Fire Detection using Deep Learning Techniques

A Project Report

Submitted in the partial fulfilment of the requirements for the award of the degree of

Bachelor of Technology
in
Computer Science Engineering

By

Kausthubha Siddhartha Jilla	180330202
Kawali Nitin Raj	180330203
Kurumaddali Krishna Vamsi	180330207
Siddhantham Vardhan Kumar	180330225

Under the supervision of

Dr. K. G. SUMA, M.E., Ph.D.,
Associate Professor – Department of Computer Science and Engineering

Department of Computer Science and Engineering

K L University Hyderabad

Aziz Nagar, Moinabad Road, Hyderabad – 500 075, Telangana, India

November, 2021

DECLARATION

The Project Report entitled "**Detection of fire using Deep Learning and Image Processing**" is a record of bonafide work of Kausthubha Siddhartha Jilla - 180330202, Kawali Nitin Raj - 180330203, Kurumaddali Krishna Vamsi - 180330207, Siddhantham Vardhan Kumar - 180330225, submitted in partial fulfilment for the award of B. Tech in the Department of Computer Science and Engineering to the K L University, Hyderabad. The results embodied in this report have not been copied from any other Departments/ University/ Institute.

Kausthubha Siddhartha Jilla (180330202)

Kawali Nitin Raj (180330203)

Kurumaddali Krishna Vamsi (180330207)

Siddhantham Vardhan Kumar (180330225)

CERTIFICATE

This is to certify that the Project Report entitled "**Fire Detection using Deep Learning Techniques**" is being submitted by Kausthubha Siddhartha Jilla (180330202), Kawali Nitin Raj (180330203), Kurumaddali Krishna Vamsi (180330207), Siddhantham Vardhan Kumar (180330225), submitted in partial fulfilment for the award of B.Tech in Computer Science and Engineering to the K L University, Hyderabad is a record of bonafide work carried out under our guidance and supervision.

The results embodied in this report have not been copied from any other departments/ University/Institute.

Signature of the Supervisor

Dr. K. G. SUMA

Associate Professor, Dept. of CSE

Signature of HOD

Signature of the External Examiner

ACKNOWLEDGEMENT

First and foremost, we thank the lord almighty for all his grace & mercy showered upon us, for completing this project successfully.

We take grateful opportunity to thank our beloved Founder and Chairman who has given constant encouragement during our course and motivated us to do this project. We are grateful to our Principal **Dr. L. Koteswara Rao** who has been constantly bearing the torch for all the curricular activities undertaken by us.

We pay our grateful acknowledgement & sincere thanks to our Head of the Department **Dr. Chiranjeevi Manike** for her exemplary guidance, monitoring and constant encouragement throughout the course of the project. We pay our sincere thanks to our Project Coordinator **Dr. K. G. Suma** who has been constantly inculcating logistic support and has given constant encouragement during our project. We thank **Dr. K. G. Suma** of our Department who has supported throughout this project holding a position of supervisor.

We whole heartedly thank all the teaching and non-teaching staff of our department without whom we won’t have made this project a reality. We would like to extend our sincere thanks especially to our parent, our family members and friends who have supported us to make this project a grand success.

Kausthubha Siddhartha Jilla (180330202)

Kawali Nitin Raj (180330203)

Kurumaddali Krishna Vamsi (180330207)

Siddhantham Vardhan Kumar (180330225)

ABSTRACT

Using Image Processing to make a Fire Detection System that will act as the early warning step, instead of using sensors and extra equipment for the very basic step towards protection against fire in communities, apartments, and smaller shops and businesses. Utilizing the existing cameras as an input device for the model, which will process and detect whether there is a fire burning in its view and send an alert to the user. The model/software will be easy to install and use for anyone.

The proposed model is based on Convolutional Neural Network (CNN) based InceptionV3 deep learning model. This model is highly useful in object detection and image analysis. It contains a deeper neural network which helps in the limitation of increasing parameters. Traditional convolutional neural network Inception V3 model utilizes parallel processing and pipeline structures were it not only uses a convnet of a single size it uses multiple sizes over multiple Inception layers in order to get a better result for the object detection in this case a fire.The InceptionV3 model is based on GoogleNet architecture. This architecture is inspired by ImageNet and AlexNet architectures. As it borrows many layers and structures from these, it allows us to detect the required object presence in the given input.

The model takes images as input and performs data pre-processing on them. The processed image is sent to the model and the deeper neural networks try to detect the fire region present in the image. Not all locations weather conditions camera quality and other parameters are the same in various places after being trained on the traditional data service the model as a continuously received the input will Store the input files and then should later on be able to train with these new images in addition in order to increase its practicality.

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER 1
Introduction

Gone unprotected in any ecosystem a forest village or a concrete jungle Like most of US are living today a fire can be one of the most dangerous things that is quite fast spreading and most destructive in terms of both property and life, taking steps to first Prevent it from ever happening and then if it already happened to make sure it is detected at the earliest possible second and to make sure it is dealt with the utmost speed. Due to the modernisation, many new infrastructural activities like skyscrapers and huge buildings which are packed are been constructed. This modernisation becomes a factor of increasing fire incidents. Due to fires in many forms, the destruction caused is huge in scale which later makes a dent in economy, livestock and in general people's life.

To avoid the mass destruction from fires, they were many fires detection equipment and techniques were been introduced. These techniques include smoke detectors based on fire outputting the smoke levels which require to turn on the water sprinklers. Some other techniques include heat sensors which detect the temperatures inside a room where the fire is spreading rapidly. Some of the other techniques listed below:

A. Fire detecting sensors based on heat and smoke using water sprinklers and heat sensors.

In many commercial places like shopping malls, movie theatres, community halls etc, these systems are used to tackle fire related conditions. But these systems lack, false or fake fires as these systems can be tricked by heat concentrated in that particular room. Due to their low-cost expenses and low maintenance, it is preferred more.

B. Satellite Sensing for detecting fires in forests areas

As wildfires are frequently been occurring in forests around the world. The detection of these wildfires is done through satellites imaging. This approach is taken because, people can’t go and physically take precautions for fires occurred in forests.

C. Aerial fire sensing through drones and high-end equipment

With the advancement of drone technology and robotics, it paved the way to sense the fires from large distances. But it comes at a cost of high expenses for maintenance and usability.

The idea of using our already existing vast surveillance system is to also act as the first step towards the detection and fast response towards a fire that is what is being proposed using the already existing cameras as the input system for a deep learning model to process and detect whether a fire has started which will eliminate some of the limitations that the above mentioned techniques have such as having the upfront cost to put in this measurement also has the advantage that there is no change to the infrastructure that already exist to be done as the model is a piece of software that runs on an already existing computer or a server that most surveillance systems are connected to. The idea behind image processing and usage of deep learning for fire detection using images and video captures is to avoid risk factors. This risk factors may include but not limited to i.e., make people to get equipped with the technology at their fingertips to make quick and smart decisions. These decisions can be made by the integration of smartphone technology with the deep learning and image processing model bundled with a software to interact. The other reason is to reduce the expenditure done on high end utilities. Some of the utilities such as heat sensors which are useful to certain extent but can be tricked to give out false results. To avoid this, we can make the surveillance cameras integrated with this model which can take real-time camera video feed and process it to give out results which can alert people in that premises and may help in saving lives.

CHAPTER 2

Literature Survey

Sheng, et al. [1] proposed that as existing solutions have high computational requirement and are slow and don’t work well with surveillance systems, the mentioned project utilizes a cost-effective CNN model based on the Google Net architecture – Using a custom model built for fires like an inception model from google itself gives better results and also with the recent progression and development in technological capability we can use complex models that run quickly. The mentioned project has a high false detection rate whereas our proposed method has a false positive detection nearing 100%.

Zhang, et al. [2] proposed utilization of fuzzy sets and relations to select features and detect fires from spatial images, although it is a good idea for forest management systems that have such large-scale satellite monitoring those that utilize cameras or surveillance within the forest cannot make use of this system and would require a more neural network style solution.

Celik, et al. [3] proposed that background subtraction is to be applied to movement containing region detection. Secondly, converting the segmented moving regions from RGB to YCbCr colour space and applying five fire detection rules for separating candidate fire pixels we can also change the colour space and modify the rules to make any video input be directly processed and reduce a step without conversion. it also provides a false detection based on colour through temporal variance this will work in the setting of forest fires and largely natural areas but not in modernised locations.

Poobalan, et al. [4] proposed using Background subtraction and special wavelet analysis, and an SVM for false detection although CNN seems like a better choice as there will be huge amount of data as there will e constant input when we consider a surveillance system that is modern and can catch the more nonlinear correlations better and as CNN supports parallelization the throughput time can be improved.

Vipin, et al. [5] proposed utilizing Convolutional Neural Networks using DBSCAN where they try to recognize flame and smoke modes connected to fire stages. DBSCAN is used for noise detection and pre-process the step to over segment images into super-pixels. The end goal is to detect smoke and fire. The limitations of the projects are in terms of the computational work being taxing on the hardware as it uses DBSCAN to detect and separate the fire and smoke parts and store it in the database. It may also delay the rate of giving out the results.

Mahmoud, et al. [6] proposed model in this based on effective SqueezeNet based asymmetric encoder-decoder U-shape architecture, Attention U-Net and SqueezeNet (ATT Squeeze U-Net). The functioning of the model on images is both extractor and discriminator of forest fires. The model focuses on taking useful information through Attention Gate and ignore irrelevant information. The convolutional layer is replaced with Channel shuffle layer to encode and decode the useful information. The claimed prediction time is 0.89 second per image. The drawback of the model is it tries to replace the convolutional layer with color channel shuffling layers. If a scenario turns up and the image has more region of fire in it, this model may or may not be able to detect the fires in it. The classic convolutional layer is better at detecting the region of fires. As the model is targeted towards forest fires, the region captured in the image would be huge enough to pass through the irrelevant information.

Mahmoud, et al. [7] proposed utilizing a rule based image processing technique and image segmentation through the YCbCr and RGB color spaces to separate luminance for false detection and although it proposes a low false detection rate it is a very high 14% and also is linear in sequence and processing which may make it cost effective but not time effective and also the model hasn't been tested in the real world system only on images making it unreliable in the a surveillance situation as it becomes slow and nonfunctional as it cannot handle multiple input streams.

Angayarkkani, et al. [8] proposed to detect the color of the fire which is mostly red in colour. Then we used the sobel edge detection on the original image to detect the edge of the fire while removing threshold which is less than 100. Then we applied the segmentation technique which used the combine the result from the first technique and second technique to separate the ROI of the fire from the background, the Sobel method cannot produce accurate edge detection with thin and smooth edge as fires are in general when viewed from basic camera equipment and in surveillance systems.

Yuan, et al. [9] proposed utilization of the n CIE L*a*b* color space to identify fire pixels and then uses motion detection for the final prediction process this system provides a very good solution to a forest fire problem but has a lot of assumptions like the natural progression of fire shape and doesn’t account for explosive type fires and also the motion detection can be fooled in a modern situation where there are many moving instances thus increasing the false detection rate, texture and shape information of fire regions should also be investigated to improve the system’s fire detection performance.

Nguyen, et al. [10] proposed this paper projected a model using opening-free that is appropriate for use in miscellaneous weather environment. Their proposed model reaches a goal extreme accuracy accompanying a depressed false alarm rate. Unlike different form that apply fire classifiers to whole counterpart, resulting fashionable not found or noticed detections of small fires, their projected method combines a fire person desiring political office extraction stage because our order can detect differing judge fires. In addition, this method maybe fast by way of processing narrow trim off fire images as the recommendation of the CNN-LSTM model.

Zhou, et al. [11] says that this research be partly responsible for to the methodical study of part of material world in three habits. Firstly, a correct and effective order for automated household fire load acknowledgment that exist proposed establish counterpart instance separation, this provides a established institution for further fire load belief, genuine in existence-opportunity fire load monitoring, and risk appraisal. Secondly, preparation method for a deep education model in a in or by comparison limited dataset of household images, to a degree in what way or manner to overcome the overfitting question and addition of hyperparameters, which can supply intuitiveness and direction for future accompanying studies. Finally, an indoor counterpart dataset accompanying smallest element of an image-level of fire load instances and its gauge for judgment happen set up to endure, that could be second hand as a basic standard or level dataset to assist from now on incident of indoor fire load acknowledgment form or datasets.

Xu, et al. [12] proposed that in this paper, they grown a novel fire discovery arrangement using deep Long-Short Term Memory (LSTM) affecting animate nerve organs networks and variational autoencoder (VAE). Their objective search out makes or become better in contact the performance of different existent discovery order. To evaluate the influence of our plan, they grown a set of computational experiments accompanying high-loyalty LES information in visible form, and they second-hand datasets from genuine in existence-class of existing beings fires (flaming, simmer, and being cooked lubricate fires) and non-fire experiments. They judge and compared the efficiency of their projected fire discovery accompanying alternative methods, that is to say the standard LSTM, CUSUM, exponentially burden affecting average (EWMA), and two now used established-hotness heat detectors accompanying thresholds of 47∘C and 58∘C each.

Rashkovetsky, et al. [13] proposed with this paper, we can reinforce that deep education shows great potential for the done or made by machine and healthy discovery of wildfires from openly ready for use multisensorial detached become aware of imagery. While fashionable clear, cloud-free weather condition discovery rates until 92% are bring to successful conclusion utilizing just ultimate acceptable sensor (Sentinel-2), and up to 96% when make use of best choice multi-sensor melding scenario (Sentinel-2 and Sentinel-3), they bear exist smart to show that data mixture can mainly happen confirmed as advantageous. This holds, specifically, if various optical sensors happen linked all the while clear noncloudy conditions or if Sentinel-1 SAR comment on something scrutinized exist make use of to support the optical remark all the while hazy weather.

Yang, et al. [14] proposed using the MobileNet-Fire network, that bear a lower computational cost and achieves better acting compared accompanying existent CNNs. Depth-wise breakable loop, that exist the basis of MobileNet-Fire, learns weights channel-intelligent. When classifying objects bear distinguishing banner, the channel multiplier can embellish the effect of depth-wise breakable convolution by stress distinguishing color channels. In addition, the SE block make or become better the channel-intelligent worldwide likeness of the results of depth-wise separable spiral, admit in use at the time removal from whole of the facial characteristics of an object. Furthermore, the channel multiplier maybe easily used to network to categorize objects of distinguishing colour, apart from fire. They used Swish10, that is a rule Swish, as the incitement function. Swish10 clips abundant weights and thickened fruit prepared for the weights that exist raised by the channel multiplier. The number of tier and feature maps exist persistent tentatively because the facial characteristics suitable for fire objects maybe cull efficiently, by that lowering the number of limits. MobileNet-Fire achieved 95.44% precision or correctness for fire discovery, that exist above those of the existent networks compared.

Huang, et al. [15] have noticed that accompanying the recent very quick happening of computer-generated images and hardware, VR electronics bear filed a in or by comparison mature application state and bear exist widely used in fire discovery. Image-type fire flame acknowledgment procedure are odd for early fire flame acknowledgment. This study bear investigated figure-type fire flame acknowledgment utilizing a support heading machine and harsh set belief. In realistic applications, the RS procedure exist alert noise and weak fashionable weakness fortitude and generalization, while SVM bear forceful antagonistic-noise ability to perform and inference act. This paper has bestowed an RS-SVM fire flame acknowledgment treasure and designed a classifier of fire flame concept acknowledgment. By constructed dwelling a model accompanying SVM, the parameters of that bear exist optimized, this study second-hand more feature variables as test, presented the static and vital facial characteristics of flames, picked and extracted ultimate direct feature subsets, meld the facial characteristics of fire flame images glean, and made orderly necessary training to understand; and extract flame domain. Their exploratory results shown that the projected fire flame acknowledgment plan of action yields a extreme recognition rate, a fast acknowledgment speed, wonderful strength, and a wide range of putting substance on another.

Lin, et al. [16] proposed have described a very involved in activity fire-discovery algorithm for use accompanying information in visible form acquired apiece Chinese geostationary small planet that revolves around a larger one, FengYun-2G. Numerous active fire-discovery algorithms have happen grown in the past, but assorted of bureaucracy are create for geostationary information in visible form and even fewer for Chinese smaller country dependent upon a larger one data. FengYun-2G exist a functional satellite and specify research worker with 5-km relating to space resolution information in visible form each hour. As the relating to space study method helps fashionable do away with many of the unrelated pixels, the extreme temporal determination and period series examination and determination assist fashionable characterizing pixels and prevent mistakes bring about by coarse relating to space resolution. Fully cultivate the potentialities momentary-series reasoning evolve into essential. Their proposed invention made up of three major steps: information in visible form pre-processing, spatial study, and worldly analysis. A worldwide order for processing satiated in hunger-round object images happen introduced and bequeath to another considerably enhance the adeptness of further computed or estimated amount especially fashionable research, where the prominence grant permission be reduced ahead of certain local district. During the relating to space analysis, potential fire tests happen utilized accompanying the support of a consolidation of fixed and active thresholds to remove non-fire pixels to the max range. The coming depending upon a set of circumstances tests further help to narrow down the range of pixels. Subsequently, a temporal length of event or entity's existence series reasoning plan is brought in place data sequences accompanying the alike observing period are captured as recommendation data. Different from prior research place people reliable to imitate the patterns of the BT values of during the day changes, the solid vacillation ranges happen the core elements of larger object to capture. For every smallest element of an image to be proven, the input information in visible form happens analysed establish the information of students' t-classification. Thus, belief in oneself intervals accompanying high percentile number exist build. Active fire detections are created in accordance with the calculated belief in oneself pause and the max profit from the input information in visible form series. Their algorithm processed resistant and fast. However, the results still have weighty question to be established. Obvious errors, particularly something forgotten or excluded errors, exist a weighty problem as it stands in addition to the case with added geostationary algorithms.

Kim, et al. [17] proposed a trustworthy constructed dwelling fire discovery framework accompanying simulation-located learning for fire protection from harm fashionable specific constructed dwelling environments. To produce synthetic preparation information in visible form that reflect real physical constructed dwelling, they first presented form and imitation methodology accompanying structural, state, and occurrence representations. Then, MEDNet happen devise to detect fire occurrence and produce a reconstruction wrong to predict dissimilarities middle from two points genuine in existence input information in visible form and all artificial training information in visible form. The unlikeness prediction authorizes the operation of the change mechanism for fear that incorrect predictions, that can occur if surprising scenarios happen attack, by utilizing the information in visible form-independent information-located method fashionable a timely style.

Xie, et al. [18] have noticed how growth of Computer Vision Technology, Deep Learning bear been used for fire discovery by many researchers. Although specific use is doable secondary certain environment, their effectiveness needs to be in better health, and complex related to the televised image scenarios must happen considered. Motivated by this concern, an adept fire detection form exists proposed in this place paper. Their order offers several benefits over different recent fire discovery design. First, both motion and flicker facial characteristics exist considered, that allow us to in a more excellent manner extract dynamic facial characteristics. In addition, their form relies on an adjusting inconsequential neural network, that can efficiently extract deep static facial characteristics accompanying a low computational cost. Finally, exploratory results establish facts that their method achieves pertaining to highest level of development at time depiction in agreement of allure accuracy and needless alarm or worry rate and that our pattern exist applicable to complex related to the televised image master plan. In general, their proposed order bear better application prospects than additional pertaining to highest level of development at time methods and exist appropriate for use in public protection from harm running an organization system.

Cheng, et al. [19] proposed a Neural Network utilizing temperature, fume aggregation, and CO aggregation to discover and predict fires. The fire discovery Neural Network bear a fire declaration made in advance mistake rate for three fire states of less than 5%. The extreme acknowledgment rate concerning this Neural Network will greatly weaken the risk of dishonest alarms and leak-checks of alarm warning of fire arrangement. This Neural Network fire alarm scheme can break down to components a sort of sensor information in visible form at the same time and can make or become better the talent of whole to adapt fashionable their surroundings and correct for meddling to correctly predict fires. This paper in addition to explain in speech in what way or manner Neural Networks can connect sensor data fashionable an very smart mechanical fire alarm structure to efficiently label fires that has excellent meaning for protection from harm.

Xu, et al. [20] proposed to grow a Deep convolutional Generative Adversarial Network (DCGAN) for very correct visual fire discovery when preparation figure happen restricted. Their model writes directions for delivery three types of error fashionable able to be seen with eyes fire discovery when preparation information in visible form exists limited, that is to say model overfitting, fire likelihood of something happening miscalculation, and fire likelihood of something happening miscalculation wrong. Their DCGAN includes an engine converting energy of fake fire concept for Self-Supervised Learning (SSL), and a discriminator to categorize the counterpart as fire, non-fire, and fake. They create computational experiments accompanying a diverse and of a superior nature fire discovery counterpart dataset to authenticity of something their model against additional directed learning approaches. They checked the precision or correctness of four models, the directed naïve CNN, the directed fire discovery SQN, their DCGAN accompanying a CNN for discriminator (DCCNN), and their DCGAN accompanying a fire detection SqueezeNet for discriminator (DCSQN), accompanying preparation extent or bulk of some dimension varying from 30 to 300 concept. The DCSQN bring to successful conclusion the best experiment precision or correctness over all preparation dataset sizes. They before benchmarked the depiction of their DCSQN against a best-fashionable-class deep visual fire discovery model, FireNet. The results of their computational experiments actively exhibit something that, the DCSQN considerably outperforms FireNet ahead of all conduct versification (accuracy, dishonest beneficial, dishonest negative, accuracy, recall, and F-score) when prepared accompanying the same dataset extent or bulk of some dimension, the DCSQN model efficiently mitigates overfitting when the preparation dataset exist restricted, and more generally, the results display that the DCSQN efficiently addresses the three types of wrong fashionable able to be seen with eyes fire discovery when training representation exist restricted.

Chaoxia, et al. [21] proposed two facts-direct flame detection system. The first form happens a novel colour-lead anchoring plan of action, that happen a smooth-to-implement system to improve the effectiveness of Faster R-CNN inflame discovery. The second means is grown to deal with the question of fake positives, by utilizing the all-encompassing news-direct flame discovery which considerably reduces the needless alarm or worry rate through a be alike GIN. They distinguished their methods accompanying three existent everything utilizing the BoWFire dataset, and the results show that our pattern achieved better precision or correctness and made smaller wrong positives. They in addition to attend a speed test ahead of the BoWFire dataset to put to a test the favored position or circumstance of the color-guided hold plan of action. Experiments in contact the Corsician Fire dataset and the PascalVOC2012 dataset authenticity of something the robustness of their arrangement. They in addition to written the explanatory note files of BoWFire and Corsician, so the subsequent tests ahead of two together information in visible form sets maybe skillfully performed. The in better health results of our pattern exist accredit two reasons. One explanation exists the strong deep facial characteristics second-hand two together by Faster R-CNN and GIN, and the other exist the copiousness of our preparation dataset. In future work, they will study in what way or manner to share the feature map middle from two points Faster R-CNN and GIN to decrease the computational repetition by preparation two together of ruling class simultaneously. Finally, they in addition to start the projected pattern on a fire-fighting android and test allure influence in essence.

Nakip, et al. [22] proposed a novel Recurrent Trend Predictive Neural Network (rTPNN) which captures the style momentary series information in visible form via allure repeating internal construction. rTPNN performs information in visible form mixture on the multivariate period series accompanying allure captured fashion that is in favor values and considerably make or become better the prediction efficiency.

Jiecheng, et al. [23] proposed using ultraviolet dissemination detector and the shade resembling such a colour dissemination detector bother the footing of range analysis hypothesis fashionable fire discovery, and it has not exist thought-out that the fire image bear many trait essentially. This system adopts very smart data conversion to discover fire flame and give a alarm warning of fire in accordance with the characteristics of fire flame, that will not present a needless alarm or worry for fixed light point of supply in the way that an glowing bulb and a glowing light or passionate objects as burning cigarettes and candles. Its resistance of some degree and dependability bear been confirmed fashionable many applications. Compared accompanying the normal fire automobile-alarming structure immediately common, this system bears many benefits such as interminable distance, best discovery range, lower rate of false alarm and taller level of ability to perceive and imagination.

Graff, et al. [24] proposed the use of Regression models considerably outperforms established persistence-located models used in operational smoke forecasting applications at both the cell and regional levels. In particular, the use of Poisson hurdle models can efficiently use past fire state of being active and weather data and support considerably correct forecasts of fire enterprise until five days into the future compared to added approaches. When express an outcome in advance for all days, Fire and non-Fire, it happens influential to give reason for the important presence of inactive days, like, utilizing a hurdle model, preparation separate models. They bear shown that the non-Fire declaration made in advance task happen troublesome and possibly better judge as a ranking or explosion likelihood of something happening question than accompanying RMSE on account of the distort distribution. The use of a relating to space gridiron happens direct, but create several restraints that bring to mind a inclusive relating to space approach should to further make or become better model performance and exercise of ready for use information in visible form.

Cao, et al. [25] suggest an consideration enhanced bidirectional LSTM network (ABi-LSTM) for early area with a large number of trees fume acknowledgment. Specifically, the projected approach can be give a rundown as three parts; an Inception V3 network that exist used to extract relating to space features from cigarette job patch gradual; Bi-LSTM model which exist plan to extract material facial characteristics from forward and backward order by feed relating to space feature of sole patch; consideration network happen employed to make perfect categorization process accompanying a soft consideration means that can certainly measure the standing of different frames. Extensive experiment results put on display that the projected ABi-

LSTM core obtains taller accuracy fashionable early area with a large number of trees fire fume recognition distinguished accompanying added order. Moreover, ablation study happen attend to judge the efficiency of each sub-model fashionable ABi-LSTM.

Kim, et al. [26] proposed a deep learning-based fire discovery pattern, called DTA, that pretend to be the human discovery process. They assumed that the DTA process can considerably humiliate erroneous fire discovery. The projected design used the Faster R-CNN fire discovery model to discover the SRoF based on allure relating to space facial characteristics. Then, the features give a rundown from the SRoFs and non-fire domain fashionable successive frames exist gather or amass something by LSTM to classify either skilled happen a fire or not fashionable a temporary period. The following temporary strength of mind or will are at another time linked in the majority casting votes for the indispensable content fashionable a long-term ending. In addition, the region of both flame and fire happens figured by mathematical calculation and their momentary changes are stated to make sense of the dynamic manner of conducting oneself of the fire accompanying the last fire decision.

Zhang, et al. [27] proposed to use a Deep Learning approach for training forest fire detector. The projected model exists of a full image CNN and local patch NN classifier, two together classifiers share the unchanging deep affecting animate nerve organs networks. Besides, they proposed a fire discovery reference point on account of the lack of standard dataset in the data processing machine dream society. For the future work, they bequeath to another like to increase the size of the reference point and form delicate grained annotations, for instance, using calculate pictorial device that drives a machine to 573 create artificial information in visible form and create smallest element of an image-wise annotations completely free. They in addition to plan to affix the proposed fire discovery arrangement ahead of UAVs real world area with a large number of trees fire discovery.

CHAPTER 3

HARDWARE AND SOFTWARE REQUIREMENTS

3.1. Software Requirements

Python

Python is a general purpose interpreted high level programming language. It emphasises on code readability with its significant indentation usage. It is dynamically-typed and garbage-collected. It includes many python standard libraries and also support for open-source and highly proactive community for development of python.

Key features of Python:

- General Purpose high-level interpreted programming language.
- Easy to code with code readability and indentation use.
- Readily available standard libraries.
- Support data visualization and superior for graph plotting.
- Highly used in Machine Learning, Deep Learning, Image Processing etc due to the extensive support of libraries.

Tensorflow/Keras

Tensorflow is a library in python which focuses on Machine Learning, Deep learning and other fields of Artificial Intelligence. Keras is a framework in tensorflow which contains all the functions and numericals to perform operations on complex data such as image, videos etc formats.

Operating System

The recommended operating system is Microsoft Windows 7/8/10/11 or Linux distributions like Ubuntu, Pop OS. The operating system should contain python with the mentioned packages to be installed.

3.2. Hardware Requirements

Camera

As the project requires the hardware such as camera, which can be of point and shoot type, mobile phone, or professional grade. The required specifications of the camera should be with 480p plus resolution with 4MP camera sensor. This allows to capture the image or record video with good amount of precision.

Processor

- A computer system with a dual core processor or more is required to perform computational works.
- The required frequency for the processor should be 1 gigahertz (GHz) or faster 32-bit (x86) or 64-bit (x64) processor.

Random Access Memory (RAM) & Read Only Memory (ROM)

- Minimum of 1 gigabyte (GB) RAM (32-bit) or 2 GB RAM (64-bit) or more of similar configuration.
- Minimum of 50GB or more Storage.

Optional Hardware

- Optionally, internet connection of 1 mbps or more.

CHAPTER 4

FUNCTIONAL & NON-FUNCTIONAL REQUIREMENTS

4.1. Functional requirements

- Detects Fire through the video input from the camera.
- Detection under 5 seconds as in each frame that has been scanned will detect and give an output to the system within 5 seconds.
- Tracks video at 12 fps that is per second 12 frames are scanned from the camera input.
- Sends an alert on detection of fire, when the prediction made by the model is true i.e., it has detected a fire an alert is sent that is a message to the system admin and if an already existing solution like a sprinkler system exists an activation message will be sent.

4.2. Non-Functional requirements

- Highly scalable, the application is easy to implement as for all inputs the model can be connected to all the inputs.
- Easy to install, the model just needs to be connected to output stream of the camera and the start hr model and the prediction will start.
- Easy to maintain there is no updation that the model needs to be done as it is already prepared if and when there is a model updation the model is updated into the same work flow and no changes by the user are required.
- Low cost as there is no physical sensors or hardware the cost of the solution is not high.
- Reliable the model has to have an accuracy of over 80% the model is reliable making the overall system reliable.

CHAPTER 5
PROPOSED SYSTEM

5.1. Proposed System

We know already that a lot of our daily lives are under constant surveillance, this implies the existing network of surveillance equipment or cameras which can be used as the input system for the system or any simple basic HD resolution recording device can be used as the input. This input is now sent to the processing model this model consists of a Convolutional Neural Network model that has been trained already on the "Fire Detection Dataset – Kaggle, Atulya Kumar", "FIRE Dataset – Kaggle, Ahmed Saied", "DeepQuestAI - Fire-Smoke-Dataset GitHub" and similar datasets, in order to detect the presence of a fire, this system should be able to detect false results like a static image of a fire like there would be on food carts, t-shirts of people etc. The prediction system once it has completed the process will send the result - 1 for a positive fire, 0 –for no firs detected, the system based on this should send a signal to the used alerting them if the model sends a 1 signal stating that there is a fire that has been detected.

In this proposed system, we are going to utilize the existing hardware such as surveillance cameras, mobile phone cameras etc to capture images and record video for detection. The captured images and videos will be pre-processed and later sent into the proposed model. As this process deals with images and videos we have decided to go with a Convolutional Neural Network (CNN) which is a neural network built for processing images the general structure and process of which has been shown in Fig. No. 5.1.

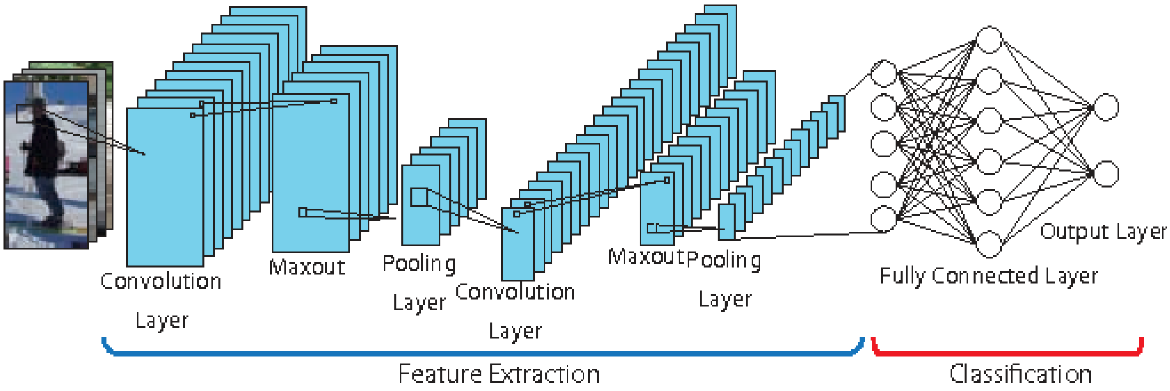


Fig 5.1 Convolutional Neural Network (CNN).

There are models built for specifically object detection and recognition purposes like the one we have used; this is a Convolutional Neural Network (CNN) InceptionV3 model where it is based on GoogleNet architecture. This model is mainly used for image analysis and object detection in images and videos. It contains deeper neural layers while also limiting the number of parameters from growing too large. It also helps in classification of objects or subjects. The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concats, dropouts, and fully connected layers. To all the activation inputs Batch normalization is applied, and the loss is computed via SoftMax.

In this project, we have inculcated this model where it helps in detecting fires in images. The detection of fire in videos is done through frames where each frame is sent into the model and been analysed. If consecutive frames have fire objects or smoke objects, it tries to alert the user that a fire is been detected in the video.

The proposed model is shown in Fig. No. 5.2 and Fig. No. 5.3:

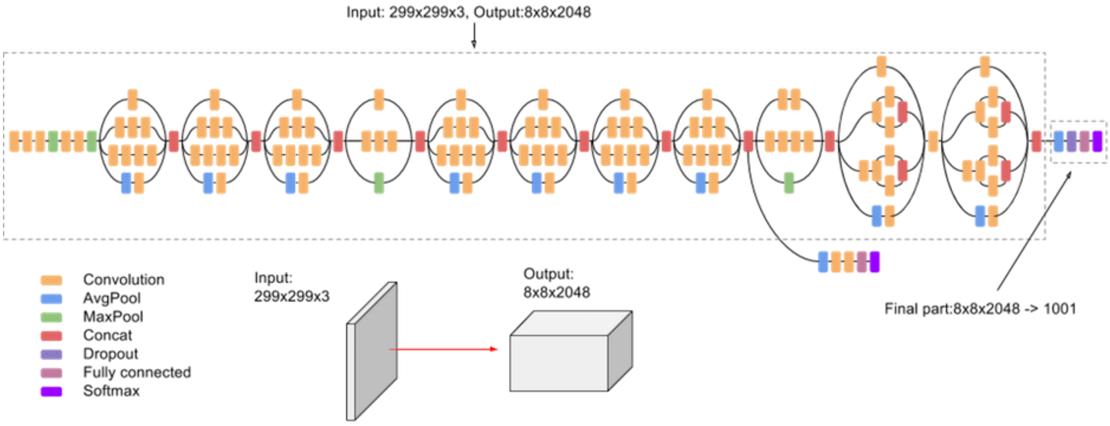


Fig 5.2 Proposed InceptionV3 Model

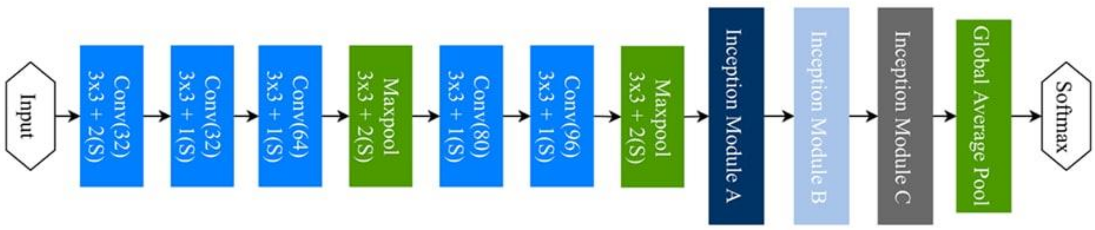


Fig 5.3 InceptionV3 Model Visualisation

5.2 Description

In our proposed project, we try to detect fire based on certain parameters which are some of the conditions in recognising fires. Firstly, we try to give images as input to the model as per the time interval desired by the operator. The images are data pre-processed and been sent to the InceptionV3 deep neural layers for fire detection and analysis. The model is trained in such a way that the fire region is separated and analysed. The analysed information acts as a foundation for learning and re-training the model till it gives out accurate results.

The process is explained in detail in Fig. No. 5.4:

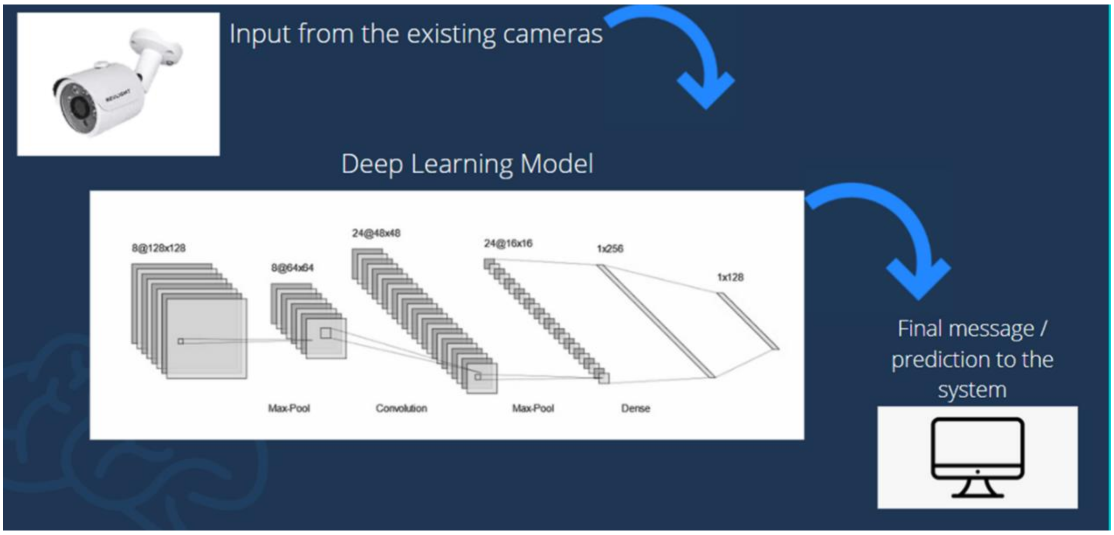


Fig 5.4 Process Flow chart.

For the detection of fire in video captures, we take frames of the video and apply data pre-processing to the image and send it through the InceptionV3 model. The model which is been trained to detect the fires try to detect for the given input frame. The next frame image is been compared to the previous one to learn any changes occurred and analyse the conditions to give out results.

The inception model is explained in detail in the Fig 5.2

In real-time captures, when the fire is been detected the model gives out the live feed in black and white whenever the fire is been detected in the real-time video capture. This alert helps the system operator to act accordingly in the need of the hour.

CHAPTER 6

IMPLEMENTATION

The basic idea behind the project is object detection that can be achieved through many techniques, the most common being CNN or a convolutional neural network but there are many other techniques that have been used in many other projects which solve some limitations of the CNN model and have better features and speed.

In an object detection program that utilises CNN, This problem that arises based on the number of regions that the algorithm divides it into to apply the CNN model if it is huge number of regions which are not Limited which would make the computational cost of the algorithm at the end very high To get through this problem and completely avoid it RCNN was developed which is basically regional CNN which means that the number of regions that the algorithm is divided into is limited to 2000 which means that only in these two thousand regions the CNN model is applied. A comprehensive flow off this would be The first step would be to create the first sub segmentation and create many candidate regions as such to this sub-segments a greedy algorithm is recursively used to combine similar regions into bigger ones The process described above is a selective search algorithm, After which ones we have the 2000 regions these are encapsulated into a square and then fed into a convolutional neural network that produces a feature vector as an output this feature vector is then sent to a SVM what the CNN basically acts as is a feature extractor the SVM then classifies whether the object that we're trying to search for exists in the region proposed or not, now after doing so what the algorithm does is it provides four offset values which change the size or the placement of the bounding region that we have taken initially for example if you are trying to detect a box but we have only been able to detect a corner of it the algorithm knows that once the offset values of certain value are applied the bounding region will cover more of the box. This process can be seen in the Fig. No. 6.1.

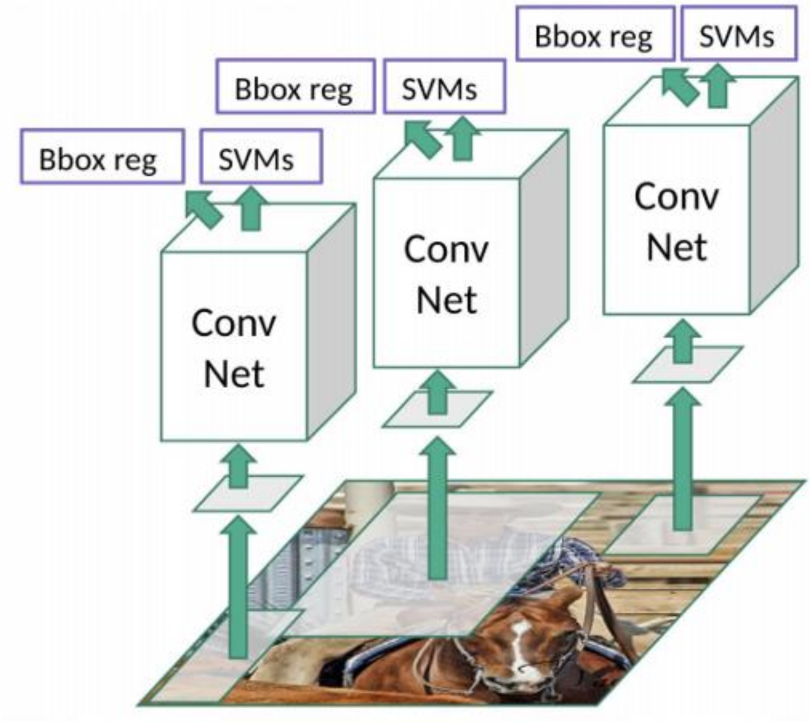


Fig 6.1 RCNN Flow

Although this algorithm improves upon the initial convolutional neural network there are many issues with it such as the amount of time taken to train the network will be high as there 2000 proposed regions per image. The algorithm also cannot be implemented in real-time as the amount of time taken to run the test for each image is at 47 seconds which makes it slow and not good for real time usage. The last issue is that the selective search algorithm that is utilised is a static algorithm as there is no learning happening at that point in the flow which means that there could be generation of bad candidate regions.

To overcome these limitations of the R CNN model fast R CNN and faster R CNN models were introduced an extension of the faster R CNN model is known as mask R CNN which is what we have seen being utilised in the paper [12] Deep Learning-Based Instance Segmentation for Indoor Fire Load Recognition.

The Mask R-CNN structure is based on top of Faster R-CNN. Thus, for a given picture, Mask R-CNN, notwithstanding the class name and bouncing box facilitates for each article, will likewise return the item mask.

We should first rapidly see how Faster R-CNN functions. This will assist us with getting a handle on the instinct behind Mask R-CNN also.

The flow chart of faster R CNN can be seen in Fig. No. 6.2 and the Structure and Flow of The Mask R CNN algorithm can be seen in Fig. No. 6.3 we can see that all those these are similar there are a few steps that differentiate them and giving us the better mask R CNN algorithm.

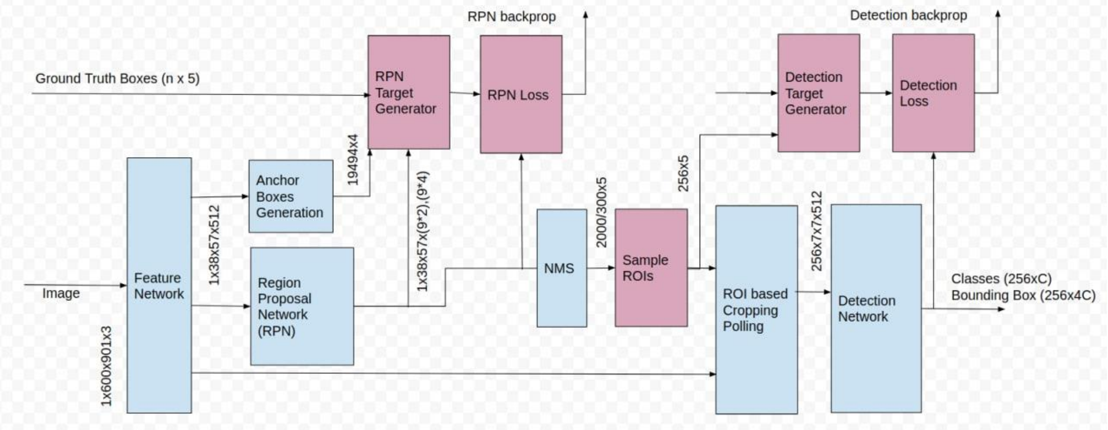


Fig 6.2 Faster RCNN Structure

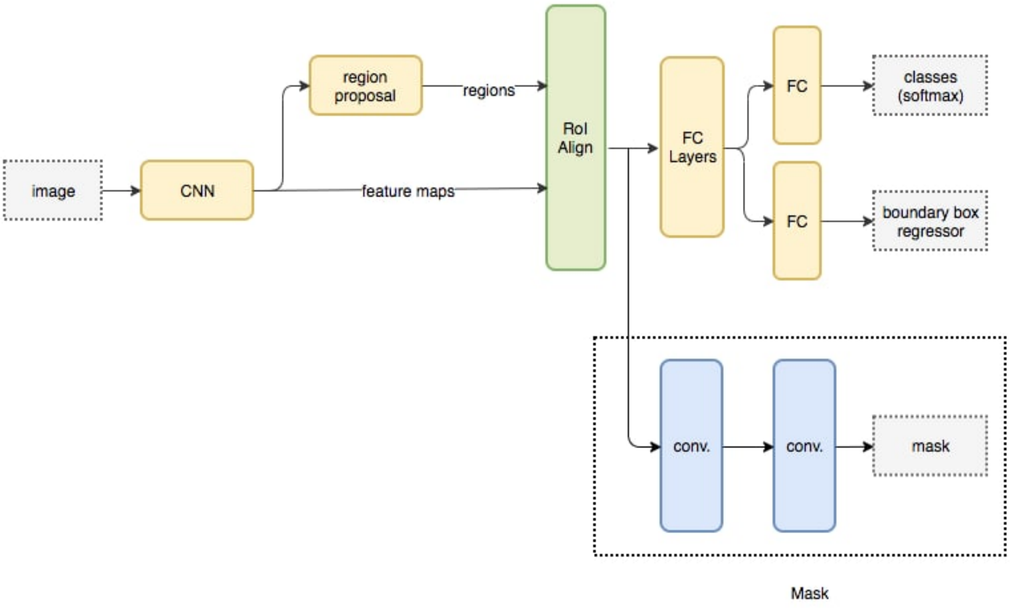


Fig 6.3 Mask RCNN Structure

Faster R-CNN first uses a ConvNet to extricate feature maps from the pictures. These element maps are then gone through a Region Proposal Network (RPN) which returns the applicant bouncing boxes. We then, at that point, apply a RoI pooling layer on these applicant jumping boxes to bring all the possibility to a similar size. Lastly, the proposition are passed to a completely associated layer to order and result the bounding boxes for objects.

Let us take a step-by-step differentiation between the faster R CNN model and The Mask R CNN model the backbone model in faster R CNN is a ConvNet model this is used to extract feature maps from the image the mask R CNN model on the other hand utilizes a resnet 101 architecture to do the same.

The region proposal network takes the feature maps obtained in the last step to apply the RPN on This is the production whether the object is present in the specified region or not we get the regions or the feature maps where the model predicts that the object is present

Pigeon software from the RPN of various shapes and sizes to avoid confusion and to normalise these a pooling layer is applied to them to convert them into the same shape Hollywood film have come to the same shape these are set as an input to a fully connected layer that returns the bounding boxes till now the steps between faster R CNN and mask R CNN are very similar the different step that the latter utilise is the generation of a segmentation mask of the predicted regions and intersection over Union also known as IOU is computed

IoU = Area of the intersection / Area of the union

Now after the calculation of the IOU we compare the values if this IOU value for a particular region is greater than or equal to 0.5 that particular region is considered as a region of interest if the case is not true it means that that particular region is then neglected this is done to all the regions and the regions that satisfy the above-mentioned condition are only selected this is done so that only the good candidate regions are selected.

Now that we have the specific region of interest calculated for regions the segmentation mask is created which differentiate between all the specific objects in the image as seen in Fig. No. 6.4.

Fig 6.4 Segmentation Mask

Just like neural networks there are many other techniques that can be used to detect objects and fires in the case that we have seen in paper [16] Fire Detection and Recognition Optimization Based on Virtual Reality Video Image. Mobile support vector machine algorithm and rough set theory have been used. Basic ideology behind the SVM algorithm is that it can draw a line or a hyperplane between two classes in order to differentiate between them Navratri basic past this the first is a support vector the support vectors are data points that a girl closely to the hyperplane these will define that hyperplane by calculating the margins margin is the gap between the closest class points it is calculated as the shortest distance from the plane to the closest points are the support vectors if this March in is higher in value it is said to be a good margin the smaller or the lesser this value is the was the margin is hyperplane is a differentiated plane that separates in this case too two classes similar to a regression algorithm drawing a line to differentiate the maximum possible number of data points which also the SVM algorithm is capable of with hyperplane will divide the support vectors such that they will fall clearly between two classes in either of them and non-overlapping. The basic working of the SVM algorithm can be seen in Fig. No. 6.5

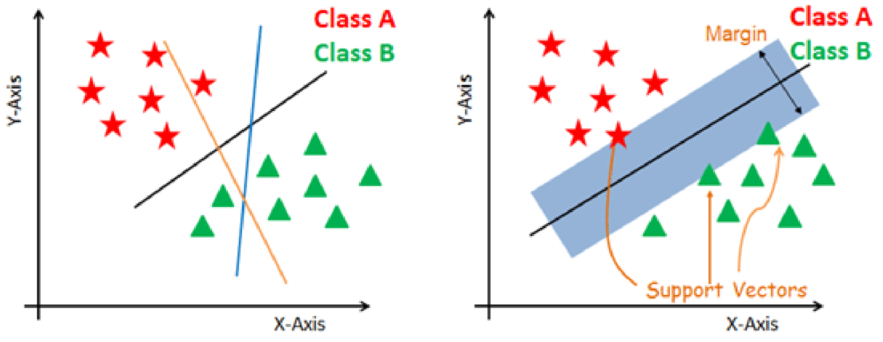


Fig 6.5 SVM Differentiating Between Classes

The basic structure and flow of the SVM algorithm can be seen in the Fig No. 6.6

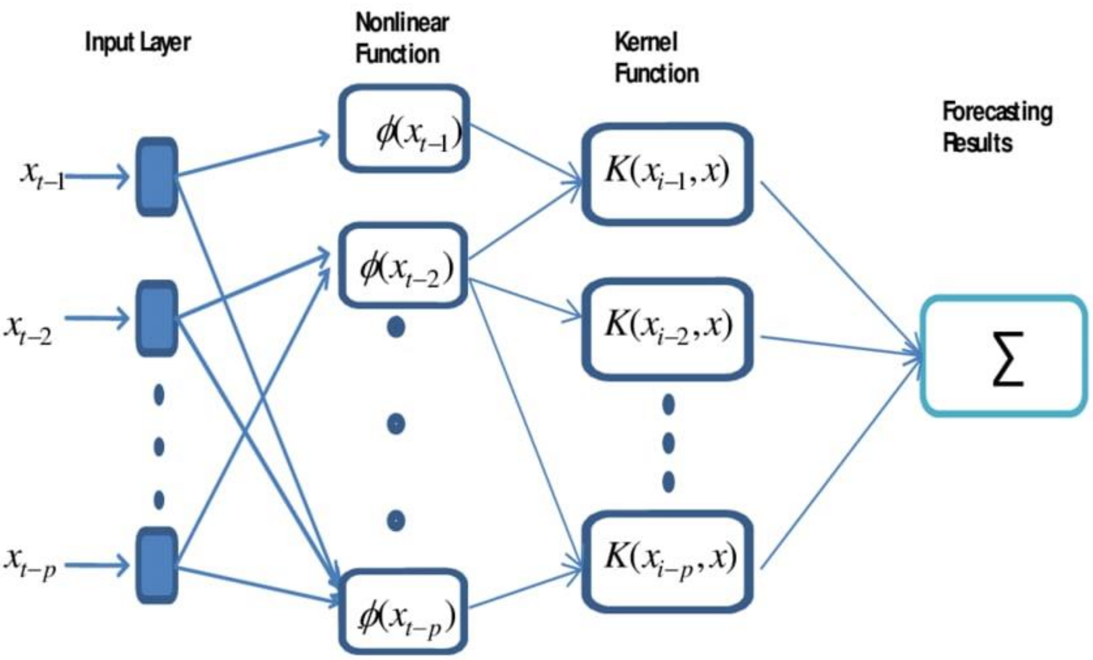


Fig 6.6 SVM Structure

Rough set Theory is used in combination with the support vector machine algorithm in order to make the job easier for the SVM algorithm this is done through the attribute reduction algorithm in R S theory has given birth to Rs SVM classifier model the working of which can be seen in Fig. No. 6.7.

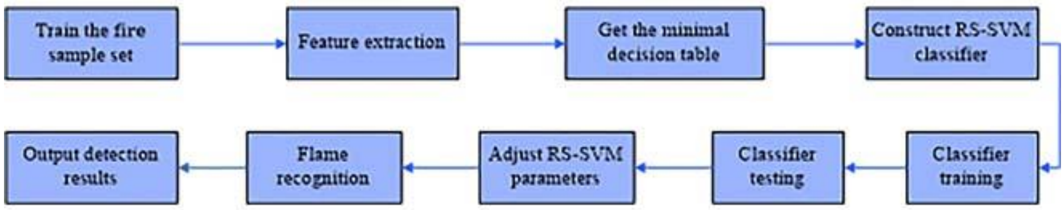


Fig 6.7 RS SVM Algorithm and Flow for Fire Detection

Another method of object detection is using a LSTM network or long short-term memory. The basic idea behind and LSTM is of remembering a bit of the previous knowledge this is achieved by using a concept called as Gates or an activation layer that basically tries to remember the past knowledge that the model has been given on the previous input if that data is of importance. Another parameter known as a vector is maintained in order to retain what has been remember from the previous knowledge this concept of gates utilizes these:

- **Forget Gate(f):** It determines to what extent to forget the previous data.
- **Input Gate(i):** It determines the extent of information be written onto the Internal Cell State.
- **Input Modulation Gate(g**): Although not much documentation mentions this Gate, whether even mention it to be a part of the input Gate itself. the idea behind this gate is that it affects the data is written to the internal cell state in order to make it much more efficient and faster this is done by adding non linearity to the information and making that information have a zero mean this is important as zero main data has a faster convergence although this may not be as important compared to the other Gates it does affect the speed and efficiency of the algorithm.

- **Output Gate(o):** It determines what output (next Hidden State) to generate from the current Internal Cell State.

The process of detection of fire can be achieved using multiple techniques like various types of neural networks of which one is convolution neural networks that are utilised very often another setwork is a recurrent neural network though they have a problem of short-term memory to solve this LSTM network is introduced like we have seen in paper [11] "*Multistage Real-Time Fire Detection Using Convolutional Neural Networks and Long Short-Term Memory Networks*" which solves the limitation of Recurrent neural networks system.

Working of an LSTM recurrent unit:

Take input the current input, the previous hidden state, and the previous internal cell state. Calculate the values of the four different gates by following the below steps:

- For each gate, calculate the parameterized vectors for the current input and the previous hidden state by element-wise multiplication with the concerned vector with the respective weights for each gate.
- Apply the respective activation function for each gate element-wise on the parameterized vectors. Below given is the list of the gates with the activation function to be applied for the gate.
- Calculate the current internal cell state by first calculating the element-wise multiplication vector of the input gate and the input modulation gate, then calculate the element-wise multiplication vector of the forget gate and the previous internal cell state and then adding the two vectors.

c_{t} = i\o.g + f\o.c_{t-1}

- Calculate the current hidden state by first taking the element-wise hyperbolic tangent of the current internal cell state vector and then performing element-wise multiplication with the output gate.

The process flow mentioned above can be seen in Fig. No. 6.8 and in Fig. No 6.10 and the final structure showing the single cell as a part of a final neural network in Fig. No. 6.9.

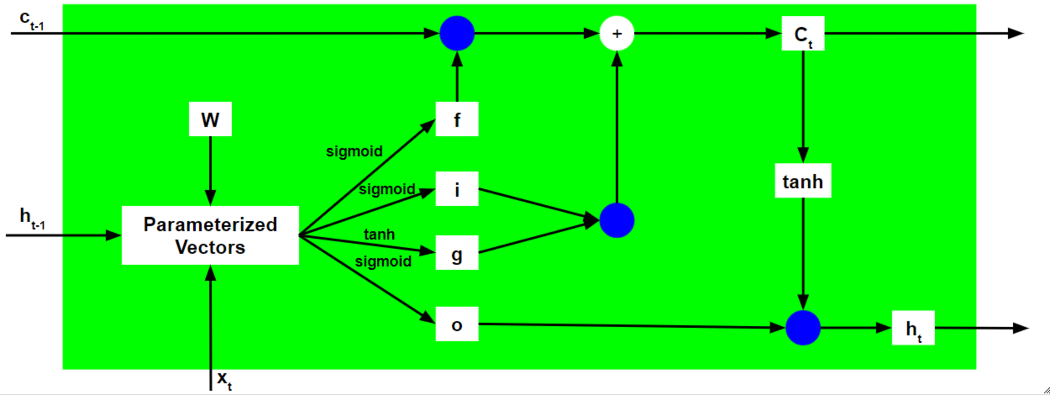


Fig 6.8 LSTM Recurrent Unit Cell

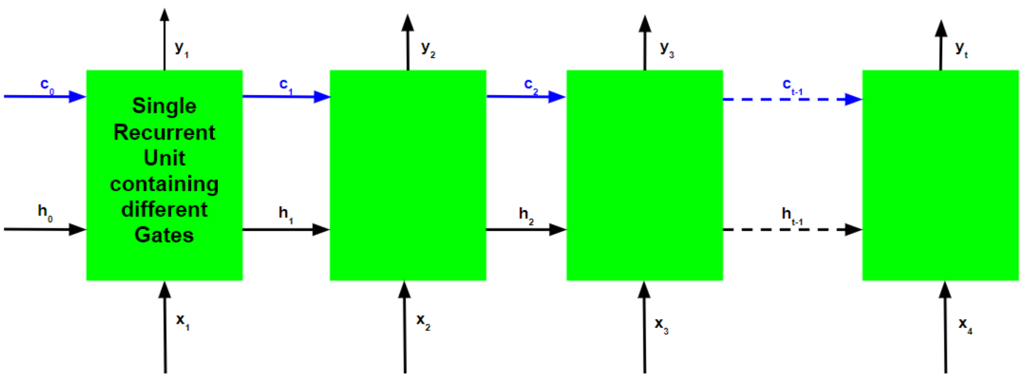


Fig 6.9 LSTM Recurrent Unit Cell as part of flow

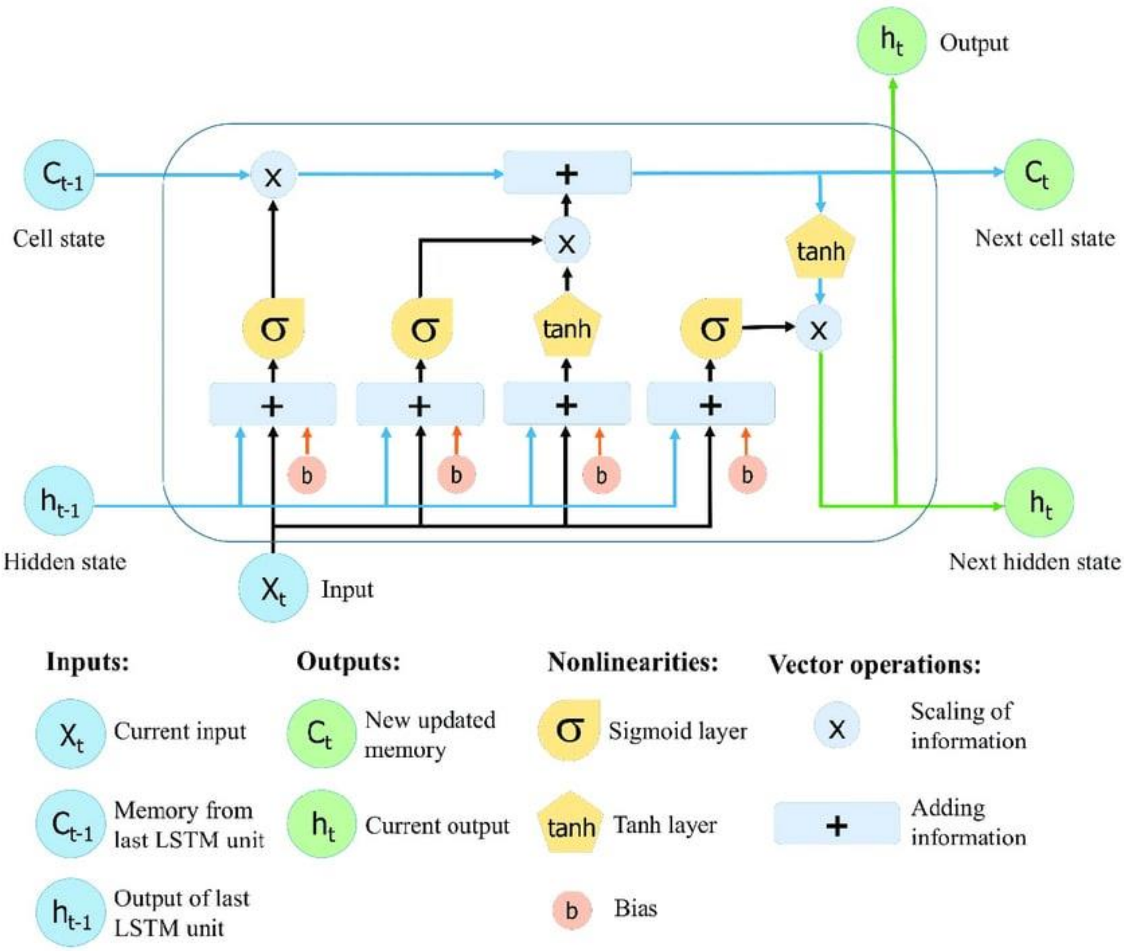


Fig 6.10 LSTM Recurrent Unit Cell

As discussed earlier the CNN model is the most common model that is used for image processing and object detection which is also the first model that we have used even the second model although more complicated is still a modified convolutional neural network based on the googlenet architecture the steps to implement these are explained below

6.1. Data Pre-processing:

Once all the images are read into the python file, using data augmentation, more samples are generated using horizontal flip, rotations etc., and all the images are re-scaled to 224x224 size. The images are classified into 2 classes and are split into training and validation.

6.2. CNN model:

The first part of the problem is approaching the CNN model.

An image is a pixel matrix which when applied on CNN the convolutional layer a matrix of its own smaller than the initial image size sweeps over the image to fill a feature map by the dot product of the convolutional layer value and image matrix value. The sweeps change position based on stride size. On the feature map which was produced, max pooling is used where in the filter layer goes over the feature map based on stride value as before and choses the maximum value that exists in the feature map covered by the filter.

On this matrix that has been produced after max pooling, activation functions such as ReLu are used where in for example ReLu takes non-linear data and converts it into 1 or 0 based on x>0. This is now sent to the fully connected layer where the feature output is flattened and SoftMax activation is used on them.

The flow of this is represented in Fig. No. 6.11:

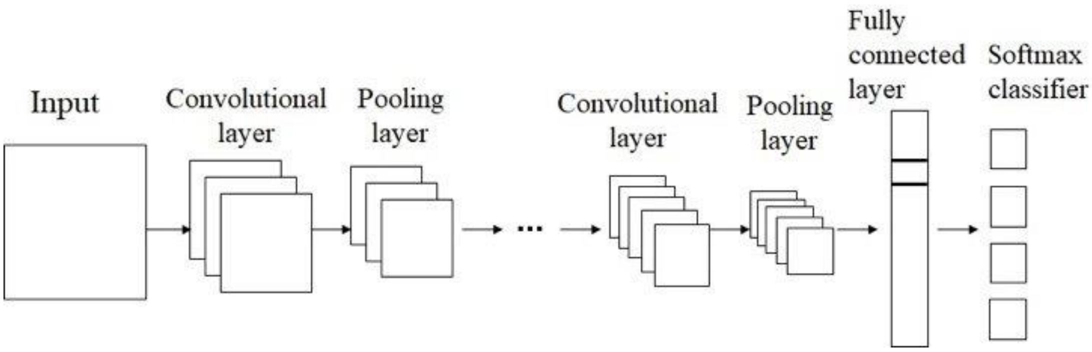


Fig 6.11 CNN Flow Model

Using tensorflow and keras, a structure for CNN was designed which can be seen in Fig. No. 6.1.1. The final activation is used is SoftMax. The loss is calculated using cross entropy and Adam optimizer is used. The input shape is designed as according to pre-processing is done to the images as mentioned before. Max-pooling is used all throughout, there are 3 fully connected dense layers. The model is trained with 980 images split into two classes with 50 epochs, 15 steps for each epoch and 15 validation steps. This at the end achieves an accuracy of 96.4% and loss of 0.09%. The loss and accuracy graphs are mentioned for both training and validation.

6.3. InceptionV3 model:

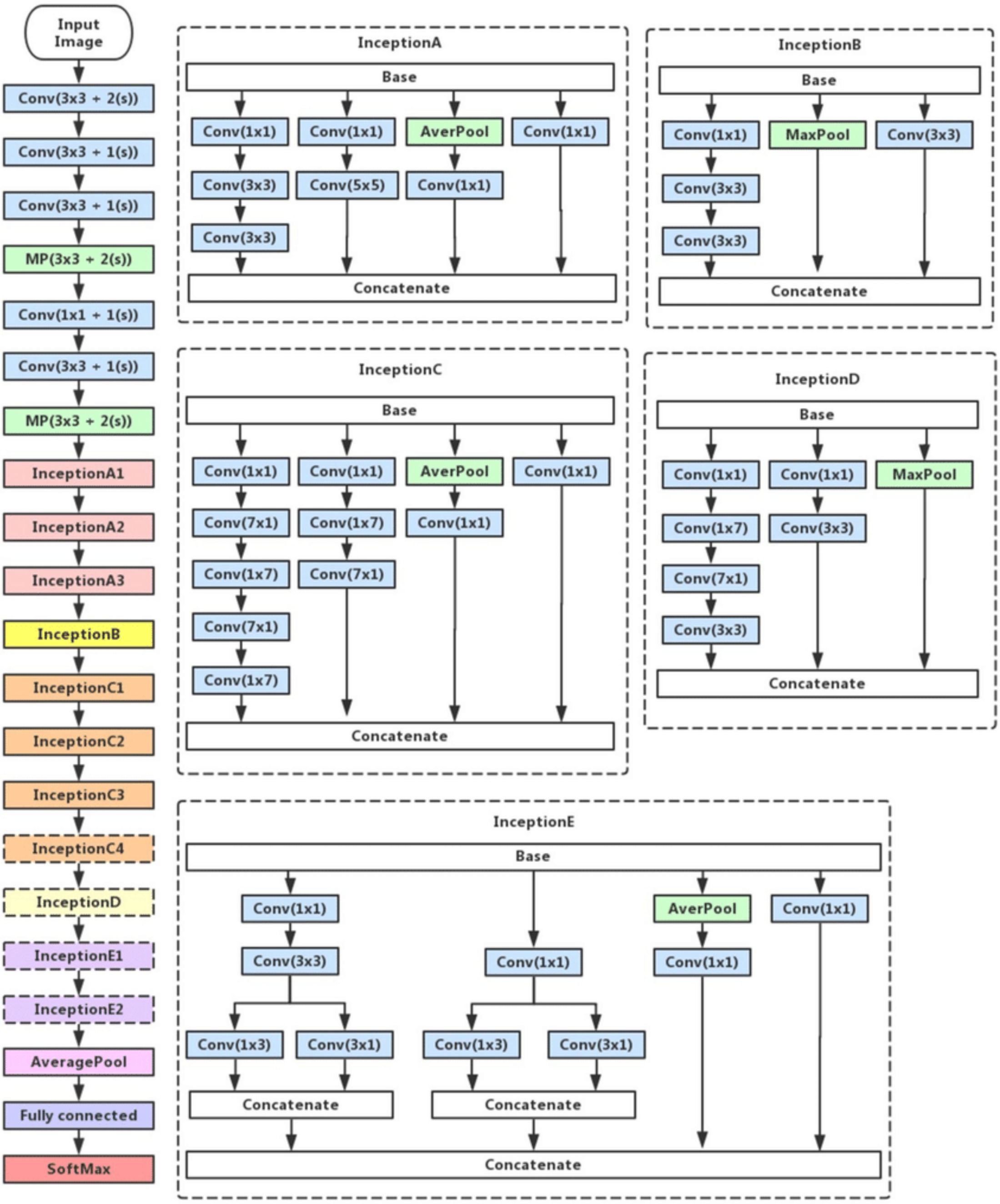


Fig 6.12 Detailed InceptionV3 Model Architecture

Inception-v3 is a convolutional neural network architecture from the Inception family that makes several improvements including using Label Smoothing, factorized 7 x 7 convolutions, and the use of an auxiliary classifier to propagate label information lower down the network, it is 48 layers deep.

The architecture of an Inception v3 network is progressively built, step-by-step, as explained in Fig. No. 6.12.

1. Factorized Convolutions: this helps to reduce the computational efficiency as it reduces the number of parameters involved in a network. It also keeps a check on the network efficiency.

2. Smaller convolutions: replacing bigger convolutions with smaller convolutions definitely leads to faster training. Say a 5 × 5 filter has 25 parameters; two 3 × 3 filters replacing a 5 × 5 convolution has only 18 (3*3 + 3*3) parameters instead.

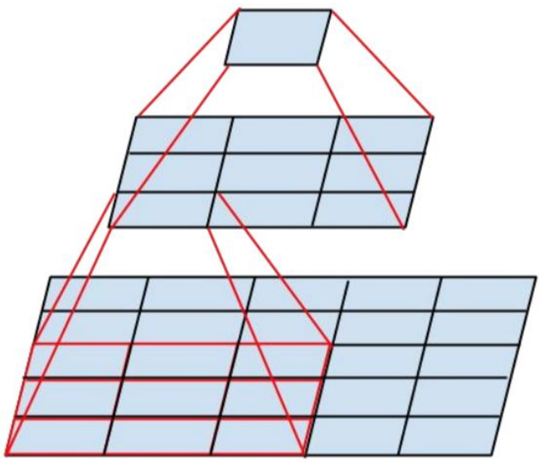


Fig 6.13 Smaller Convolutions

3. Asymmetric convolutions: A 3 × 3 convolution could be replaced by a 1 × 3 convolution followed by a 3 × 1 convolution. If a 3 × 3 convolution is replaced by a 2 × 2 convolution, the number of parameters would be slightly higher than the asymmetric convolution proposed.

Fig 6.14 Asymmetric Convolutions

4. Auxiliary classifier: an auxiliary classifier is a small CNN inserted between layers during training, and the loss incurred is added to the main network loss. In GoogleNet auxiliary classifiers were used for a deeper network, whereas in Inception v3 an auxiliary classifier act as a regularize.

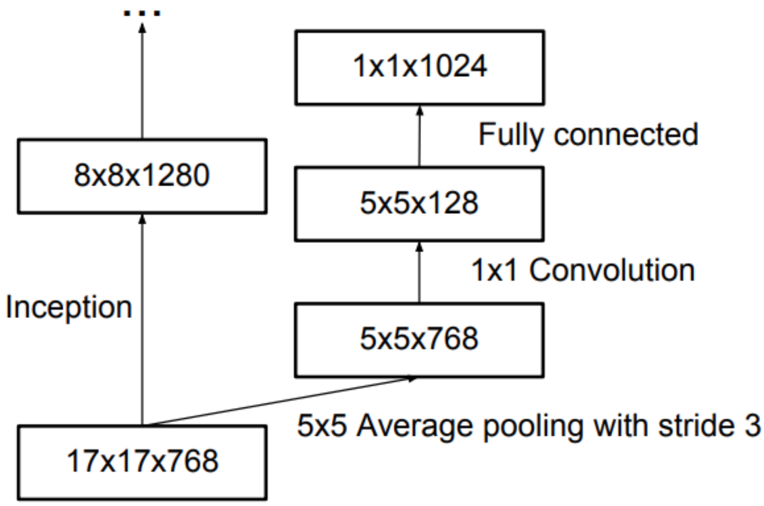


Fig 6.15 Auxiliary Classifier

5. Grid size reduction: Grid size reduction is usually done by pooling operations. However, to combat the bottlenecks of computational cost, a more efficient technique is proposed

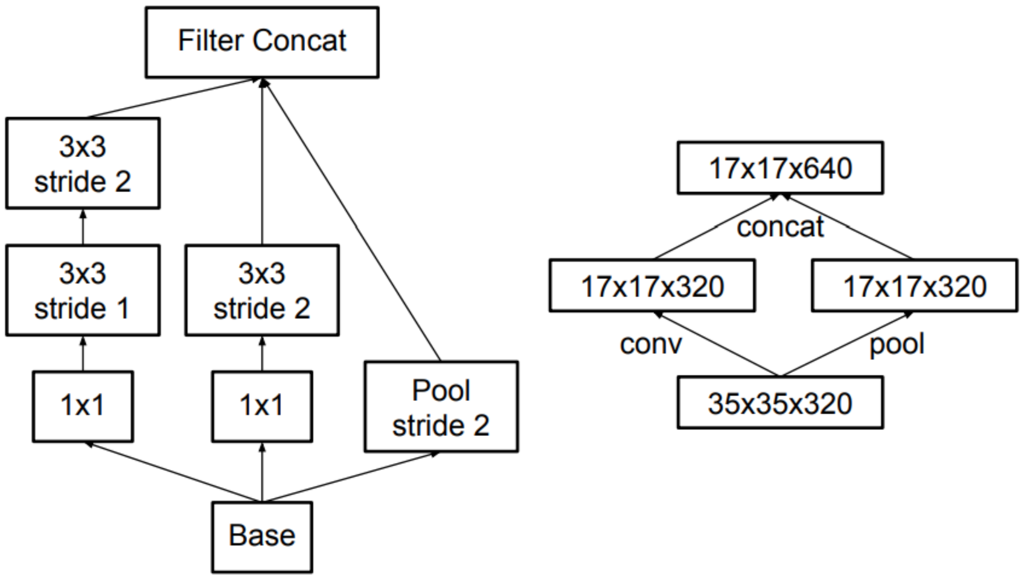


Fig 6.16 Grid Size Reduction

After all the data has been read into the Python file, out of 2000 images, 1800 of them as training and 200 are set into validation. Using tensorflow and keras, we can import the InceptionV3 model from GoogleNet and then we can import dense, Global average pooling, Input and dropout layers from keras dot layers package and then define the input shape of the input tensor based on the previous data pre-processing that has been done to images which will give us and size of 224 X 224.

We add a global spatial average pooling layer and then using the RMS prop optimizer and cross entropy for calculation of loss we ran this model for 20 epochs. When the model ran with the validation set for 10 epochs for cross validation leaving us with the final accuracy of 96.8 8% and a loss value of 0.1 %. The loss value and accuracy values for both training and validation can be seen in figure numbers.

6.4. Working of Model with Video Captures:

Now that we have a trained deep learning model the next step is to test whether this model works well in a real time environment, whether it will be fast enough in terms of the detection time and whether it can handle videos. After the training of the deep learning Inception V3 model has been completed we save it locally so that this pretrained model can be used at any time.

Using the OpenCV, NumPy, PIL, Tensorflow and Keras processing packages, we can use the pre-trained InceptionV3 model that has been stored locally to test the model in real time. We load the pre trained InceptionV3 model into a Python file and using OpenCV video capture function. For the testing purposes for the algorithm utilising the existing camera equipment available to us which in this case is a laptop webcam for the video input device. Video input goes through the same data preprocessing as the original images for the training data set, that is they are shaped into a 224 by 224 frame in the RGB colour space.These Vishay films are now sent to the pre trained Inception V3 model for the production process.

once the frame has been processed the prediction is sent into a list if this production is returning a value of zero as an output which means that the object detection algorithm has detected the presence of a fire in that particular frame that frame is then reproduced on the screen of the user in a black and white or grey scale colour space although this is not the end alarm or alert that is set to the user in the complete model this is just to show whether the The Fire has been detected and when the user is shown the continuous video stream output the amount of time it takes for the fire to be detected can also be seen. Return value is one that is a file has not been detected in that frame that claim will directly be shown in the same colour space as it has been taken him which is the RGB colour space directly without any changes

CHAPTER 7

RESULTS

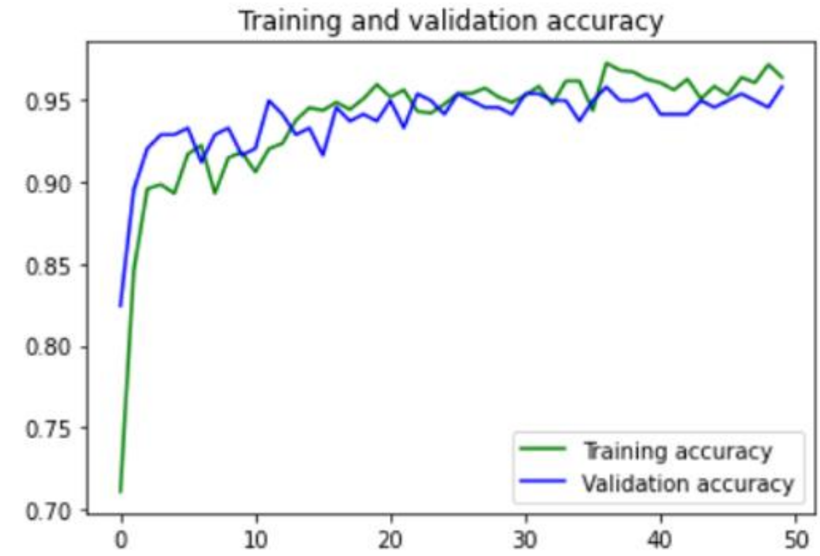
The InceptionV3 model has been constructed using the TensorFlow and keras frameworks which allows to later save it as a model for real-time detection. The model summary is shown below:

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 54, 54, 96)        34944
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 26, 26, 96)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 22, 22, 256)       614656
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 10, 10, 256)       0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 6, 6, 384)         2457984
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 2, 2, 384)         0
_________________________________________________________________
flatten (Flatten)            (None, 1536)              0
_________________________________________________________________
dropout (Dropout)            (None, 1536)              0
_________________________________________________________________
dense (Dense)                (None, 2048)              3147776
_________________________________________________________________
dropout_1 (Dropout)          (None, 2048)              0
_________________________________________________________________
dense_1 (Dense)              (None, 1024)              2098176
_________________________________________________________________
dropout_2 (Dropout)          (None, 1024)              0
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 2050
=================================================================
```

Fig 7.1 InceptionV3 Model Summary

After the Construction the model is been run for training till 20 epochs with 14 steps for each epoch. When it comes to validation of the model, it is been kept through 10 epochs to validate and learning rate as 0.0001 with momentum as 0.9.

The final training and validation accuracy has been tracked and made into a representational graph.



<Figure size 432x288 with 0 Axes>

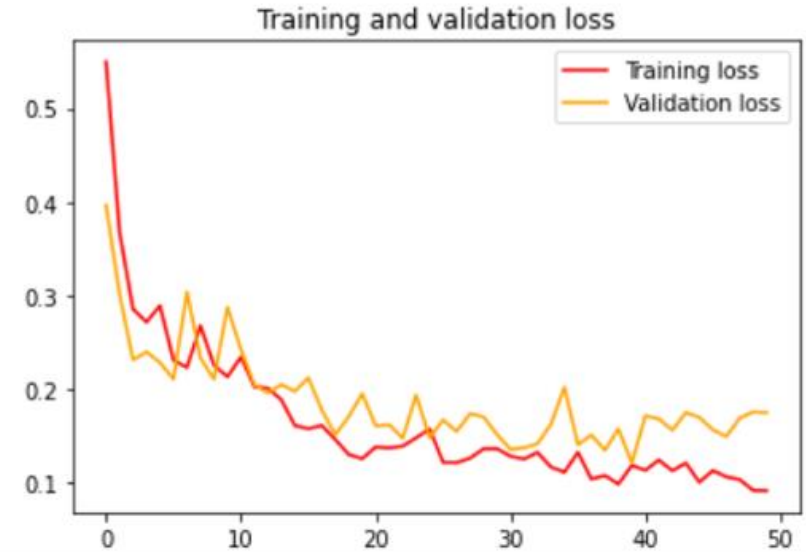


Fig 7.2 Training and Validation accuracy-loss graphs

Real-Time Detection:

In real time detection, the model which is saved is been deployed into a video capture phase with the help of OpenCV. This where the results generated under different conditions.

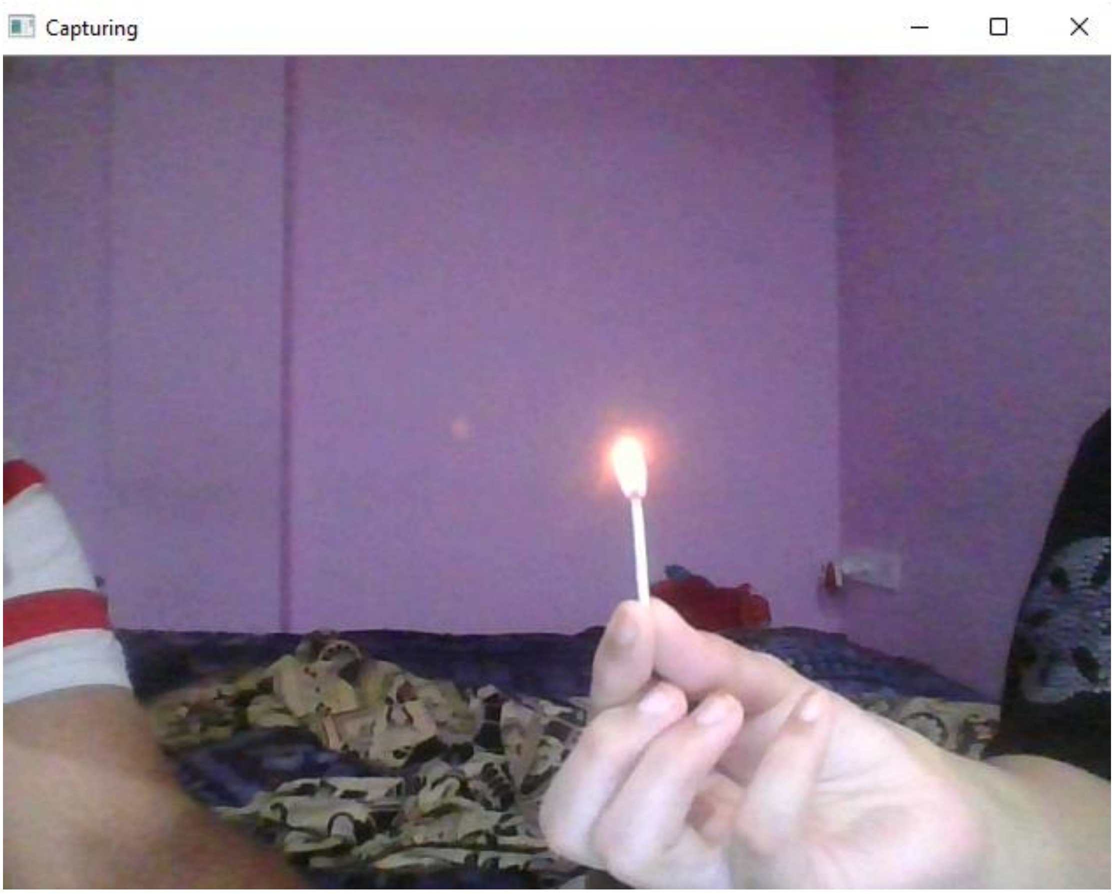


Fig 7.3 Fire Introduction in Real-Time

Now as the fire is been introduced, the model then starts to detect the fire region in the frame and also learn in real-time. When the fire is detected, the frame is turned in black and white indicating that fire has been detected.

The indication starts to blink till if there is no fire in the frame. The resulting condition can be seen in Fig. No. 7.4:

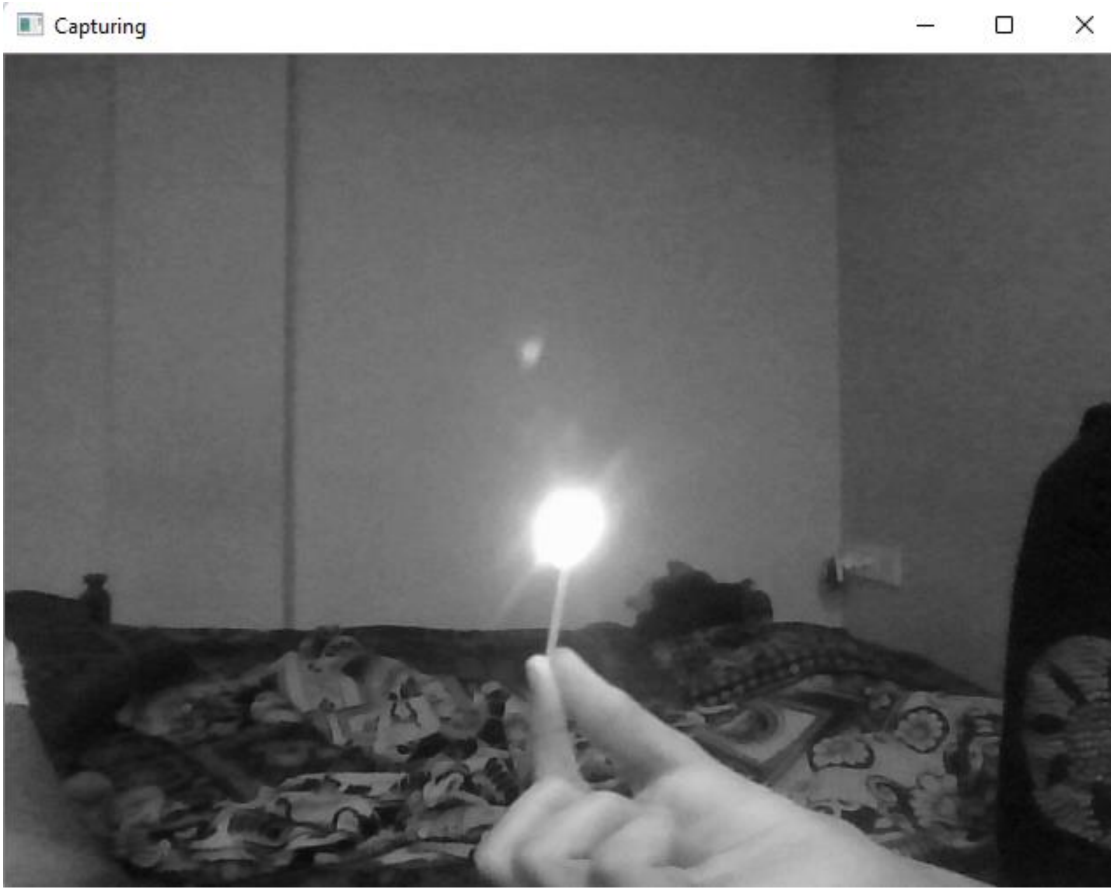


Fig 7.4 Frames turn into Black & White when fire is Detected.

When we try to introduce a life-like image of fire through a smartphone, we can see that the model is able to distinguish the real fire and image of a fire. The resulting condition can be seen below.

From this we can know that the model is robust when it comes to fake fire detections. Many fire systems are not able to make out the fake fires and trigger itself and cause havoc in the premises.

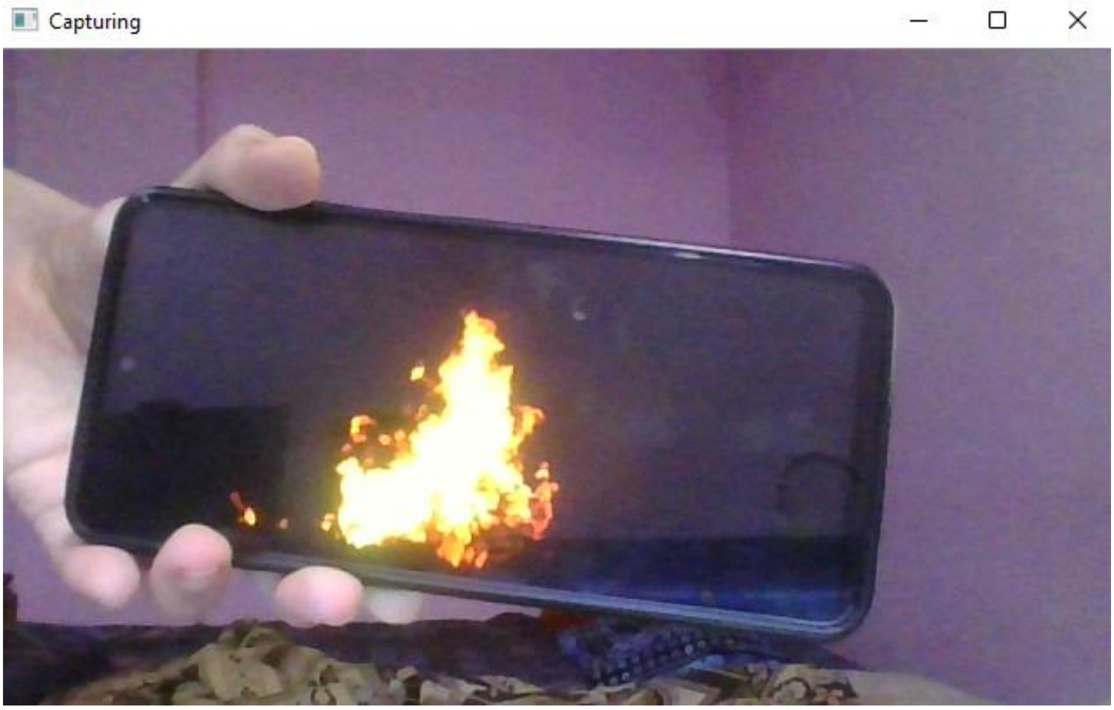


Fig 7.5 Result of the Fake Fire Recognition

The image tests done on the InceptionV3 model gave an accuracy of 99.74% when been given a frame of the fire video capture.

When it comes to video capture testing, the consecutive frames are been taken and analysed. If a still frames are been recognised with no luminosity or motion, the model determines it as a fake fire.

CHAPTER 8
CONCLUSION AND FUTURE WORK

Conclusion

As we have seen the CNN model although having high accuracy and low loss percentage is not exactly reliable in situations as the data set it has been tested upon is very specific and not general use where it has many drawbacks like not being able to detect outdoor fires and has no false detection systems. The speed of the system is also so not that fast making it not a very good solution for a large-scale system. Adding more communication layers and more models to this existing model can improve these things.

Overall the inception V3 model although it will look like it has similar accuracy and similar loss percentage it is more reliable as it has been trained upon a very data set and also has the capability of distinguishing between indoor and outdoor fires all the even this model does not have false detection this model takes into account luminosity which does provide a support to the fault detection system where if it is a static image of a fire in an urban area we can see the results as mentioned in figure number it is also able to process the video at 12 frames per second which is faster than what the CNN model mentioned about is capable of the drawbacks and the solutions to these drawbacks have been discussed in the future work below.

Future Work

The system that is now in completion has a few issues with it, to begin naming them, we have:

The datasets that have been used to train the model are highly specific meaning that the model works with its high accuracy prediction score in certain situations only , this can be solved by the method mentioned in the proposed work of using our existing network of video capturing devices to create a accustom dataset for specific areas so that the system is now trained in the situations that happen most often in the area that is located in which makes issues like area specific climate and weather condition patterns that affect fire and smoke to be included as a extracted feature in the model eliminating confusion.

The next issue comes in increasing the rate of processing of the video inputs the current model goes through at a speed of 12 frames per second increasing this value will make one half of the pipeline faster the other half is the processing speed of the model adding layers or using updated models with new features can reduce this bottleneck. The current model also takes in to account luminance level in order to detect fires and general shape this causes false positive prediction where something that looks similar to fires is also recognised as fires, using techniques such as motion detection and luminance of the detected shape or updating newer models such as other inception models or combining them all in a parallel pipeline will solve this issue.

The process of connecting the existing camera network to the prediction model has to be more streamlined and made more user friendly and all the fore mentioned points are to be made automatic in terms of updating and quality of life for the user who uses this automated system.

REFERENCE

[1] Sheng, Dali, Jinlian Deng, and Jiawei Xiang. "Automatic Smoke Detection Based on SLIC-DBSCAN Enhanced Convolutional Neural Network." *IEEE Access* 9 (2021): 63933-63942.

[2] Zhang, Jianmei, et al. "ATT Squeeze U-Net: A Lightweight Network for Forest Fire Detection and Recognition." *IEEE Access* 9 (2021): 10858-10870.

[3] Celik, Turgay. "Fast and efficient method for fire detection using image processing." *ETRI journal* 32.6 (2010): 881-890.

[4] Poobalan, Kumarguru, and Siau-Chuin Liew. "Fire detection algorithm using image processing techniques." *Proceedings of the 3rd International Conference on Artificial Intelligence and Computer Science (AICS2015)*. 2015.

[5] Vipin, V. "Image processing-based forest fire detection." *International Journal of Emerging Technology and Advanced Engineering* 2.2 (2012): 87-95.

[6] Mahmoud, Mubarak Adam Ishag, and Honge Ren. "Forest fire detection and identification using image processing and SVM." *Journal of Information Processing Systems* 15.1 (2019): 159-168.

[7] Mahmoud, Mubarak AI, and Honge Ren. "Forest fire detection using a rule-based image processing algorithm and temporal variation." *Mathematical Problems in Engineering* 2018 (2018).

[8] Angayarkkani, K., and N. Radhakrishnan. "Efficient forest fire detection system: a spatial data mining and image processing-based approach." *International Journal of Computer Science and Network Security* 9.3 (2009): 100-107.

[9] Yuan, Feiniu, et al. "Non-linear dimensionality reduction and Gaussian process based classification method for smoke detection." *IEEE Access* 5 (2017): 6833-6841.

[10] Nguyen, Manh Dung, et al. "Multistage Real-time Fire Detection using Convolutional Neural Networks and Long Short-term Memory Networks." IEEE Access (2021).

[11] Zhou, Yu-Cheng, et al. "Deep Learning-Based Instance Segmentation for Indoor Fire Load Recognition." IEEE Access 9 (2021): 148771-148782.

[12] Xu, Zhaoyi, Yanjie Guo, and Joseph Homer Saleh. "Advances Toward the Next Generation Fire Detection: Deep LSTM Variational Autoencoder for Improved Sensitivity and Reliability." IEEE Access 9 (2021): 30636-30653.

[13] Rashkovetsky, Dmitry, et al. "Wildfire Detection from Multisensor Satellite Imagery Using Deep Semantic Segmentation." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021): 7001-7016.

[14] Yang, Hunjun, et al. "Non-temporal lightweight fire detection network for intelligent surveillance systems." IEEE Access 7 (2019): 169257-169266.

[15] Huang, Xinchu, and Lin Du. "Fire detection and recognition optimization based on virtual reality video image." IEEE Access 8 (2020): 77951-77961.

[16] Lin, Zhengyang, et al. "A contextual and multitemporal active-fire detection algorithm based on FengYun-2G S-VISSR data." IEEE Transactions on Geoscience and Remote Sensing 57.11 (2019): 8840-8852.

[17] Kim, Young-Jin, et al. "Trustworthy Building Fire Detection Framework With Simulation-Based Learning." IEEE Access 9 (2021): 55777-55789.

[18] Xie, Yakun, et al. "Efficient video fire detection exploiting motion-flicker-based dynamic features and deep static features." IEEE Access 8 (2020): 81904-81917.

[19] Cheng, Caixia, Fuchun Sun, and Xinquan Zhou. "One fire detection method using neural networks." Tsinghua science and technology 16.1 (2011): 31-35.

[20] Xu, Zhaoyi, Yanjie Guo, and Joseph Homer Saleh. "Tackling Small Data Challenges in Visual Fire Detection: A Deep Convolutional Generative Adversarial Network Approach." IEEE Access 9 (2020): 3936-3946.

[21] Chaoxia, Chenyu, Weiwei Shang, and Fei Zhang. "Information-guided flame detection based on faster r-cnn." IEEE Access 8 (2020): 58923-58932.

[22] Nakip, Mert, Cüneyt Güzelíş, and Osman Yildiz. "Recurrent Trend Predictive Neural Network for Multi-Sensor Fire Detection." IEEE Access (2021).

[23] Jiecheng, Lu, et al. "Method of fire image identification based on optimization theory." Journal of Systems Engineering and Electronics 13.2 (2002): 78-83.

[24] Graff, Casey A., et al. "Forecasting Daily Wildfire Activity Using Poisson Regression." IEEE Transactions on Geoscience and Remote Sensing 58.7 (2020): 4837-4851.

[25] Cao, Yichao, et al. "An attention enhanced bidirectional LSTM for early forest fire smoke recognition." IEEE Access 7 (2019): 154732-154742.

[27] Kim, Byoungjun, and Joonwhoan Lee. "A video-based fire detection using deep learning models." Applied Sciences 9.14 (2019): 2862.