COMPUTER VISION-BASED EARLY FIRE DETECTION USING MACHINE LEARNING

A PROJECT REPORT

Submitted by

ISHWARIYA.P	**412818205021**
LAKSHMI PRABHA.R	**412818205034**
MYTHIREYENI.P.V	**412818205042**
NEELAVENI.M	**412818205045**

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

SRM VALLIAMMAI ENGINEERING COLLEGE
(AN AUTONOMOUS INSTITUTION)

SRM NAGAR, KATTANKULATHUR,

KANCHEEPURAM

ANNA UNIVERSITY: CHENNAI 600 025

MARCH 2022

ANNA UNIVERSITY: CHENNAI-600025

BONAFIDE CERTIFICATE

Certified that this project report **"COMPUTER VISION-BASED EARLY FIRE DETECTION USING MACHINE LEARNING"** is the bonafide work of **"ISHWARIYA.P, LAKSHMI PRABHA.R, MYTHIREYENI.P.V AND NEELAVENI.M"** who carried out the project work under my supervision.

Dr. A.R. REVATHI	**Mr. M.ASAN NAINAR**
HEAD OF DEPARTMENT	SUPERVISOR
Associate Professor	Assistant Professor
Department of Information Technology	Department of Information Technology
SRM Valliammai Engineering College (AN AUTONOMOUS INSTITUTION) SRM Nagar, Kattankulathur Kancheepuram - 603 203.	SRM Valliammai Engineering College (AN AUTONOMOUS INSTITUTION) SRM Nagar, Kattankulathur Kancheepuram - 603 203.

Submitted for the viva voce held on ………………………

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

First and foremost, we would like to extend our heartfelt respect and gratitude to Management, Principal **Dr. B. Chidhambara Rajan, M.E., Ph.D.,** and Vice Principal **Dr. M. Murugan, M.E., Ph.D.,** who helped us in our endeavors.

We also extend our heartfelt respect to our beloved Head of the Department **Dr. R. Meenakshi M.E., Ph.D.,** for offering her sincere support throughout the project work.

We thank our Project Coordinator(s) **Dr. A. R. Revathi, B.E., M.Tech., M.B.A.,Ph.D. Associate Professor** and **Dr. S. Ravikumar, M.E., Ph.D. Assistant Professor (Selection Grade)** for their consistent guidance and encouragement throughout the progress of the project.

We thank our Project guide **Dr. M. ASAN NAINAR Assistant Professor** for her valuable guidance, support and active interest for the successful implementation of the project.

We would also thank all the **Teaching and Non-Teaching staff members** of our department for their constant support and encouragement throughout the course of this project work.

Finally, the constant support from our lovable Parents and Friends is untold and immeasurable.

ABSTRACT

The project aimed to detect fire by using the image processing technology will alert people by early detection of fire. The automatic fire alarm systems already existed the sensor method has limitations and designed to sense fire with the smoke, limited areas. To reduce limitations and to optimize with present technology, Computer Vision Based Early Fire Detection using machine learning is proposed. The implementation is done by using pycharm IDE and to connect the webcam as hardware. Webcam is taken as an input source, which captures the video feed from the surrounding and feeds into the system for analysis. The code is written in python language using the open CV library for image processing. The theoretical parts emphasize more in computer vision, machine learning, image processing, color model, and the working algorithm of the project to detect the fire. The project use algorithms such as Haar Cascade classifier, adaptive boosting and integral image processing, which is used in the object detection through the image or video. Using an integral image, the algorithms rapidly calculate summations over image sub regions. Integral images facilitate summation of pixels and can be performed in constant time, regardless of the neighborhood size. Adaptive boosting is a statistical classification meta-algorithm, used to boost the performance of any machine learning algorithm.

Keywords: Fire detection, Image processing, Color model, Alarm, Notification mail.

TABLE OF CONTENT

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Computer Vision based fire detection using image processing has the potential to be useful in conditions in which conventional methods cannot be adopted. The fire detection algorithm uses visual characteristics of fires like brightness, color, spectral texture, spectral flicker, and edge trembling to discriminate them from other visible stimuli. There are various fire detection techniques such as infrared sensor, a thermal detector, smoke detector, flame detector, and optical smoke detector. These methods are not always reliable as they do not always detect the fire itself but detect one or more phenomena resulting from fire, such as smoke, heat, infrared, ultraviolet light radiation or gas, which could be produced in other ways and hence, produces many false alarms. By the help of computer vision and image processing techniques, it is possible to get better results than conventional systems because images can provide more reliable information. In recent times, research on detection of flame and smoke using surveillance cameras with machine vision has gained momentum. The image processing approach involves the extraction of the smoke-plume or flame from the background by using frame difference technologies. In the case of the segmentation of fire features, color processing scores over gray-scale processing. Color processing can avoid the generation of false alarms due to variations in the lighting conditions, e.g. natural background illumination, better than gray-scale processing. Further, a video camera is a volume sensor, and potentially monitors a larger area. The traditional point sensor looks at a point in space. Since the point sensor may not be affected by smoke or flame, fire would be undetected. However, vision-based flame and smoke detection still has great technical challenges, since flame and smoke are non-rigid objects, with none of the primitive image features and variability in density, lighting, etc.

1.2 OBJECTIVES

- Once the clues of the fire is inspected in a real time process, it has obvious distinguishing features both in terms of motion, color spectrum and textural structure.
- To detect fire early based on computer vision using Machine learning Techniques.
- To distinguish the fire image from the environment image using haar-like features and integral images.
- To alert people through alarm and email once fire is detected.

1.3 CHALLENGES

- Automatic fire detection usually senses smoke or heat and it can be difficult to set this to avoid false alarms in the immediate welding environment.
- The camera must be of good quality to detect the fire. If the camera is damaged then the detection process will not be carried further

- Automatic systems can be set to trigger sprinklers and/or to call the fire service automatically.
- The camera might get damaged due to climatic barriers.

1.4 BENEFITS

- Fire detection systems increase response times, as they are able to alert the correct people in order to extinguish the fire.
- It reduces the amount of damage to the property. Fire detection systems can be connected to sprinklers that will automatically respond when a fire is detected.
- A smoke detection unit is one of the best fire protection/prevention equipment one can have. The equipment works as an alarm system that sounds when smoke is detected
- Irrespective of how big or small your business is it is crucial that your building possess a fire alarm system.

CHAPTER 2

LITERATURE SURVEY

2.1 LITERATURE SURVEY

The proposed system has gone through various references and research over different journals and conference papers that are evolved for the past 10 years. The research and referrals of survey is conducted on different domains like AI, NLP,, Home environments that helped to developed the proposed system of AI based Control AAS. Those references and the detailed report on the references are listed as follows.

1. KB Deve, GP Hancke and BJ Silva, (2016), Design of a smart fire detection system IEEE International Conference

Fire detection systems have been promoted immensely in the past few years and have helped in the safety of people and property against fire hazards. The detection of fire hazards on the other hand can lead to unnecessary false alarms that can be very expensive if the occurrence happens in a commercial building. As well, false fire alarms have been a nuisance to the fire department and cause tie ups in resources and needless commotion that leads to panic. The problem that was addressed by this work was to detect fires and reduce the occurrence of false positives in a kitchen environment.

2. Khan Muhammad, Jamil Ahmad,Zhihan, Paolo Bellavist c 2018 IEEE. TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS:

SYSTEMS Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications

Their application in fire detection systems will substantially improve detection accuracy, which will eventually minimize fire disasters and reduce the ecological and social ramifications. However, the major concern with CNN-based fire detection systems is their implementation in real-world surveillance networks, due to their high memory and computational requirements for inference. In this paper, we propose an original, energy-friendly, and computationally efficient CNN architecture, inspired by the Squeeze Net architecture for fire detection, localization, and semantic understanding of the scene of the fire. It uses smaller convolutional kernels and contains no dense, fully connected layers, which helps keep the computational requirements to a minimum. Despite its low computational needs, the experimental results demonstrate that our proposed solution achieves accuracies that are comparable to other, more complex models, mainly due to its increased depth. Moreover, this paper shows how a tradeoff can be reached between fire detection accuracy and efficiency, by considering the specific characteristics of the problem of interest and the variety of fire data

3. Avi Bar-Massada, Todd J. Hawbaker, Susan I. Stewart, and Volker C. Radeloff 2012 Combining Satellite-Based Fire Observations and Ground-Based Lightning Detections to Identify Lightning Fires Across the

Conterminous USA IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING

Lightning fires are a common natural disturbance in North America, and account for the largest proportion of the area burned by wildfires each year. Yet, the spatiotemporal patterns of lightning fires in the conterminous US are not well understood due to limitations of existing fire databases. Our goal here was to develop and test an algorithm that combined MODIS fire detections with lightning detections from the National Lightning Detection Network to identify lightning fires across the conterminous US from 2000 to 2008. The algorithm searches for spatiotemporal conjunctions of MODIS fire clusters and NLDN detected lightning strikes, given a spatiotemporal lag between lightning strike and fire ignition. The algorithm revealed distinctive spatial patterns of lightning fires in the conterminous US While a sensitivity analysis revealed that the algorithm is highly sensitive to the two thresholds that are used to determine conjunction, the density of fires it detected was moderately correlated with ground based fire records. When only fires larger than 0.4 km were considered, correlations were higher and the root-mean-square error between datasets was less than five fires per 625 km for the entire study period. Our algorithm is thus suitable for detecting broad scale spatial patterns of lightning fire occurrence, and especially lightning fire hotspots, but has limited detection capability of smaller fires because these cannot be consistently detected by MODIS.

4. Tian Qiu, Yong Yan,Gang Lu,2011 An Autoadaptive Edge-Detection Algorithm for Flame and Fire Image Processing IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT

The determination of flame or fire edges is the process of identifying a boundary between the area where there is thermochemical reaction and those without. It is a precursor to image-based flame monitoring, early fire detection, fire evaluation, and the determination of flame and fire parameters. Several traditional edge-detection methods have been tested to identify flame edges, but the results achieved have been disappointing. Some research works related to flame and fire edge detection were reported for different applications; however, the methods do not emphasize the continuity and clarity of the flame and fire edges. A computing algorithm is thus proposed to define flame and fire edges clearly and continuously. The algorithm detects the coarse and superfluous edges in a flame/fire image first and then identifies the edges of the flame/fire and removes the irrelevant artifacts. The auto adaptive feature of the algorithm ensures that the primary symbolic flame/fire edges are identified for different scenarios. Experimental results for different flame images and video frames proved the effectiveness and robustness of the algorithm.

5. Gareth J. Roberts and Martin J. Wooster 2008 Fire Detection and Fire Characterization over Africa Using Meteosat SEVIRI IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING

Africa is the single largest continental source of biomass burning emissions and one where emission source strengths are characterized by strong diurnal and seasonal cycles. The aim of the paper is to describe the development of a fire detection and characterization algorithm for generating high temporal resolution African pyrogenic emission data sets using data from the geostationary Spinning Enhanced Visible and Infrared Imager (SEVIRI). The algorithm builds on a prototype approach tested previously with preoperational SEVIRI data and utilizes both spatial and spectral detection methods whose thresholds adapt contextually within and between imaging slots. Algorithm validation is carried out via comparison to data from ~800 temporally coincident Moderate Resolution Imaging Spectroradiometer (MODIS) scenes, and performance is significantly improved over the prior algorithm version, particularly in terms of detecting low fire radiative power (FRP) signals. On a per-fire basis, SEVIRI shows a good agreement with MODIS in terms of FRP measurement, with a small (3.7 MW) bias. In comparison to regional-scale total FRP derived from MODIS, SEVIRI underestimates this by, on average, 40% to 50% due to the no detection of many low-intensity fire pixels (FRP < 50 MW). Frequency-magnitude analysis can be used to adjust fire radiative energy estimates for this effect, and taking this and other adjustments into account, SEVIRI-derived fuel consumption estimates for southern Africa from July to October 2004 are 259–339 Tg, with emission intensity peaking after midday and reducing by more than an order of magnitude each night

6. Ying Li, Anthony Vodacek, Robert L. Kremens, Ambrose Ononye, and Chunqiang Tang SEPTEMBER 2000 A Hybrid Contextual Approach to Wildland Fire Detection Using Multispectral Imagery IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING,

The aim of the paper is to propose a hybrid contextual fire detection algorithm for airborne and satellite thermal images. The proposed algorithm essentially treats fire pixels as anomalies in images and can be considered a special case of the more general clutter or background suppression problem. It utilizes the local background around a potential fire pixel and discriminates fire pixels based on the squared Mahalanobis distance in multispectral feature space. It also employs the normalized thermal index to identify background fire pixels that should be excluded from the calculation of the statistical properties of the local background. The use of the squared Mahalanobis distance naturally incorporates the covariance of the multispectral image into the decision and requires the setting of a single detection threshold. By contrast, previous contextual algorithms only incorporate the statistical properties of individual bands and require the manual setting of multiple thresholds. Compared with the latest Moderate Resolution Imaging Spectroradiometer fire product (version 4), our algorithm improves user accuracy and producer accuracy by 1.5% and 2.6% on average, respectively, and up to 28% for some images. In addition, the novel use of the squared Mahalanobis distance allows us to create fire probability images that are useful for fire propagation modeling. As an example, we demonstrate this use for the airborne data

7. Paulo Vinicius Koerich Borges,Ebroul Izquierdo MAY 2010 A Probabilistic Approach for Vision-Based Fire Detection in Videos IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY

Automated fire detection is an active research topic in computer vision. The aim of the paper, we propose and analyze a new method for identifying fire in videos. Computer vision-based fire detection algorithms are usually applied in closed-circuit television surveillance scenarios with controlled background. In contrast, the proposed method can be applied not only to surveillance but also to automatic video classification for retrieval of fire catastrophes in databases of newscast content. In the latter case, there are large variations in fire and background characteristics depending on the video instance. The proposed method analyzes the frame-to-frame changes of specific low-level features describing potential fire regions. These features are color, area size, surface coarseness, boundary roughness, and skewness within estimated fire regions. Because of flickering and random characteristics of fire, these features are powerful discriminants. The behavioral change of each one of these features is evaluated, and the results are then combined according to the Bayes classifier for robust fire recognition. In addition, a priori knowledge of fire events captured in videos is used to significantly improve the classification results. For edited newscast videos, the fire region is usually located in the center of the frames. This fact is used to model the probability of occurrence of fire as a function of the position. Experiments illustrated the applicability of the method.

8. Ivan A. Csiszar, Jeffrey T. Morisette, and Louis Giglio Validation of Active Fire Detection From Moderate-Resolution Satellite Sensors: The MODIS Example in Northern Eurasia JULY 2006 IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING

The aim of the paper is to discuss the process of validating active fire "yes/no" binary fire detection products from moderateresolution satellite sensors. General concepts and practical issues are illustrated by the validation of the Moderate Resolution Imaging Spectroradiometer (MODIS) active fire product in Siberia. Coincident Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) imagery is used to characterize spatial patterns of flaming at sub-MODIS pixel scale. It is shown that for proper evaluation reference fire observations are needed at the scale of the satellite pixel, as only 60% of the MODIS footprints contain single contiguous clusters of ASTER fire pixels. In Siberia the size of a single ASTER fire cluster within the MODIS footprint that has a 50% probability of being flagged as "fire" is 60, compared to 45 in the Brazilian Amazon, whereas previous radiative transfer simulations suggested similar detection probabilities. The lower-than-expected detection rates in Siberia are largely attributable to flaming underneath heavy smoke, which is not detected by the current MODIS algorithm. Pixel-based and cluster-based omission error rates are derived, and it is shown that the probability of flagging as “fire” a MODIS pixel which contains a given number of 30-m ASTER fire pixels is typically 3–5 times lower than detecting a contiguous cluster with the same number of ASTER fire pixels. The procedures described are recommended for a consensus active fire validation protocol, but with the inclusion of multiplatform sensor configurations to complement the near-nadir angular sampling from single-platform observations such as MODIS and ASTER on Terra. Index Terms—Fire detection, multisensor systems.

9. XINCHU HUANG1 AND LIN DU April 11, 2020,Fire Detection and Recognition Optimization Based on Virtual Reality Video Image

Accidental fire is a natural disaster that seriously threatens public safety. In recent years, accidental fire has frequently occurred in many places, including superstores, communities and forests, yielding huge losses to production and human life. After several decades of development, virtual reality technology has matured quickly and has changed people’s lifestyles by being widely applied in many fields. For example, VR technology has been used to manage accidental fire in industry, agriculture, hospitals, aviation, aerospace, and firefighting. Thus, virtual fire environment technology has become integral to future fire protection. Due to their detection principles or system structures, traditional fire detectors, which include temperature detectors, smog detectors and optical detectors, usually have inherent defects or application restrictions. Because flames and smog have specific colors, textures, shapes and other image features, people have begun to consider using computer visual features to improve the efficiency of fire detection (e.g., video flame detection technology based on image processing)

10 Gao Xu, Qixing Zhang, Dongcai Liu, Gaohua Lin, Jinjun Wang, Yongming Zhang Adversarial Adaptation from Synthesis to Reality in Fast Detector for Smoke Detection, Video smoke detection is a promising method for early fire prevention. However, it is still a challenging task for application of video smoke detection in real world detection systems, as the limitations of smoke image samples for training and lack of efficient detection algorithm. This paper proposes a method based on two state-of-the-art fast detectors, single shot multi box detector and multi-scale deep convolutional neural network, for smoke detection using synthetic smoke image samples. The virtual data can automatically offer rich samples with ground truth annotations. However, the learning of smoke representation in the detectors will be restricted by the appearance gap between real and synthetic smoke samples, which will cause significant performance drop. To train a strong detector with synthetic smoke samples, we incorporate the domain adaptation into the fast detectors. A series of branches with the same structure as the detection branches are integrated into the fast detectors for domain adaptation. We design an adversarial training strategy to optimize the model of the adapted detectors, to learn a domain-invariant representation for smoke detection. The domain discrimination, domain confusion and detection are combined in the iterative training procedure.

S.NO	TITLE	YEAR	METHODOLOGY	BENEFITS
1	Fire Detection and Recognition Optimization Based on Virtual Reality Video Image	2020	To achieve this, the support vector machine (SVM) method in machine learning the RS-SVM classifier model based on parameter optimization proposed	Fire detection technology based on video images can avoid many flaws in conventional methods and detect fires.
2	A Probabilistic Approach for Vision-Based Fire Detection in Videos	2010	Analyzes the frame-to-frame changes of specific low-level features describing potential fire regions.	Computer vision-based fire detection algorithms are usually applied in closed-circuit Television surveillance.

S.NO	TITLE	YEAR	METHODOLOGY	BENEFITS
3	Validation of Active Fire Detection From Moderate-Resolution Satellite Sensors: The MODIS Example in Northern Eurasia	2006	Coincident Advanced Space borne Thermal Emission and Reflection Radiometer (ASTER) imagery is used to characterized spatial patterns of flaming at sub-MODIS pixel scale	This paper discusses the process of validating active fire “yes/no” binary fire detection products from moderate resolution satellite sensors.
4	Combining Satellite-Based fire Observations and Ground-Based Lightning Detections to Identify lightning Fires Across the Conterminous USA	2012	Our goal here was to develop and test an algorithm that combined MODIS fire detections with lightning detections from the National Lightning	These results may enhance our understanding of large scale patterns of lightning fire activity, and can be used to identify the broad scale factors controlling fire occurrence.

S.NO	TITLE	YEAR	METHODOLOGY	BENEFITS
5	Design of a smart fire detection system	2016	Wireless Sensor Network (WSN) and Global System for Mobile (GSM) communication to detect fires effectively and reduce false positives.	In this paper, we discuss the design and implementation of a smart fire detection
6	A Hybrid Contextual Approach to Wildland Fire Detection Using Multispectral Imagery	2005	The use of the squared Mahalanobis distance naturally incorporates the covariance of the multispectral image into the decision and requires the setting of a single detection threshold.	The proposed algorithm essentially treats fire pixels as anomalies in images and can be considered a special case of the more general clutter or background suppression problem

S.NO	TITLE	YEAR	METHODOLOGY	BENEFITS
7	Fire Detection and Fire Characterization Over Africa Using Meteosat SEVIRI	2010	Carbon, fires, remote sensing	The algorithm builds on a prototype approach tested previously with preoperational SEVIRI data and utilizes both spatial and spectral detection methods
8	Early fire smoke movements and detection in high large volume spaces Build	2019	An approach to the study of forest fire prediction methods based on artificial intelligence has been suggested.	Forest fire risk forecast algorithm is built on help vector machines Forest fire risk forecast algorithm is built
9	Fire and smoke detection without sensors: image processing-based approach Proc.	2017	in this paper authors have considered an area called Lebanon to predict the occurrence of forest fire	Artificial Neural Networks to evolve in order to anticipate forest fires

S.NO	TITLE	YEAR	METHODOLOGY	BENEFITS
10	A system for real-time fire detection Proc. IEEE Computer Vision and Pattern Recognition	2020	in their paper have presented the by analyzing a series of pixel values, an image mining technique	It is a precursor to image-based flame monitoring, early fire detection, fire evaluation, and the determination of flame and fire parameters

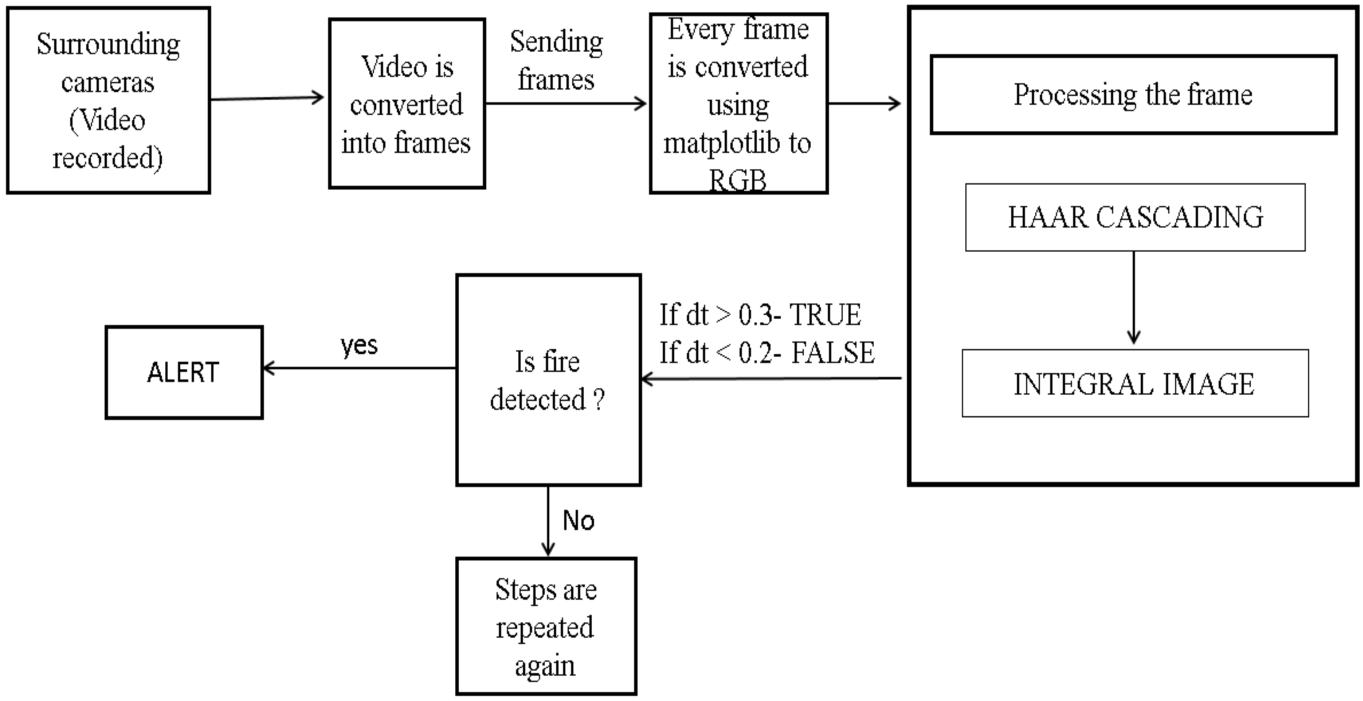
Table No. 2.1 Summary of Literature Survey

CHAPTER 3

SYSTEM ANALYSIS AND DESIGN

3.1 ARCHITECTURE OF PROPOSED SYSTEM

The first step is the surrounding cameras will capture the video in real time. Then the next step is video is converted into frames and the frames are sent for conversion each and every frames are converted into gray scale used for converting an image from the other color spaces, dimension reduction and reduce model complexity. It varies between complete black and white. And next step is processing the frame using Haar Cascading algorithm to detect fire is based on the machine learning concept using the Haar-like features , Haar features are similar to convolutional kernels, which are used to detect the presence of those features in the given image. The grayscale has some features some of the regions are black, some regions are white, and some region is slightly darker or vice-versa from each other which helps the machine to understand an image. Then integral image, in an integral image the value of pixel (x,y) is the sum of pixels above and to the left of (x,y) and the calculation of sum of the pixels inside any given rectangle using four values at the corner of the rectangle if the integral value is greater than 0.3 the fire is detected the alert and email sent. If the integral value is less than 0.2 the fire is not detected the steps are repeated again.



3.2 METHODOLOGY

For fire detection, the traditional method is to use a sensor for detection. One of the defects is the high false rate because the trigger alarm is based on the concentration of particles or the surrounding temperature and is therefore easily disturbed by the surroundings. The proposed technique receives run based color model because of its less multifaceted nature and viability. YCbCr shading space adequately isolates luminance from chrominance contrasted with other color spaces like RGB color space. The proposed technique isolates fire pixels as well as isolates high fire focus pixels by assessing accurate parameters of flame images in YCbCr color space.

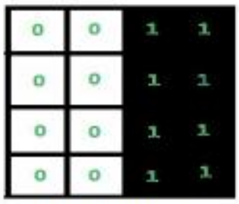
DETECTION

The procedure for detecting fire classifies images based on the value of simple features from an image. While detecting the fire, first of all, image is converted into grayscale since it is easy to work with, and it has less data rather than RGB color images. The algorithm outlines a box and searches for the fire in the image,, the box is searching for Haar-like features. Along with small steps, the box detects the features like edges, brightness level from the image of fire, and then data collected by boxes are put together, which helps to determine where the fire is located

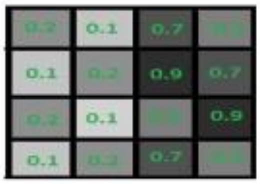
HAAR-LIKE FEATURES

Haar features are similar to convolutional kernels, which are used to detect the presence of those features in the given image. The grayscale has some features some of the regions are black, some regions are white, and some region is slightly darker or vice-versa from each other which helps the machine to understand an image. Haar features like edge features, line features, center-surround features rectangle is moved around the image. The color white and black is decided according to the region. Which compares the brightness and extracts features by assigning some values as it goes and inspects through that region in an image.

Applying Haar-features



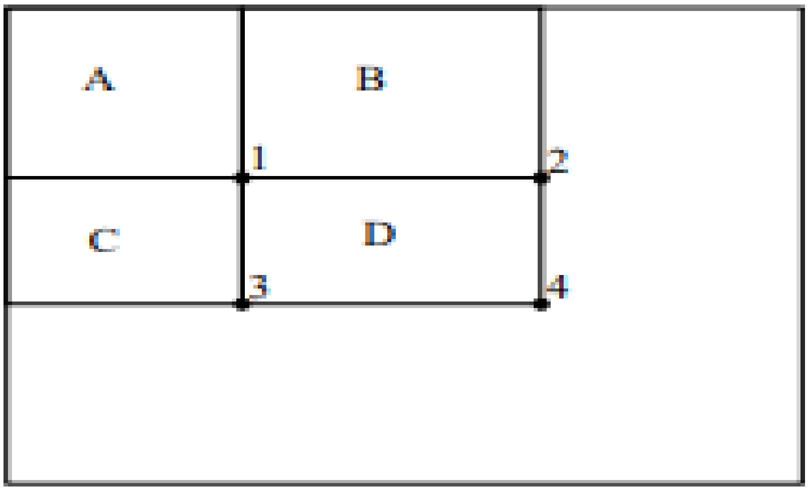
Ideal values



Real values

INTEGRAL IMAGE

The integral image plays an important role. It gives a boost to the calculation, which makes the approach as fast as possible because there are thousands of pixels that need to be calculated. While applying rectangular haar features the sum of pixels in unshaded rectangles sides are subtracted from the sum of the pixel in the shaded side of rectangles. Even for a small size images there are lots of features (over 160,000 for a 24x24 image). Since due to large number of features the algorithm requires iterating overall features, the features must be computed efficiently. So to solve this issue integral image is introduced. The sum of the pixel in rectangle D can be calculated with reference to four arrays. The value of the image at location 1 is the sum of pixels in a rectangle. Respectively value at position 2 is A+B, value at position 20 3 is A+C, and value at position 4 is A+B+C+D.

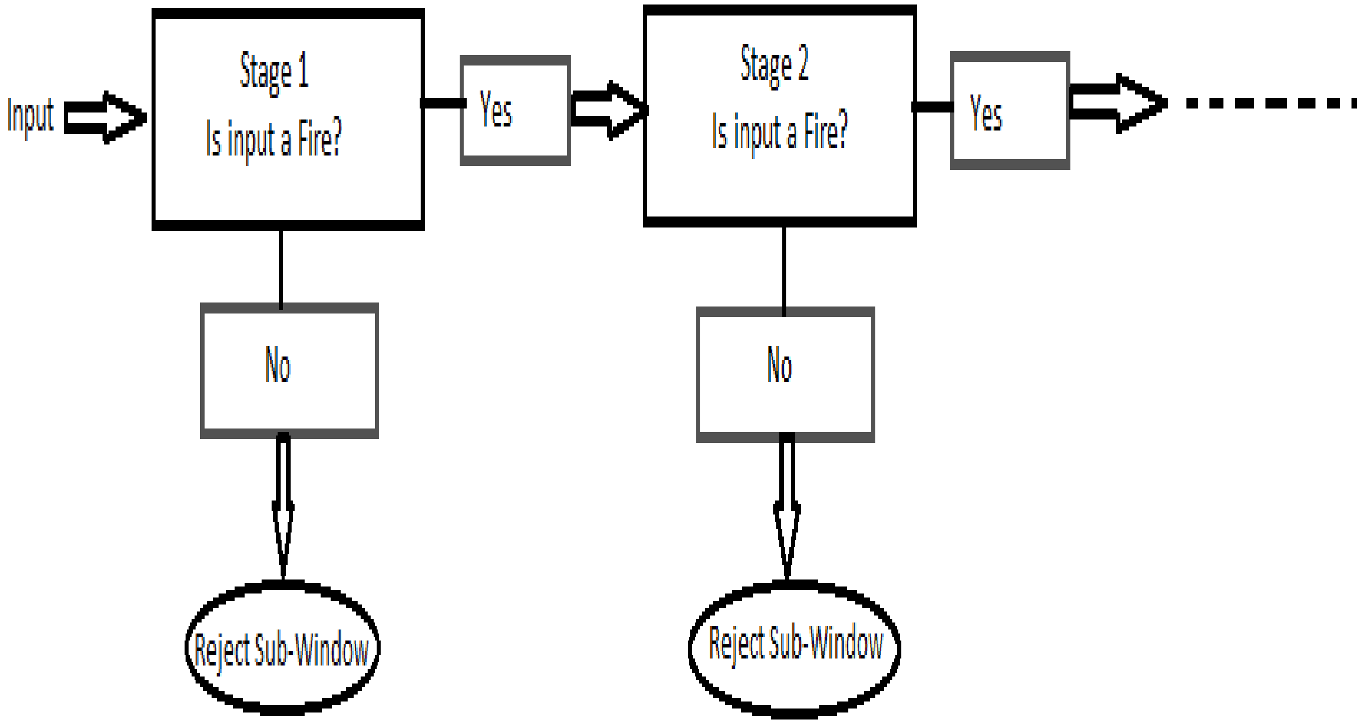


ADAPTIVE BOOSTING (ADA BOOSTING)

- Adaboost algorithm is a learning algorithm used to train the classifier and selects the best subset of features. The algorithm learns from the data that is given and determine the false positive and true negatives in data.
    - F(x) = a1f1(x) + a2f2(x) +....
- Here F(x) is the strong classifier, and a1f1(x), a2f2(x) are weak classifiers where a1, a2 are the weights, and f1, f2 are the features. So the strong classifier is made up of many weak classifiers since one weak classifier is not good as adding more classifiers makes the algorithm stronger and accurate, which is called an ensemble.

CASCADING

Cascade classifier is used for the accuracy of identification. It is composed of several stages consisting of a strong classifier. Those strong classifiers are passed by so all the features are grouped in several stages where each stage has a certain number of features. The use of these several stages is used to determine whether the given input sub window has features of fire or not if there are no features of fire, then the given sub window is discarded and fails to go for other stages. There are some stages stage 1 and stage 2. Usually, the first few stages will contain very less numbers of features. If the window fails, it is discarded if not apply the second stage of features and continue the process. The window or stage which passes all the features of fire, then it is detected as fire.



3.3 EXISTING SYSTEM

The existing system uses Convolutional neural networks (CNN) , the major concern with CNN-based fire detection systems is their implementation in real-world surveillance networks, due to their high memory and computational requirements for inference. They uses correlation based analysis and transformation methods to study the strength of a relationship between two, numerically measured, continuous variables.

The system uses Wireless Sensor Network (WSN) and Global System for Mobile (GSM) to discuss the design and implementation of a smart fire detection system to detect fires effectively and reduce false positives.

In the existing system the process of validating active fire “yes/no” binary fire detection products from moderate resolution satellite sensors is used. General concepts and practical issues are illustrated by the validation of the Moderate Resolution Imaging Spectroradiometer (MODIS) active fire product in Siberia.

The system uses Data Mining Approach to Predict Forest Fires using Meteorological Data and Circle-based Approximation to Forest Fires with Distributed Wireless Sensor Networks using clustering. And they uses Lebanon to predict the occurrence of forest fire.

3.4 REQUIREMENT SPECIFICATION

HARDWARE SPECIFICATION

- System: Pentium i3 Processor.

- Hard Disk: 500 GB.
- Monitor: 15’’ LED
- Input Devices: Keyboard, Mouse
- Ram: 2 GB
- Camera: Min 24MP

SOFTWARE SPECIFICATION

- Operating system: Windows 7.
- Coding Language: python
- Tool: Anaconda Navigator
- Libraries: cv2, numpy, math, pygame.

3.5 UML DIAGRAM

Unified Modelling Language (UML) is simply another graphical representation of a common semantic model. The proposed system has been designed by using use case diagram, class diagram, sequence diagram, collaboration diagram, state chart diagram and component diagram.

3.3.1 USECASE

In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors.

3.4.2 CLASS DIAGRAM

Class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction and it is used for general conceptual modelling of the structure of the application, and for detailed modelling translating the models into programming code.

Class diagrams can also be used for data modelling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

3.4.3 SEQUENCE DIAGRAM

The control flow between various participants or entity roles of the corresponding system in the form of messages is represented in the Sequence Diagram. The participants are represented within the rectangular object. The swim line or the lifeline that is dragged below every participant represents the lifetime of the corresponding participant.

The UML representation of a class is rectangle containing three compartments stacked vertically. The top compartments shows the class's name. The middle compartments list the class's attributes. The bottom compartment lists the class operations known as the methods of the class.

A class diagram consists of any number of classes which will be connected by the lines, which may have arrows at one or both ends, connecting the boxes. These lines define the relationships, also called associations, between the classes. These lines will have multiplicity to represent the number of instances of the classes.

3.4.4 ACTIVITY DIAGRAM

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another with different components of activity diagram. Some of the components of activity diagram Start/Stop symbol, Action symbol, Joint and Fork symbol, Decision symbol, Connector symbol.

3.4.5 COLLABORATION DIAGRAM

Collaboration diagram is defined as one of the interaction diagram, which consists of the set of objects related in a particular context and interaction among those objects. The collaboration diagram is also called as the set of message exchange among the objects within the collaborative nature of message exchange between the corresponding objects. There are three primary elements of a collaboration diagram Objects, Links, Messages.

CHAPTER 4

TECHNICAL BACKGROUND

4.1 LIST OF MODULES

The proposed system is based on the Haar Cascade classifier and OpenCV which is very popular in object detection through the image or any other video feeds. The process is divided into three different phases which employs various algorithms and methods for implementation.

- MODULE 1 : Fire detection
- MODULE 2: Alarm
- MODULE 3: Notification mail

4.1.1 FIRE DETECTION MODULE

The procedure for detecting fire classifies images based on the value of simple features from an image. While detecting the fire, first of all, image is converted into grayscale since it is easy to work with. The algorithm outlines a box and searches for the fire in the image, the box is searching for Haar-like features. Along with small steps, the box detects the features like edges, brightness level from the image of fire, and then data collected by boxes are put together, which helps to determine where the fire is located.

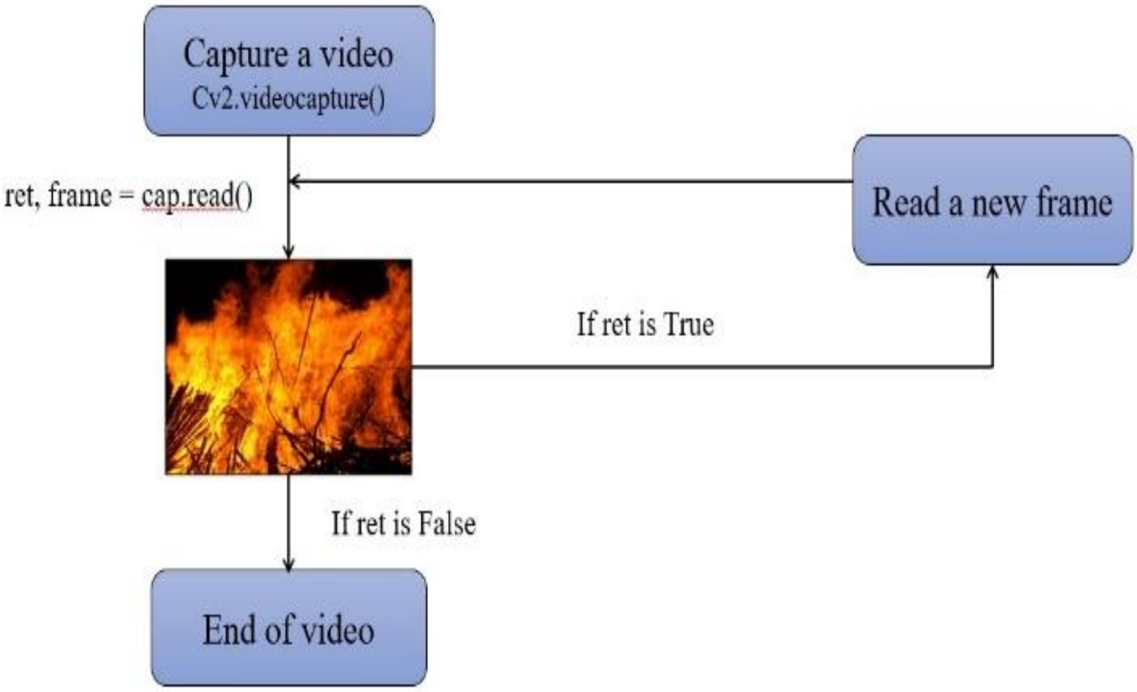


Fig-1: Shows the flow of detection process

3.2 ALARM MODULE

A siren is a loud noise-making device. Civil defense sirens are mounted in fixed locations and used to warn of natural disasters or attacks. If the fire is detected the alarm is played and notifies the monitor. Putting forward a while loop which takes the argument of the time, the user wants to set the alarm on and automatically breaks when the time is up, with sound.

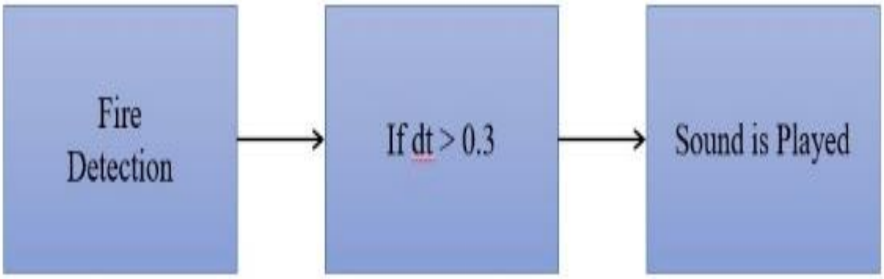


Fig2: alarm

3.3 NOTIFICATION MAIL

When you click send, the message is transmitted from your computer to the server associated with the recipient's address. This process typically occurs via several other servers before the message gets to its intended recipient's mailbox. Electronic mailboxes are central to how emails work for the end-user. The standard protocol used for sending Internet e-mail is called SMTP (Simple Mail Transfer Protocol). The SMTP protocol is used to both send and receive email messages over the Internet. When a message is sent, the email client sends the message to the SMTP server.

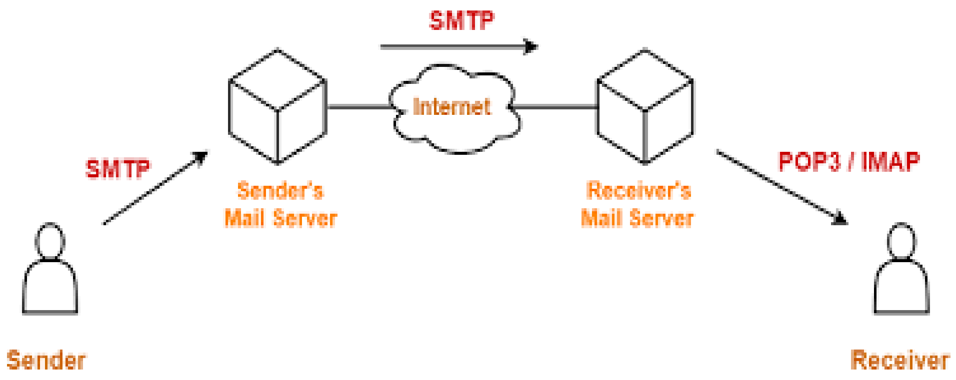


Fig-3: Shows the flow of E-Mail

4.3.1 HARDWARE REQUIREMENT

PENTIUM I3 PROCESSOR.

Pentium is a 7th generation processor whereas Core i3 is a 9th generation processor. Pentium processors have less cache memory as compared to that of the Core i3 processor. Core processor has more memory channels while Pentium has a low number of memory channels. The Core i3-10100 is the best Core i3 CPU for gaming. It comes with four cores and eight threads and is capable of reaching similar gaming performance to the Ryzen 3 3300X and the Core i7-7700K.

CAMERA OF 24 MEGAPIXEL

Camera of 24 megapixel should be provided as the detection depends totally on the visual characteristics of the fire. The pixels are arranged across the sensor in a grid. To calculate the number of megapixels we need to multiply the number of pixels across the length and the width of the sensor. This number represents the total resolution of the sensor and is measured in millions of pixels, or megapixels. If the camera is of lesser resolution then the fire might not clearly detected to avoid the accident.

4.3.2 SOFTWARE REQUIREMENT

ANACONDA NAVIGATOR

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda distribution that allows users to launch applications and manage conda packages, environments and channels without using command-line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for Windows, macOS and Linux

OPENCV

Open CV is an open-source package and machine learning library, which is designed for real-time computer vision applications. Open CV is the cross-platform library that supports many programming languages like Python, Java, C++, C, etc. It is one of the most widely used libraries for implementing video detection, image detection, deep learning application, machine learning, human-computer interaction, 2D, and 3D feature toolkits. The library has more than 2500 algorithms, which include both computer vision and machine learning algorithms. It is used for video analysis, CCTV footage analysis and image analysis. This also makes it easier to integrate with other libraries that use Numpy such as SciPy and Matplotlib.

NUMPY

NumPy stands for Numerical Python. NumPy is a Python library used for working with arrays. Arrays are very frequently used in data science, where speed and resources are very important. In Python we have lists that serve the purpose of arrays, but they are slow to process. NumPy aims to provide an array object that is up to 50x faster than traditional Python lists. The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy. It also has functions for working in domain of linear algebra, Fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

PYGAME

Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used with the Python programming language. Pygame uses the Simple

Direct Media Layer (SDL) library, with the intention of allowing real-time computer game development without the low-level mechanics of the C programming language and its derivatives. This is based on the assumption that the most expensive functions inside games can be abstracted from the game logic, making it possible to use a high-level programming language, such as Python, to structure the game. Other features that SDL does have include vector math, collision detection, 2D sprite scene graph management, MIDI support, camera, pixel-array manipulation, transformations, filtering, advanced free type font support, and drawing.

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

The project aimed to detect fire with a different approach rather than using an existing system. As technology is getting better and better as to keep it up with the technology and to minimize the limitations also, the new system has created. By using image processing technology for detecting the fire, these limitations can be reduced because in this system camera acts like a human eye, as it detects a fire, the video is captured, and the image is processed using the software alert user. It can be used everywhere e.g. Hospital, railway station, forest etc. The designed prototype successfully detects fire. Gives the review analysis, designing system, and algorithm, test, and result. Currently, we not used the systems like a smoke detector and sprinkler water discharge systems, but in future it can be included.

6.2 FUTURE ENHANCEMENT

The demonstration of a quantitative analysis to compare the performance of different strategies. Comparing this method with well-known fire detection algorithms, which are based on YOLO networks and DL approaches. By using the results in their papers for comparison, but it is not sure whether they are true because source codes and datasets of these methods are not publicly available to check the real performance. The computed metrics such as F-measure (FM), precision, and recall, as in our earlier study. The FM score is the weighted average that balances measurements between the means of precision and recall rates. Hence, this score considers both false positives and false negatives. Intuitively, it is not easy to understand accuracy, but FM is more common than accuracy. Accuracy works best if false positives and false negatives have similar costs. If the costs of false positives and false negatives are different, it is better to consider both precision and recall. Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. Recall is the ratio of correctly predicted positive observations to all observations in the actual class, as expressed in Equation. The proposed method's average of FM, recall, and precision was 98.9%. False detection occurred in 1.1% of cases, owing to the blurring of objects at night. To calculate the average precision and recall rates of the shadow remover methods, the equations can be used.

APPENDIX

SAMPLE CODE

```
import cv2

import numpy as np

import math

import pygame

pygame.init()

pygame.mixer.music.load("alarm.wav")

windowName = "OpenCV Video Player"

cv2.namedWindow(windowName)

cap = cv2.VideoCapture(0)#REAL TIME CAMARA 0 FOR PRIMARY

ret, lframe = cap.read()

Ll, A, B = cv2.split(lframe)

SI = np.zeros(np.shape(Ll))

print(SI)

NOl = 0

NOc = 0

CGO = 0

x = 1

dt = 0

nop = 0

count = 0

mc = 0
```

```
#Now start reading the footage in infinite while loop ,
while (ret):
    ret, frame = cap.read()
    if ret:
        Ll, A, B = cv2.split(lframe)
        #isOpened() check whether the camera has initialized properly
        ret, frame = cap.read()
        cframe = frame
        cv2.imshow("FIRE", frame)
        # OpenCV reads images in BGR format whereas Matplotlib reads image in RGB format. So to get output image in RGB form, we'll have to convert the frame into RGB from BGR.
        b, g, r = cv2.split(frame)
        rt = 230
        gb = cv2.compare(g, b, cv2.CMP_GT)
        rg = cv2.compare(r, g, cv2.CMP_GT)
        rrt = cv2.compare(r, rt, cv2.CMP_GT)
        rgb = cv2.bitwise_and(rg, gb)
        im = cv2.bitwise_and(rgb, rrt)
        # cv2.imshow("RGB",im)
```

```
t = 5

p = 1

k = cv2.getStructuringElement(cv2.MORPH_CROSS, ((t, t,)))

# print(k)

dil = cv2.dilate(im, k, iterations=p)

er = cv2.erode(im, k, iterations=p)

fin = cv2.bitwise_and(er, dil)

# cv2.imshow("FIRE",frame)

# cv2.imshow("segmented fire",cv2.bitwise_and(frame,frame,mask=fin))

img_ycrcb = cv2.cvtColor(frame, cv2.COLOR_BGR2YCR_CB)

Y, Cr, Cb = cv2.split(img_ycrcb)

Ym = np.mean(Y)

Crm = np.mean(Cr)

Cbm = np.mean(Cb)

I1 = cv2.compare(Y, Ym, cv2.CMP_GT)

I2 = cv2.compare(Cb, Cbm, cv2.CMP_LT)

I3 = cv2.compare(Cr, Crm, cv2.CMP_GT)

I12 = cv2.bitwise_and(I1, I2)

I23 = cv2.bitwise_and(I2, I3)

I123 = cv2.bitwise_and(I12, I23)

# cv2.imshow("I123",I123)

cbcrdiff = cv2.absdiff(Cb, Cr)
```

```
asd = cv2.compare(cbcrdiff, 40, cv2.CMP_GT)

# cv2.imshow("asd", asd)

img_cie = cv2.cvtColor(frame, cv2.COLOR_BGR2Lab)

L, a, b = cv2.split(img_cie)

Lm = np.mean(L)

am = np.mean(a)

bm = np.mean(b)

R1 = cv2.compare(L, Lm, cv2.CMP_GT)

R2 = cv2.compare(a, am, cv2.CMP_GT)

R3 = cv2.compare(b, bm, cv2.CMP_GT)

R4 = cv2.compare(b, a, cv2.CMP_GT)

R12 = cv2.bitwise_and(R1, R2)

R34 = cv2.bitwise_and(R3, R4)

R14 = cv2.bitwise_and(R1, R4)

# cv2.imshow("R14",R14)

kl = cv2.getStructuringElement(cv2.MORPH_CROSS, ((t, t,)))

e_R14 = cv2.erode(R14, kl, iterations=p)

d_R14 = cv2.dilate(R14, kl, iterations=p)

bin_cie = cv2.bitwise_and(e_R14, d_R14)

# cv2.imshow("cie_res",cv2.bitwise_and(frame,frame,mask=bin_cie))

im = cv2.bitwise_and(bin_cie, im)

# cv2.imshow("RGBCIE",im)
```

```
cframe = cv2.cvtColor(frame, cv2.COLOR_BGR2Lab)
Lc, A, B = cv2.split(cframe)
DIfd = cv2.absdiff(Lc, Ll)
#   cv2.imshow("difd",DIfd)
u_DIfd = np.mean(DIfd)
sd_DIfd = np.std(DIfd)
if u_DIfd + sd_DIfd >= 10:
    Tfd = u_DIfd + sd_DIfd
else:
    Tfd = 10
_, FD = cv2.threshold(DIfd, Tfd, 255, cv2.THRESH_BINARY)
# cv2.imshow("difd2",FD)
# print(FD)
cntr = cv2.compare(FD, 0, cv2.CMP_EQ) / 255
SI = np.add(cntr, SI)
# print(SI)
MPM = cv2.bitwise_and(FD, im)
MPM = cv2.bitwise_and(MPM, asd)
MPM = cv2.erode(MPM, kl, iterations=p)
# cv2.imshow("MPM", MPM)
CF = cv2.bitwise_and(im, MPM)
```

```
# cv2.imshow("candi", CF)
retval, labels, stats, centroids = cv2.connectedComponentsWithStats(CF)
for i in range(retval):
    if i > 0:
        if stats[i][4] > 2:
            NOc = NOc + stats[i][4]
        # print(NOc)
    if NOc > NOl:
        CGO = CGO + 1
    NOl = NOc
    NOc = 0
    fps = 24
    if x % fps == 0:
        dt = CGO * 1.0 / fps
        print(dt)
        CGO = 0
        x = 1
    x = x + 1
    # print(dt)
    lframe = cframe
    if dt > 0.3:
        count = count + 1
```

```
                if count > 10:
                    mc = mc + 1
                if mc % 15 == 1:
                    print("fire")
                    try:
                        pygame.mixer.music.play()
                    except:
                        pass
                    count = 0
            if dt < 0.2:
                print("not fire")
        k = cv2.waitKey(1)
        if k == ord('q'):
            break
cv2.destroyAllWindows()
cap.release()
```

REFERENCES

REFERENCES

1. N. Dziengel, G. Wittenburg, and J. Schiller, "Towards distributed event detection in Wireless Sensor Networks," in Adjunct Proc. of 4th IEEEfACM IntI. Conf. on Distributed Computing in Sensor Systems (DCOSSa: AZ’08), Santorini Island, Greece, 2020.
2. S. Jarupadung, "Distributed event detection and semantic event processing," in The 6th ACM International Conference on Distributed Event- Based Systems (DEBS 2020)(Doctoral Symposium), 2020.
3. Y. Li and L. E. Parker, "Detecting and monitoring time-related abnormal events using a Wireless Sensor Networks and mobile robot," in Intelligent Robots and Systems, 2019. IROS 2019. IEEEfRSJ International Conference on. IEEE, 2019.
4. P. Radivojac, U. Korad, K. M. Sivalingam, and Z. Obradovic, "Learning from class-imbalanced data in Wireless Sensor Networks," in Vehicular Technology Conference, 2020. VTC 2020-Fall. 2020 IEEE 58th, vol. 5, IEEE, 2020.
5. Z.-J. Zhang, J.-S. Fu, H.-P. Chiang, and Y.-M. Huang, "A novel mechanism for fire detection in subway transportation systems based on Wireless Sensor Networks," International Journal of Distributed Sensor Networks, vol. 2019, 2019.
6. T. Celik, H. Demirel, H. Ozkaramanli, and M. Uyguroglu, “Fire detection in video sequences using statistical color model,” in Proc. IEEE Int. Conf. Acoust. Speech Signal Process. vol. 2. Toulouse, France, May 2006, p. II.

7. P. V. K. Borges, J. Mayer, and E. Izquierdo, “Effificient visual fifire detection applied for video retrieval,” in Proc. 16th Eur. Signal Process. Conf., Aug. 2008.
8. B. U. Toreyin and A. E. Cetin, “Online detection of fifire in video,” in Proc. IEEE Conf. Comput. Vision Pattern Recognit., Jun. 2007, pp. 1–5.
9. C. Liu and N. Ahuja, “Vision-based fifire detection,” in Proc. Int. Conf. Pattern Recognit., vol. 4. Aug. 2004, pp. 134–137.
10. T. Celik, H. Ozkaramanli, and H. Demirel, “Fire pixel classifification using fuzzy logic and statistical color model,” in Proc. Int. Conf. Acoust. Speech Signal Process, vol. 1. Apr. 2007, pp. 1205–1208.
11. T. Chen, P. Wu, and Y. Chio, “An early fifire-detection method based on image processing,” in Proc. IEEE Int. Conf. Image Process., vol. 3. Oct. 2004, pp. 1707–1710.
12. S. C. Liew, A. Lim, and L. K. Kwoh, “A stochastic model for active fifire detection using the thermal bands of modis data,” IEEE Geosci. Remote Sensing Lett. vol. 2, no. 3, pp. 337–341, Jul. 2005.
13. Y. Li, A. Vodacek, R. L. Kremens, A. Ononye, and C. Tang, “A hybrid contextual approach to wildland fifire detection using multispectral imagery,” IEEE Trans. Geosci. Remote Sensing, vol. 43, no. 9, pp. 2115– 2126, Sep. 2005.
14. H. A. Mooney, P. M. Vitousek, and P. A. Matson, “Exchange of materials between terrestrial ecosystems and the atmosphere,” Science, vol. 238, pp. 926–932, 1987.
15. T. L. Clark, J. L. Coen, and D. Latham, “Description of a coupled atmosphere-

fire model,” Int. J. Wildland Fire, vol. 13, pp. 2783–2799, 1992.

16. J. Mandel, M. Chen, J. L. Coen, C. C. Douglas, L. Franca, C. Johns, R. Kremens, A. Puhalskii, A. Vodacek, and W. Zhao, “Dynamic data driven wildfire modeling,” in Dynamic Data Driven Applications Systems, F. Darema, Ed. Amsterdam, The Netherlands: Kluwer, 2005, to be published.
17. J. Dozier, “A method for satellite identification of surface temperature fields of subpixel resolution,” Remote Sens. Environ., vol. 11, pp. –229, 1981.
18. Y. J. Kaufman, C. O. Justice, L. P. Flynn, J. D. Kendall, E. M. Prins, L. Giglio, D. E. Ward, W. P. Menzel, and A. W. Setzer, “Potential global fifire monitoring from EOS-MODIS,” J. Geophys. Res., vol. 103, pp. 32 215–32 238, 1998.