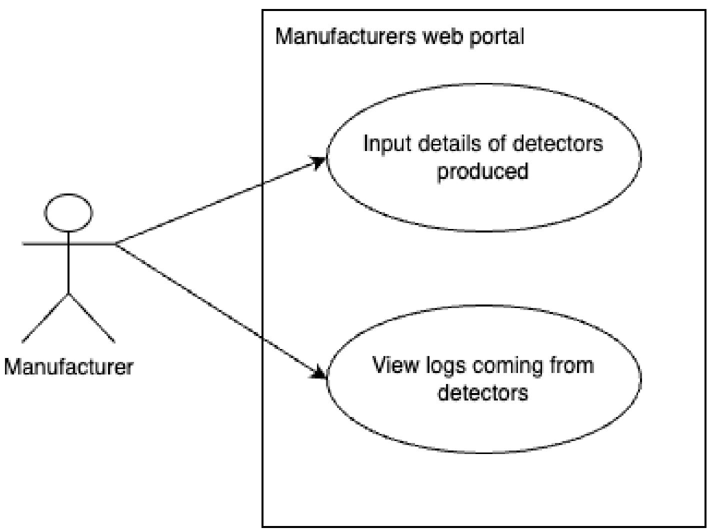


Fig. 5: Use case diagram for manufacturer

ending with the output. A convolutional layer with 32 filters makes up the first layer. Each of these 3x3 filters is applied to the input image in order to identify features and patterns. The non-linearity is introduced using the activation function 'relu'. Each convolutional layer is followed by a max-pooling layer (MaxPool2D) with a 2x2 pool size. By reducing the spatial dimensions of the feature maps, max-pooling captures

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 254, 254, 32)	896
max_pooling2d_3 (MaxPooling2D)	(None, 127, 127, 32)	0
conv2d_4 (Conv2D)	(None, 125, 125, 64)	18496
max_pooling2d_4 (MaxPooling2D)	(None, 62, 62, 64)	0
conv2d_5 (Conv2D)	(None, 60, 60, 128)	73856
max_pooling2d_5 (MaxPooling2D)	(None, 30, 30, 128)	0
flatten_1 (Flatten)	(None, 115200)	0
dense_2 (Dense)	(None, 128)	14745728
dense_3 (Dense)	(None, 1)	129

Fig. 6: CNN Layers added to the model

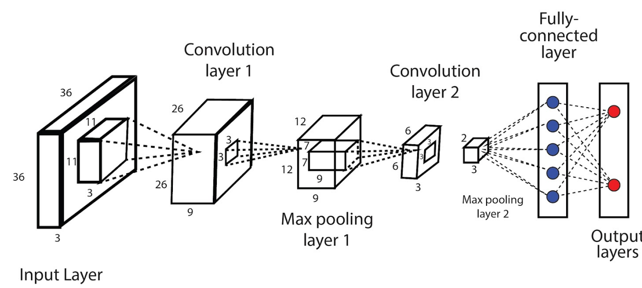


Fig. 7: Working of CNN model

the most pertinent data. Three further sets of max-pooling and convolutional layers with progressively larger filter sizes (64 and 128) are included in the model. The model is able to recognize hierarchical patterns and characteristics in the photos thanks to its deep architecture. A flattening layer that converts the feature maps into a one-dimensional vector comes after the convolutional layers. The flattened characteristics are then processed further by a fully connected layer (Dense) with 128 units and a 'relu' activation function.

A single neuron with a "sigmoid" activation function makes up the output layer. This is perfect for problems involving binary classification, in which the model determines whether an image is likely to contain smoke or fire. Binary cross-entropy loss and the Adam optimizer are used to construct the model. The binary cross-entropy loss function, which measures the difference between expected and actual labels, is a common loss function for binary classification, and Adam is an effective optimizer for training deep neural networks. While training the data we get an accuracy of 90.25%, while testing the model on unseen images it gives an accuracy of 96.37% .

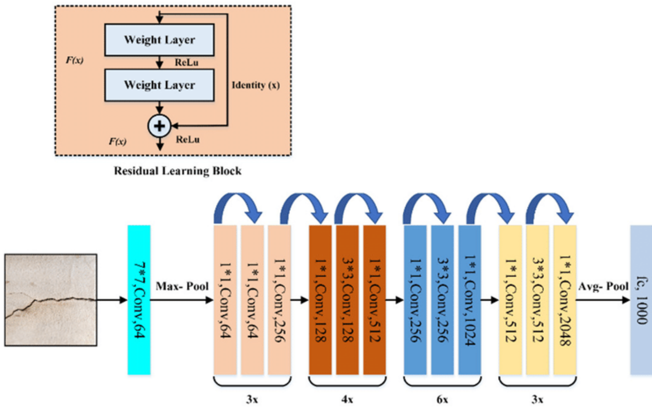


Fig. 8: Working of ResNet50 model

Keras is used in the ResNet50 model implementation to load pre-trained weights from the ImageNet dataset into the model [18] [19]. Fig. 8 shows the internal layers of this model. With its broad range of visual patterns and objects recognition, this pre-training gives the model a strong basis for feature extraction. The top (classification) layers of ResNet50 are not included since the ”include_top” option is set to ”False,” meaning that these layers will be specially designed to meet the demands of the particular fire detection task. The next phases describes customized top layers for the base model, which is an essential modification to tackle the problem of fire and smoke detection. The output from the base model can easily be retrieved by saving the model. A layer is added that presents a method called global average pooling, which produces a fixed-length vector for every image by reducing the spatial dimensions of the feature maps. This modification is frequently used to enable classification tasks to be performed on the ResNet50 output. Afterward, a fully connected layer with 128 units is added using the expression ”Dense(128, activation=’relu’)(x)” and an activation function of ’relu’. Processing the flattened characteristics and identifying higher-level patterns depend on this dense layer.

The output layer is defined by ”Dense(1, activation=’sigmoid’)(x)”. This layer, which consists of a single neuron and uses a ’sigmoid’ activation function, determines the likelihood that an image contains smoke or fire. A rating that is nearer to 1 suggests that there is a high chance of smoke or fire. The unique top layers are smoothly linked with the main Resnet model to complete the model architecture. Interestingly, the layers in the basic model stay frozen, meaning that they won’t be modified while the model is being trained. Rather, the custom top layers are where the fine-tuning is concentrated. This is a deliberate decision that makes use of the information that ResNet50 has learned from ImageNet to help it recognize features that are relevant to smoke and fire detection. The binary cross-entropy loss function and the Adam optimizer, which are common choices for binary classification problems, are used to assemble the model and train it. The binary cross-entropy loss is a crucial tool for improving the performance of the model since it measures the difference between the expected and actual labels.

While training the Resnet model we get an accuracy of 88.23%, while testing the model on unseen images it gives an accuracy of 89.25% .The model is trained using the ’fit_generator’ function. Working with data generators, which manage huge datasets with minimal memory usage, is the purpose of this function. Utilized are the validation and training data that were previously generated. During training, batches of preprocessed images are produced by these generators. The whole training and validation datasets are handled during training due to settings for steps per epoch and validation steps. Model-defined callbacks are used to record the optimal model checkpoints, keep track of the training process, and, in case of need, initiate early termination. These methodology demonstrate the effectiveness of our approach in smoke and fire prediction. Fig. 9 shows the training and testing of the CNN model with predictions.

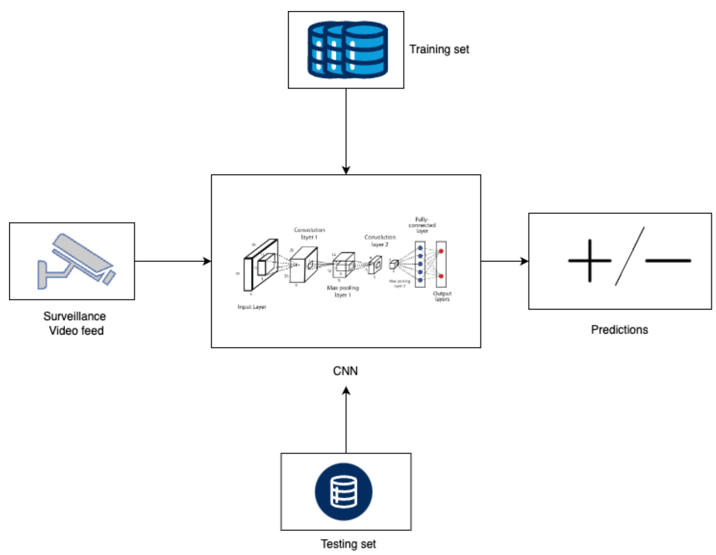


Fig. 9: Prediction diagram

Our work goes beyond image analysis to characterize regions that are prone to fire events on a regular basis. These strategies are vital for fire incidents prevention and minimizing the potential impact. An essential step in early detection is the calculation of fire risk probabilities, which can be modeled mathematically:

$$P(FireRisk) = f_{risk}(A, F) \tag{3}$$

Where P is the fire risk probability, A denotes the area characterization, and F represents fire incident features. This probability guides the allocation of resources and intervention efforts.In the mitigation phase, strategies may include controlled burns, early warning systems, and public awareness campaigns. The effectiveness of these measures can be assessed using performance metrics like precision, recall, and F1-score.

V. Result and Discussions

A. Simulation setup and tools

A powerful computational infrastructure, consisting of Kaggle notebook with an GPU P100 offers disk space of 73.1GB, RAM of 29GB, and 15.7GB GPU memory, supported our research by offering the capability needed for intricate calculations. Kaggle Notebook, a flexible cloud-based environment that allowed us to develop ResNet-50 and CNN. The libraries of choice were TensorFlow (Version 2.x) and Keras (Version 2.x), which were fine-tuned to extract complex characteristics from dataset photos. With the aid of this extensive toolkit, we were able to carry out an extensive research that included risk assessment, deep learning, and image analysis, leading to the development of an all-encompassing strategy for managing fire incident.

B. Performance analysis

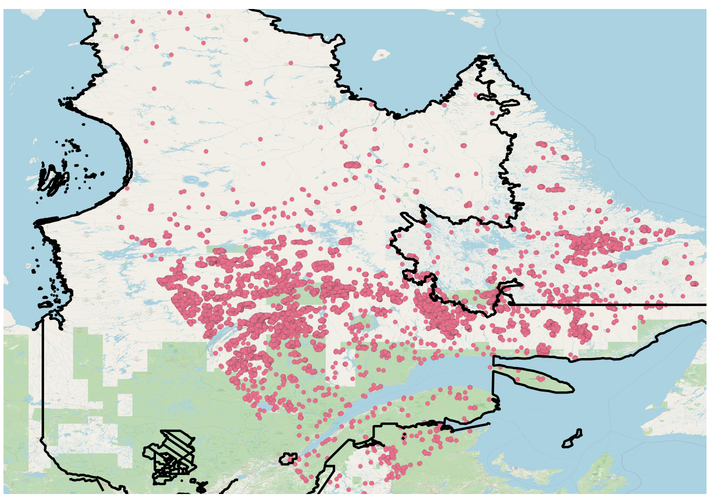
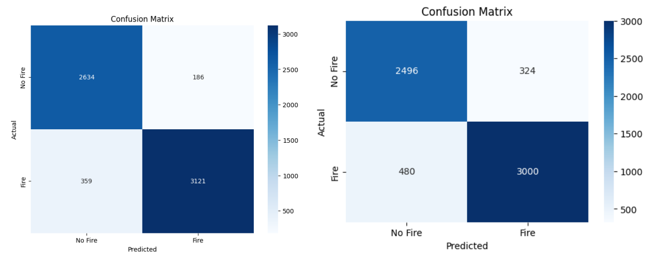


Fig. 10: Fire events map of Quebec, Canada

Fig. 10 shows Fire incident map of Quebec in QGIS, which helps in better visualisation of affected areas. The image shows that the central and southern part of the state is severely affected by this fire events since long time. Keeping in mind this situation the government authorities should take necessary action to reduce its impact.

Fig. 11a shows the confusion matrix for test images. This matrix proves the effectiveness of using DL techniques for fire detection. It shows that out of 2820 actual fire images it predicts 2496 correct and out of 3480 smoke image 3000 are predicted correctly with smoke hence it gives a good ratio of correct predictions.This means that a good number of guesses were right, which shows how strong our CNN model is. Our results strongly suggest that CNN is very good at making models for our fire data.



(a) CNN (b) ResNet50

Fig. 11: Confusion matrix

In parallel, Fig. 11b presents the confusion matrix for our ResNet-50 model, delivering equally good results. It shows that out of 2820 actual fire images it predicts 2634 correct and out of 3480 smoke image 3121 are predicted correctly with smoke hence it gives a good ratio of correct predictions. The output of the ResNet-50 model is very similar to that of the CNN, which shows that it is good at this task. These results support the use of deep learning models to find fire incidents and set the stage for successful early response and prevention plans.

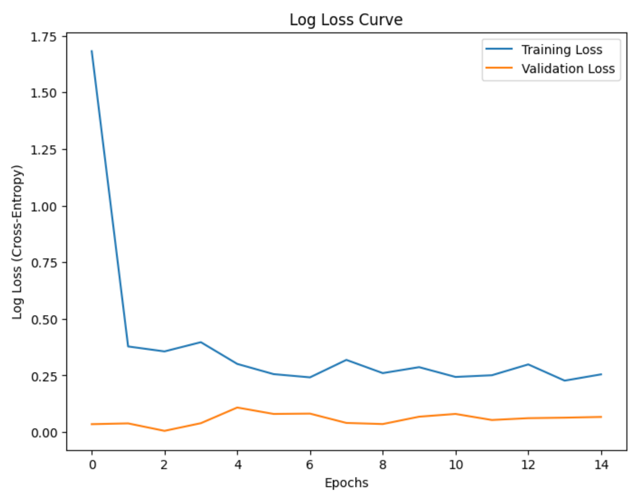


Fig. 12: Log loss curve for CNN

Fig. 12 shows the log loss (cross-entropy) curve for our CNN model, which shows how it learns over time. The initial sharp drop in log loss clearly shows that the model quickly picked up new features and learned representations from the data. After that, the curve steadily goes down for both the training and testing datasets. This shows that the model can transfer what it knows about fire incidents patterns. This downward trend shows that the model can make more and more sure and correct predictions. On the other hand, Fig. 13 shows that the log loss curve for the ResNet model has a more wavy shape, which means that the model is over-fitting and can't generalize well. Additionally, the ResNet model shows promise for feature extraction and complex representation capture. However, the

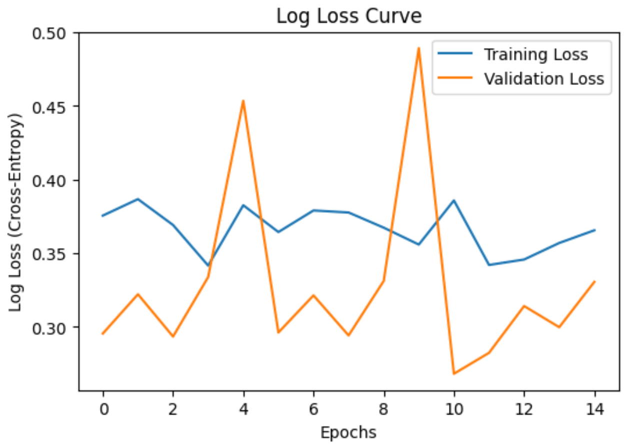


Fig. 13: Log loss curve for Resnet

CNN model's loss curve shows stability and consistency, which suggests that it would perform better at finding fire incident. These results show how well our DL models work and how important it is to choose the right model when trying to solve the problems of finding fire incident.

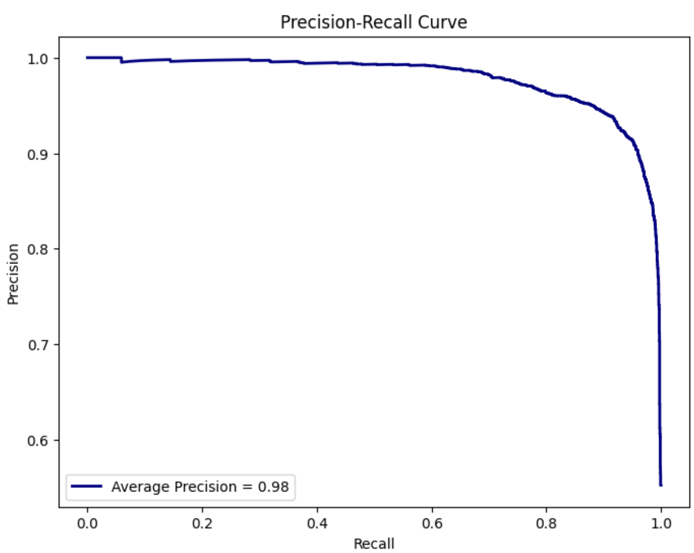


Fig. 14: Precision Curve CNN

Fig. 14 and Fig. 15 shows the most important part of our review. It shows how well a model can accurately predict positive classes while keeping false positives in check. The trade-off between recall and accuracy at different classification thresholds with these precision-recall curves. Looking more closely at our CNN model's accuracy-recall curve, it's clear that it knows how to get the best results for both recall and precision, with the goal of getting high numbers for both. The precision-recall graph for the ResNet model, on the other hand, is very different. It shows how well it can reduce false positives while still making accurate positive class predictions. There is, however, one feature that stands out during this comparison:

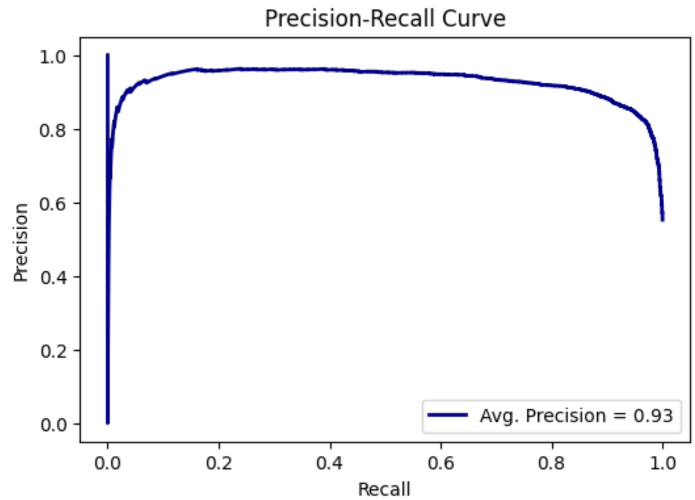


Fig. 15: Precision Curve ResNet50

a clear kink in the ResNet curve that shows a change in the model's behavior. When you put the two models next to each other, the CNN model stands out because it has a bigger area under the precision-recall curve. This means it does a better job of keeping the balance between precision and recall across different classification levels. This study shows that CNN is very good at finding fire incident, especially when it needs to accurately identify the positive class while reducing the number of fake positives. This makes it a powerful tool for real-world use.

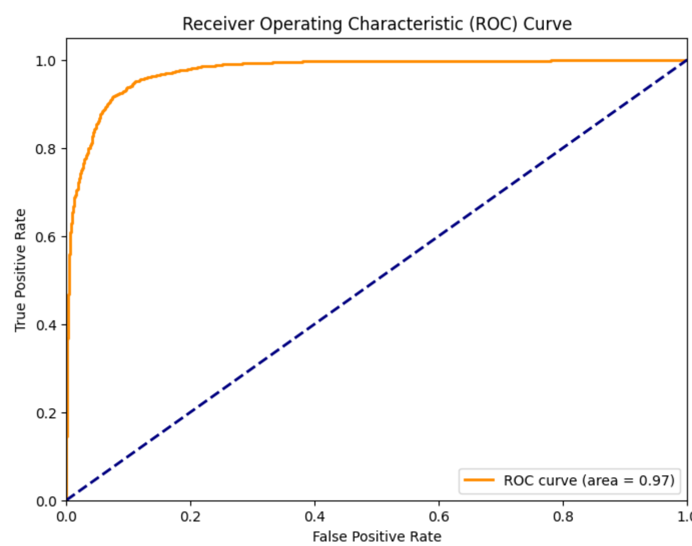


Fig. 16: ROC curve for CNN model

We tested our fire incidents detection models using ROC-AUC (Receiver Operating Characteristic - Area Under the Curve) measures, which gave us useful information about how well they could tell the difference between different types of fires. As the most important tool for testing binary classification models, ROC-AUC gives you a detailed picture of how well the model works at different levels of judgment. This measure is very helpful for figuring out how much the True Positive Rate (TPR) and False Positive Rate (FPR) are worth. Fig. 16 shows the ROC-AUC curve, which shows how well the CNN model can tell the difference between positive and negative samples. The path of the curve shows that the model can get a high TPR while keeping a low FPR, which shows that it works well for binary classification tasks. In this case, a bigger AUC score means that the ability to discriminate is better.

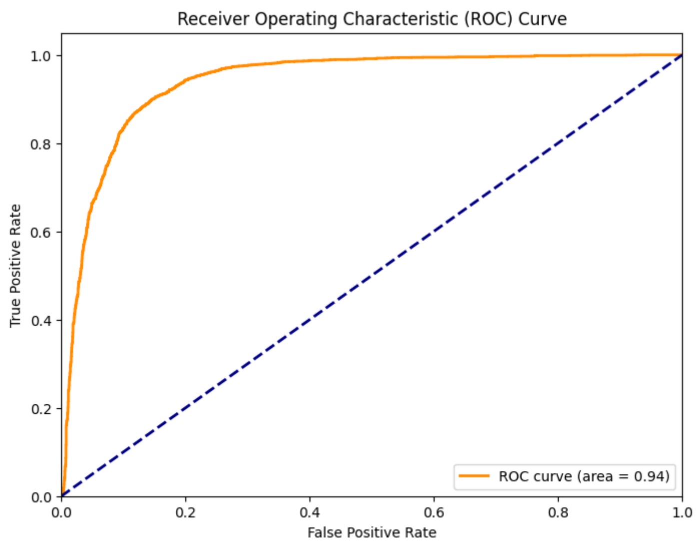


Fig. 17: ROC curve for ResNet50 model

Fig. 17 shows the ROC-AUC graph for the ResNet model, which also shows how well it sorts things into groups. The goal for both models is to get a higher ROC-AUC score, which shows how good the model is at binary classification tasks. The CNN model is better at telling the difference between classes than the other models. This is shown by its bigger AUC under the ROC curve, which shows that it can accurately tell the difference between forest fire incidents and non-fire incident cases. This analysis shows how good our models are at telling the difference between things and proves that the CNN model is the best at this particular forest fire detection job. These ROC-AUC tests are very important for figuring out how well our models work, which proves that our fire incident detection warning system works.

VI. Conclusion

In this study, we looked into fire incidents detection in a lot of detail, with a focus on how it works in the testing images. The main part of our study was using image data available and carefully visualizing it using various DL techiques. It became clear how important it was to have a strong fire prediction system because fire incident are becoming more dangerous, they can damage the environment, and they can ruin lives and businesses. We acquired image datasets, which we then analysed and processed to prepare them for deep learning techniques. We proposed CNN and ResNet, two powerful DL models that can automatically take features and classify them.
The CNN and ResNet parts of our study, without any external hyperparameter tuning, produced interesting results. The CNN model in particular, emerged as better model in our investigation, achieving an impressive maximum accuracy rate of 96.34 percent. This exceptional accuracy underscores the potential of CNN in the domain of fire incidents detection and demonstrates its proficiency in classifying images for this critical task. The significance of our research extends to its potential impact on fire incidents management by sending alert notification to it's user. By achieving high accuracy rates in early fire detection, we have the opportunity to minimize the ecological, economic, and societal repercussions of fire incidents. Early detection of these incidents promises public safety and environmental preservation in addition to enabling quick response and control.

In the future, we aspire to extend the application of our proposed architecture to address various image recognition challenges. Moreover, to enhance predictive accuracy, we intend to explore transformer-based architectures. Additionally, collaboration with local authorities and communities will be helpful to ensure the seamless implementation and effectiveness of real-time detection and response strategies.

References

[1] B. J. Cardinale, G. M. Mace, D. Tilman, D. A. Wardle *et al.*, "Biodiversity loss and its impact on humanity," *Nature*, vol. 486, no. 7401, pp. 59–67, 2012.
[2] A. S. Oliver, U. Ashwanthika, and R. Aswitha, "Detection of forest fire using convolutional neural networks," in *2020 7th International Conference on Smart Structures and Systems (ICSSS)*, 2020, pp. 1–6.
[3] M. D. Flannigan, B. J. Stocks, and B. M. Wotton, "Climate change and forest fires," *Sci. of the total env.*, vol. 262, no. 3, pp. 221–229, 2000.
[4] A. Isaev, G. Korovin, H. Shugart, N. French, B. Orlick, and T. Murphy, "Using remote sensing to assess russian forest fire carbon emissions," *Climatic Change*, vol. 55, pp. 235–249, 2002.
[5] C. Modarres, N. Astorga, and V. Meruane, "Convolutional neural networks for automated damage recognition and damage type identification," *Structural Control and Health Monitoring*, vol. 25, pp. e22–30, 2018.
[6] C. Funk, L. Harrison, D. Korecha, T. Magadzire, G. Husak, G. Galu, and A. Hoell, "Assessing the contributions of local and east pacific warming to the 2015 droughts in ethiopia and southern africa," *Bulletin of the American Meteorological Society*, vol. 97, no. 12, pp. S75–S80, 2016.
[7] M. Arif, K. Alghamdi, M. Alsahaft, M. Alharthi, and M. Arif, "Role of machine learning algorithms in forest fire management: A literature review," *J. Robot. Autom*, vol. 5, pp. 212–226, 2021.
[8] L. Alzubaidi, J. Zhang, O. Al-Shamma, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," *Jour. of big Data*, vol. 8, pp. 1–74, 2021.
[9] L. Wen, X. Li, and L. Gao, "A transfer convolutional neural network for fault diagnosis based on resnet-50," *Neural Comp. and App.*, vol. 32, pp. 6111–6124, 2020.
[10] P. Barmpoutis, P. Papaioannou, and N. Grammalidis, "A review on early forest fire detection systems using optical remote sensing," *Sensors*, vol. 20, no. 22, p. 6442, 2020.
[11] A. Jadon, M. Omama, A. Varshney, M. S. Ansari, and R. Sharma, "Firenet: a specialized lightweight fire & smoke detection model for real-time iot applications," *arXiv preprint arXiv:1905.11922*, 2019.

[12] C. Tao, J. Zhang, and P. Wang, "Smoke detection based on deep convolutional neural networks," in *2016 International conference on industrial informatics-computing technology, intelligent technology, industrial information integration (ICIICII)*. IEEE, 2016, pp. 150–153.

[13] A. Namozov and Y. Im Cho, "An efficient deep learning algorithm for fire and smoke detection with limited data," *Advances in Electrical and Computer Engineering*, vol. 18, no. 4, pp. 121–128, 2018.

[14] J. Sharma, O.-C. Granmo, M. Goodwin, and J. T. Fidje, "Deep convolutional neural networks for fire detection in images," in *Engineering Applications of Neural Networks: 18th International Conference, EANN 2017, Athens, Greece, August 25–27, 2017, Proceedings*. Springer, 2017, pp. 183–193.

[15] Y. Zhou, J. Wang, T. Han, and X. Cai, "Fire smoke detection based on vision transformer," in *2022 4th International Conference on Natural Language Processing (ICNLP)*. IEEE, 2022, pp. 39–43.

[16] L. K. Sharma, R. Gupta, and N. Fatima, "Assessing the predictive efficacy of six machine learning algorithms for the susceptibility of indian forests to fire," *Intl. jour. of wildland fire*, vol. 31, no. 8, pp. 735–758, 2022.

[17] K. Vani *et al.*, "Deep learning based forest fire classification and detection in satellite images," in *11th ICoAC*, 2019, pp. 61–65.

[18] Y. You, Z. Zhang, J. Demmel, and K. Keutzer, "Imagenet training in minutes," in *Proceedings of the 47th ICOPP*, 2018, pp. 1–10.

[19] J. Deng, W. Dong, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE conf. on comp. vision and pat. rec.*, 2009, pp. 248–255.